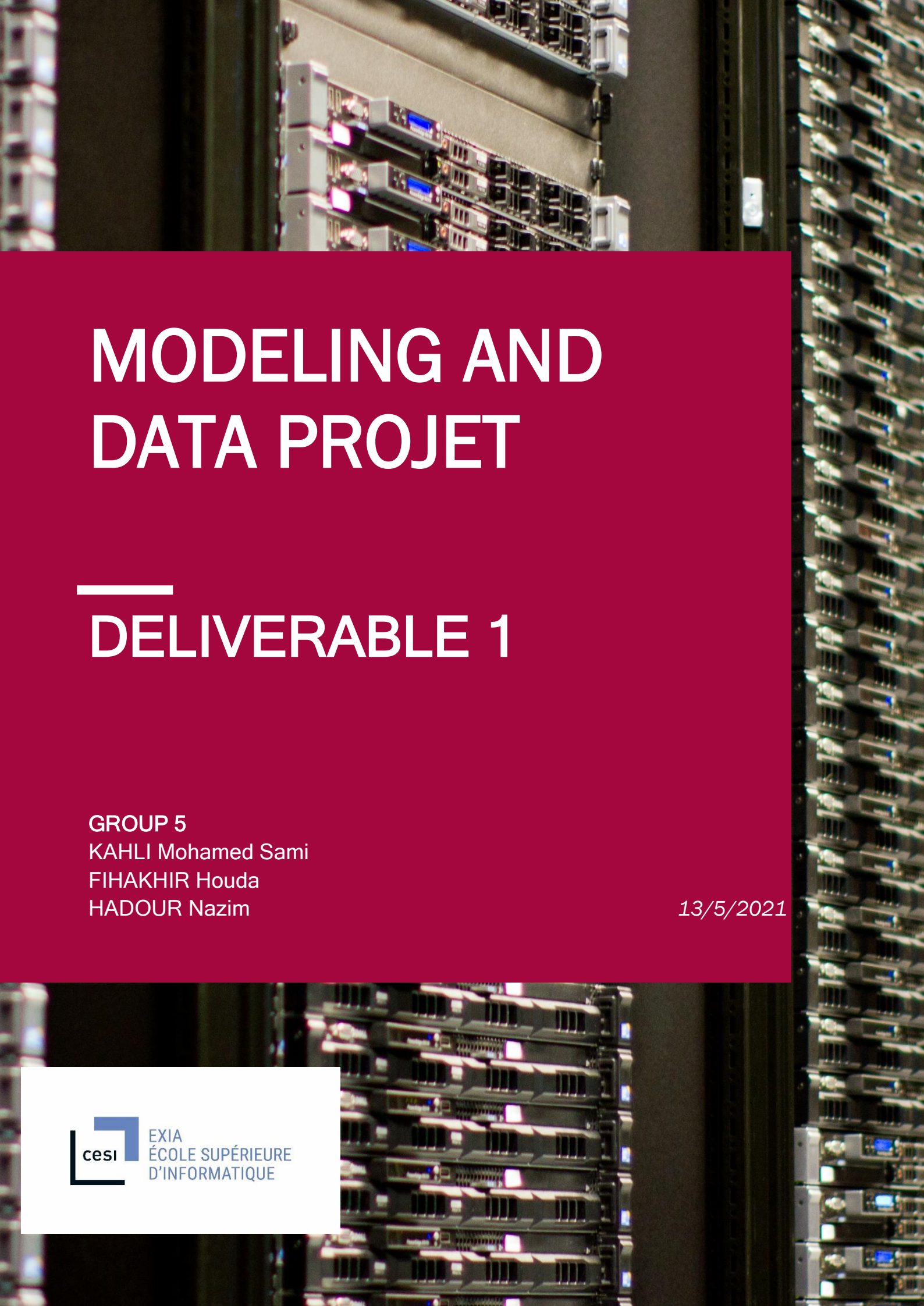


MODELING AND DATA PROJET

DELIVERABLE 1

GROUP 5
KAHLI Mohamed Sami
FIHAKHIR Houda
HADOUR Nazim

13/5/2021

cesi EXIA ÉCOLE SUPÉRIEURE D'INFORMATIQUE

Table of content :

1. PROJECT INTRODUCTION

A British travel agency wants to digitize its system, so they went tasking to us to do that.
We have access to the project specifications, as well as extracts from "raw" interviews.
The customer compartmentalized their business process into several categories:

- Client management
- Staff management
- Ticket office management
- Management of means of transport
- Management of stages
- Statistics management

All of these categories each have their own set of features.
We have until the end of May (26/05) to complete the project.

2. TEAM MEMBERS PRESENTATION

Our group is group 5, it is composed of 3 students as requested. He is composed of KAHLI Mohamed Sami, who will be the group leader, as well as of HADOUR Nazim and of FIHAKHIR Houda.

3. DATA DICTIONARY

A **data dictionary** is a collection of the names, definitions, and attributes for data elements and models.

Our case is about a travel agency.

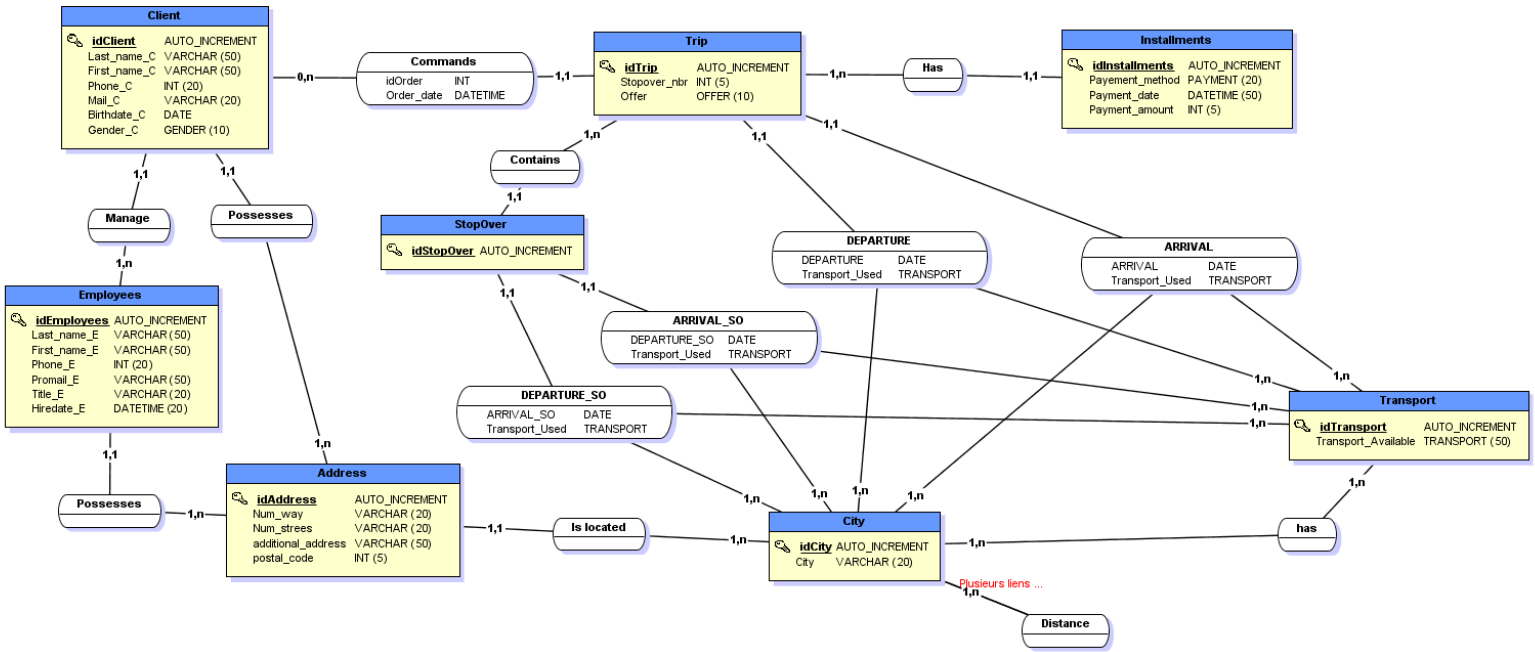
	Signification	Type	length
Last_name_C	client last name	VARCHAR	50
First_name_C	client first name	VARCHAR	50
Phone_C	client phone number	INT	20
Mail_C	client mail	VARCHAR	20
Birthdate_C	client birthdate	DATE	
Gender_C	F/M	ENUM	10
Last_name_E	employee last name	VARCHAR	50
First_name_E	employee first name	VARCHAR	50
Phone_E	employee phone number	INT	20
Promail_E	employee profesionnal mail	VARCHAR	50
Title_E	employee title	VARCHAR	20
Hiredate_E	employee hiredate	DATETIME	20
Num_way	way number	VARCHAR	20
Num_street	street number	VARCHAR	20
Aditional_address	floor/building number	VARCHAR	50
Postal_code	postal code	INT	5
Stopover_nbr	stopover number	INT	5
Offer	Forfait/KM	ENUM	10
Order_date	order date	DATE	
Departure	departure date	DATE	
Departure_SO	stopover departure date	DATE	
Arrival	arrival date	DATE	
Arrival_SO	stopover arrival date	DATE	
Payment_method	CB/PAYPAL/CHEQUES/ESPECES	ENUM	20
Payment_date	payement date	DATETIME	
Payment_amount	amount	INT	5
Transport_available	bus/car/train/plane/boat	ENUM	50
City	cities	VARCHAR	20
Transport_used	bus/car/train/plane/boat	ENUM	

Dependency structure Matrix (DSM) is a square matrix used to represent the project dependencies. A quick look at the DSM should convey what are the other tasks that are dependent on the output of a given task. Its visually compact way to represent complex systems is one of its biggest advantages. Here is ours:

		2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
2	Last_name_C	X																												
3	First_name_C	1	X																											
4	Phone_C	1		X																										
5	Mail_C	1			X																									
6	Birthdate_C	1				X																								
7	Gender_C	1					X																							
8	Last_name_E							X																						
9	First_name_E							1	X																					
10	Phone_E							1		X																				
11	Promail_E							1			X																			
12	Title_E							1				X																		
13	Hiredate_E							1					X																	
14	Num_way													X																
15	Num_street													1	X															
16	Aditional_address													1		X														
17	Postal_code													1			X													
18	Stopover_nbr																	X												
19	Offer																	1	X											
20	Order_date																			X										
21	Departure																			1	X									1
22	Departure_SO																					X								1
23	Arrival																						X							1
24	Arrival_SO																							X						1
25	Payment_method																								X					
26	Pyament_date																								1	X				
27	Payment_amount																								1		X			
28	Transport_available																											X		
29	City																												X	
30	Transport_used																													X

4. THE CONCEPTUAL DATA MODEL

A Conceptual Data Model is a diagram identifying the business concepts (entities) and the relationships between these concepts in order to gain, reflect, and document understanding of the organization's business, from a data perspective.
The use of the software « Jmerise » gave us the following model :



Our MCD was made this way, cause it seemed the most logical for us. Let's begin, with the "Client" entity, it and the entity "Employee" share "Address", cause everyone has an address. The "Address" is located is located in a "City". The "Client" commands a "Trip". He pays the "Trip" with "Installments". His "Trip" can be composed "Stopovers" or no, his choice, when going and returning he will use a "Transport", and this "Transport" will change depending on the availability of the latter in the "City" the "Client" will be going too. There is also the fact that "Employees" will manage "Clients" and that we need to calculate the "Distance" between each "City". The MLD and MPD will be generated with the help of jmerise.

5. THE LOGICAL DATA MODEL

The Logical Data Model is used to define the structure of data elements and to set relationships between them. The logical data model adds further information to the conceptual data model elements. The advantage of using a Logical data model is to provide a foundation to form the base for the Physical model. However, the modeling structure remains generic.

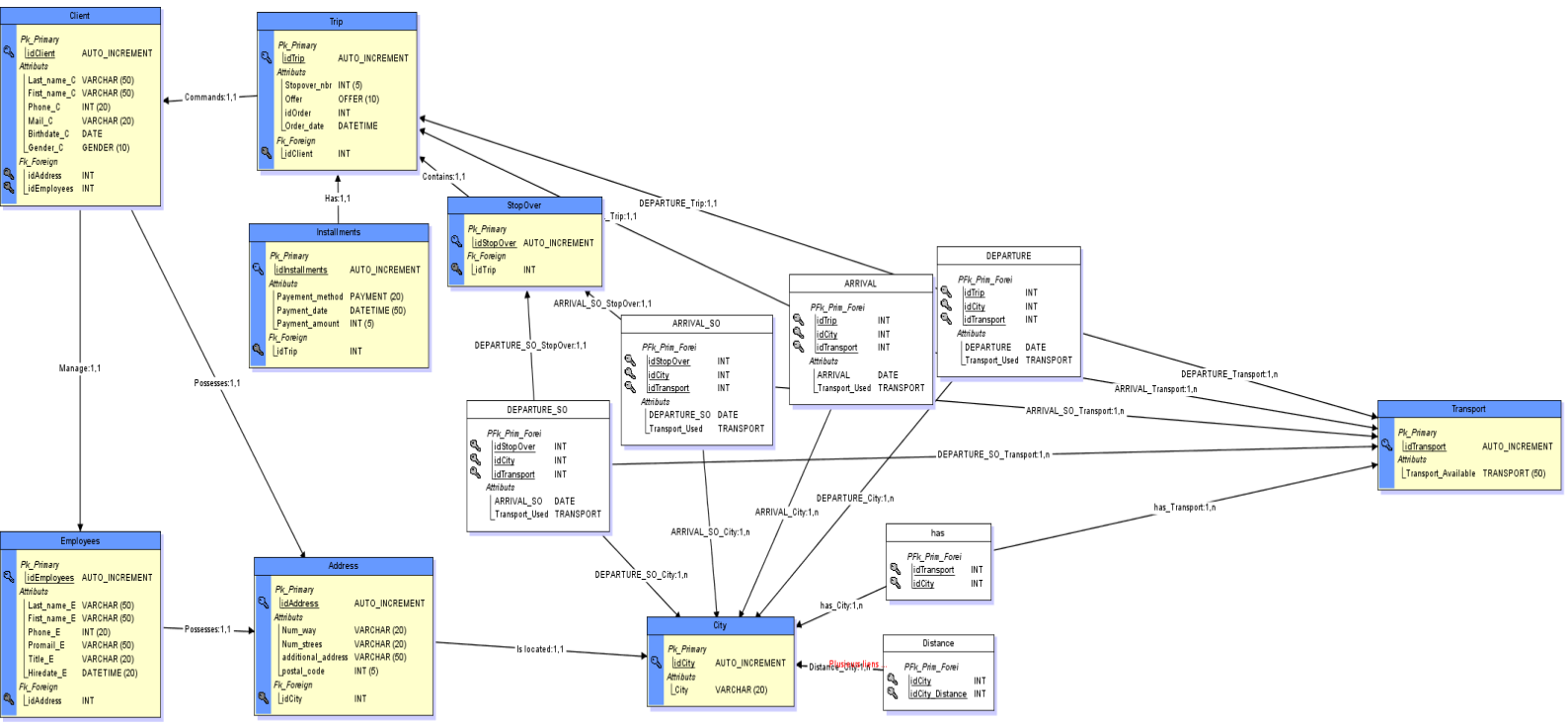
Here is ours:

In red the Primary keys, and in green the Foreign keys, this doesn't follow the rules of MLD, but it was done to clarify it for a better reading.

Client (idClient, Last_name_C, First_name_C, Phone_C, Mail_C, Birthdate_C, Gender_C, #idAddress, #idEmployees)
Employees (idEmployees, Last_name_E, First_name_E, Phone_E, Promail_E, Title_E, Hiredate_E, #idAddress)
Address (idAddress, Num_way, Num_streets, Additional_address, Postal_code, #idCity)
Trip (idTrip, Stopover_nbr, Offer, idOrder, Order_date, #idClient)
DEPARTURE (idTrip, idCity, idTransport, Transport_used, Departure)
ARRIVAL (idTrip, idCity, idTransport, Transport_used, Arrival)
StopOver (idStopOver,#idTrip)
DEPARTURE_SO (idStopOver, idCity, idTransport, Transport_used, Departure_so)
ARRIVAL_SO (idStopOver, idCity, idTransport, Transport_used, Arrival_so)
Installments (idInstallments, Payment_method, Payment_date, Payment_amont, #idTrip)
City (idCity, City)
Distance (idCity, idCity_Distance)
Has (idTransport, idCity)
Transport (idTransport , Transport_Available)

6. THE PHYSICAL DATA MODEL

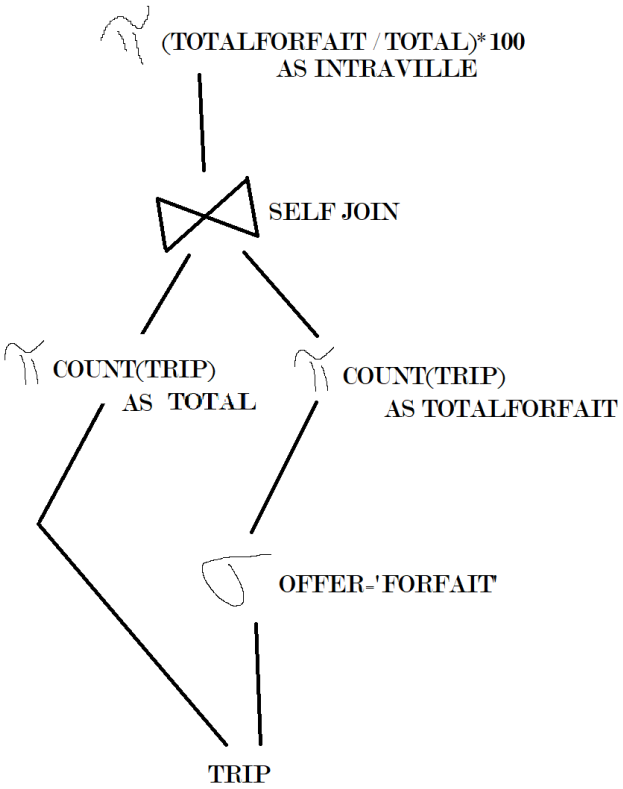
A Physical Data Model describes a database-specific implementation of the data model. It offers database abstraction and helps generate the schema. This is because of the richness of meta-data offered by a Physical Data Model. Ours is just here:

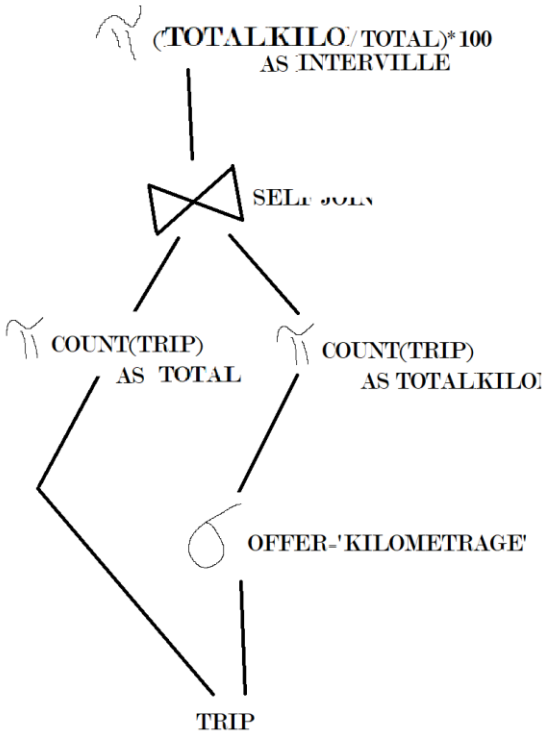


7. ALGEBRAIC TREES

First query:

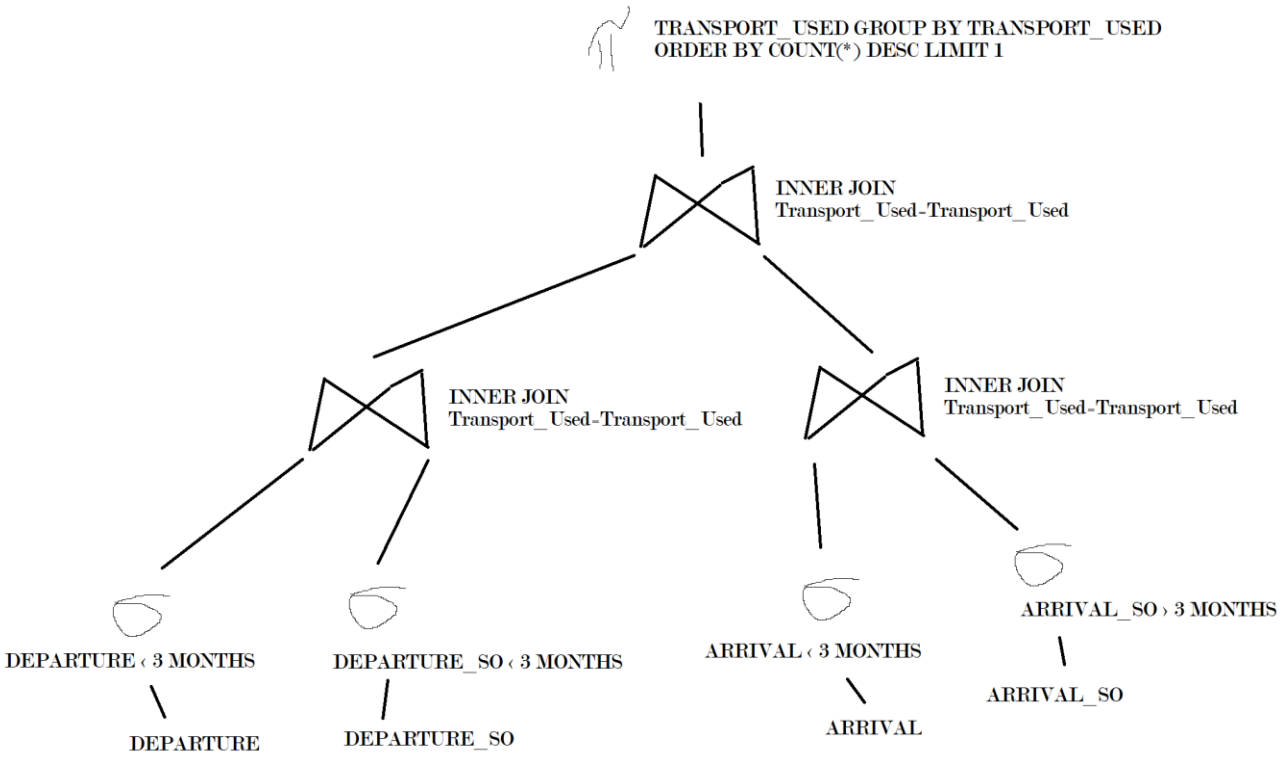
This query needs a SELF JOIN, a SELECTION and a PROJECTION, we will need to calculate the proportion manually in the PROJECTION. And we are using COUNT() to calculate how many lines there are in the SELECTION, and by that we mean how many trips.



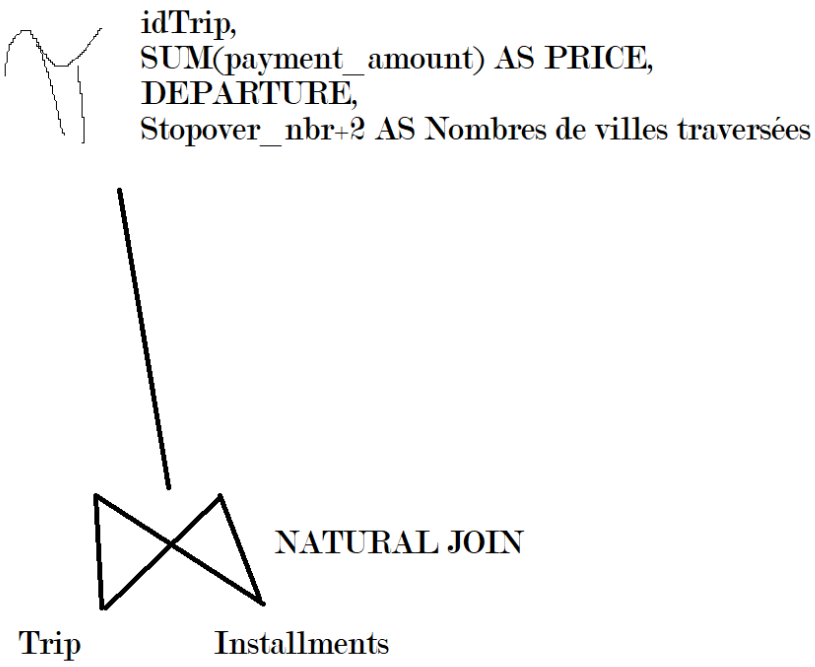


Second query:

The second query requires many INNER JOINS using transport_used, but before we use SELECTIONS to only use the data of the last 3 months, then we use the GROUP BY, ORDER BY and COUNT() feature in the PROJECTION and a LIMIT to find the most used one.

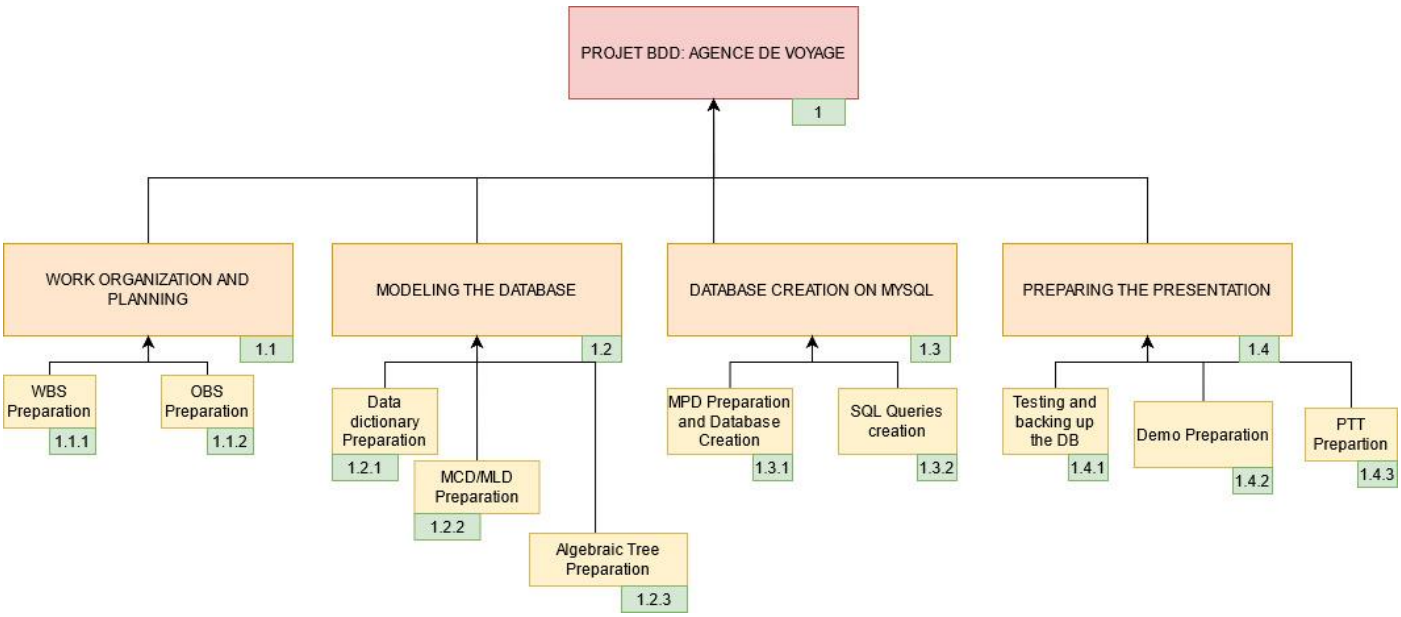


Third query:



The third query only requires a simple projection and a natural join.

8. WBS (from deliverable 0)



9. OBS (from deliverable 0)