



LIVRABLE 1

É Q U I P E 0 3 :

M A R T I N L É O
P A S Q U E T E N Z O
R A K O T O N I R I N A T O N Y
S U C H E T A N T O I N E

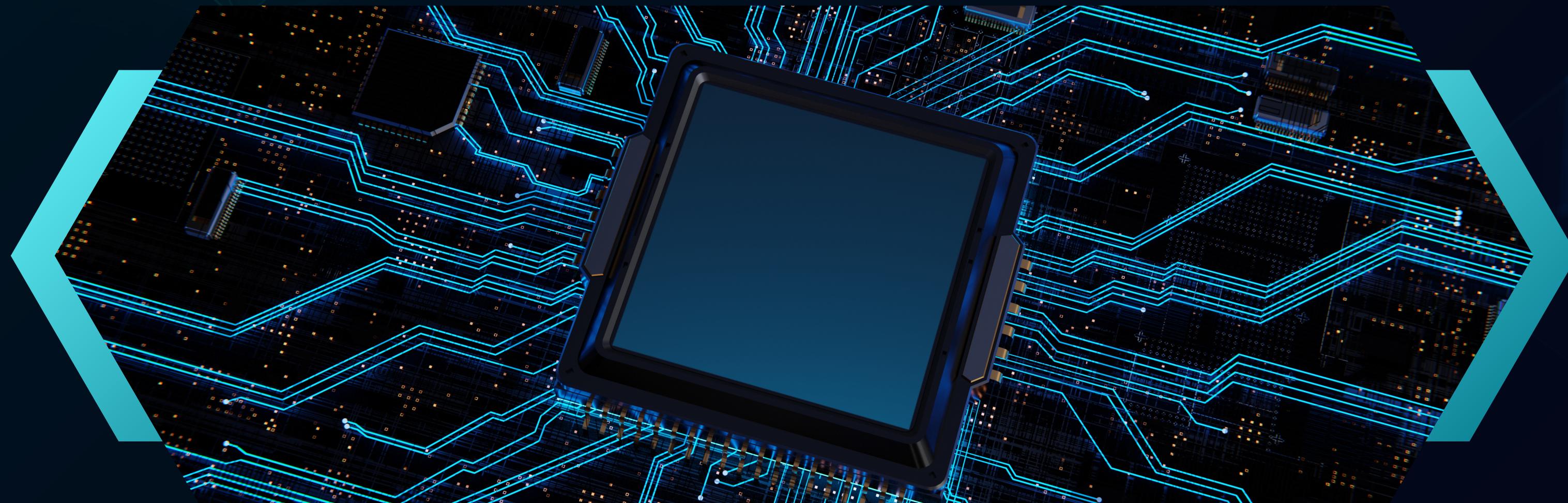
TABLE DES MATIÈRES

- 03 Introduction
- 04 Dictionnaire des données
- 06 Modèle conceptuel des données
- 08 Modèle logique des données
- 12 Modèle physique des données
- 14 Arbres algébriques
- 25 Conclusion



INTRODUCTION

Tout d'abord pour rappel dans notre dernier rapport nous avions établie le début du projet en définissant le besoin de notre client et ses enjeux par différentes méthodes d'organisations. Pour la suite de ce projet, nous allons maintenant commencer à définir les éléments et les liaisons entre elles pour la création de notre futur base de données.



DICTIONNAIRE DES DONNÉES

Définition :

Un dictionnaire de données sert à inventorier les informations d'un système. Il sert lors d'un projet à dresser la liste de toutes les informations utiles pour modéliser un système.

Nom	Type	Taille	Minimum	Maximum	Commentaires
Id_age	INT	4	0	4294967296	Identifiant agence
Nom_age	C	40			Nom de l'agence
Adresse_age	C	40			Adresse de l'agence
Id_per	INT	4	0	4294967296	Identifiant personnel
Nom_per	C	40			Nom personnel
Prénom_per	C	40			Prénom personnel
Date_nai_per	DATE	10	01/01/0001	31/12/9999	Date naissance personnel
Date_job	DATE	10	01/01/0001	31/12/9999	Date prise de poste
Type_per	C	40			Type personnel
Taux_pro	DEC	4			Taux de productivité
Id_gaz	INT	4	0	4294967296	Identifiant gaz
Nom_gaz	C	40			Nom du gaz
Type_gaz	C	40			Type de gaz
Id_cap	INT	4	0	4294967296	Identifiant capteur
Capteur_dep	BOOL	2			Capteur_déployé
Id_sec_act	Int	4	0	4294967296	Identifiant secteur d'activité
Secteur_act	C	40			Secteur d'activité
Id_reg	Int	4	0	4294967296	Identifiant région
Région	C	40			Région
Id_vil	C	40			Identifiant ville
Ville	C	40			Ville
Code_pos	INT	4	0	4294967296	Code postal
Id_rel	INT	4	0	4294967296	Identifiant donnée
Valeur_ppm	DEC	4			Valeur ppm
Date_rel	DATE	10	01/01/0001	31/12/9999	Date de relevé
Id_rap	INT	4	0	4294967296	Identifiant rapport
Titre_rap	C	40			Titre du rapport
Date_par	DATE	10	01/01/0001	31/12/9999	Date de parrution

DICTIONNAIRE DES DONNÉES

Nous avons décidé de catégoriser notre dictionnaire de données en neuf grands groupes de données représentés ci-contre par un code couleur. Ceci nous permet d'identifier plus facilement les groupes de données avec des similitudes.

Chaque colonne représente un détail de la donnée, tout d'abord son “nom” qui va se retrouver partout par la suite, le “type” de la donnée, la “taille” en octet de la donnée puis sa portée “minimale” et “maximale”. Pour finir, la colonne explicative “commentaire” qui permet de clarifier le nom complet de la donnée. Nous avons fait ce choix pour éviter d'alourdir nos prochains schémas avec des noms de données qui peuvent être très long et ainsi gagner du temps d'écriture et de lecture.

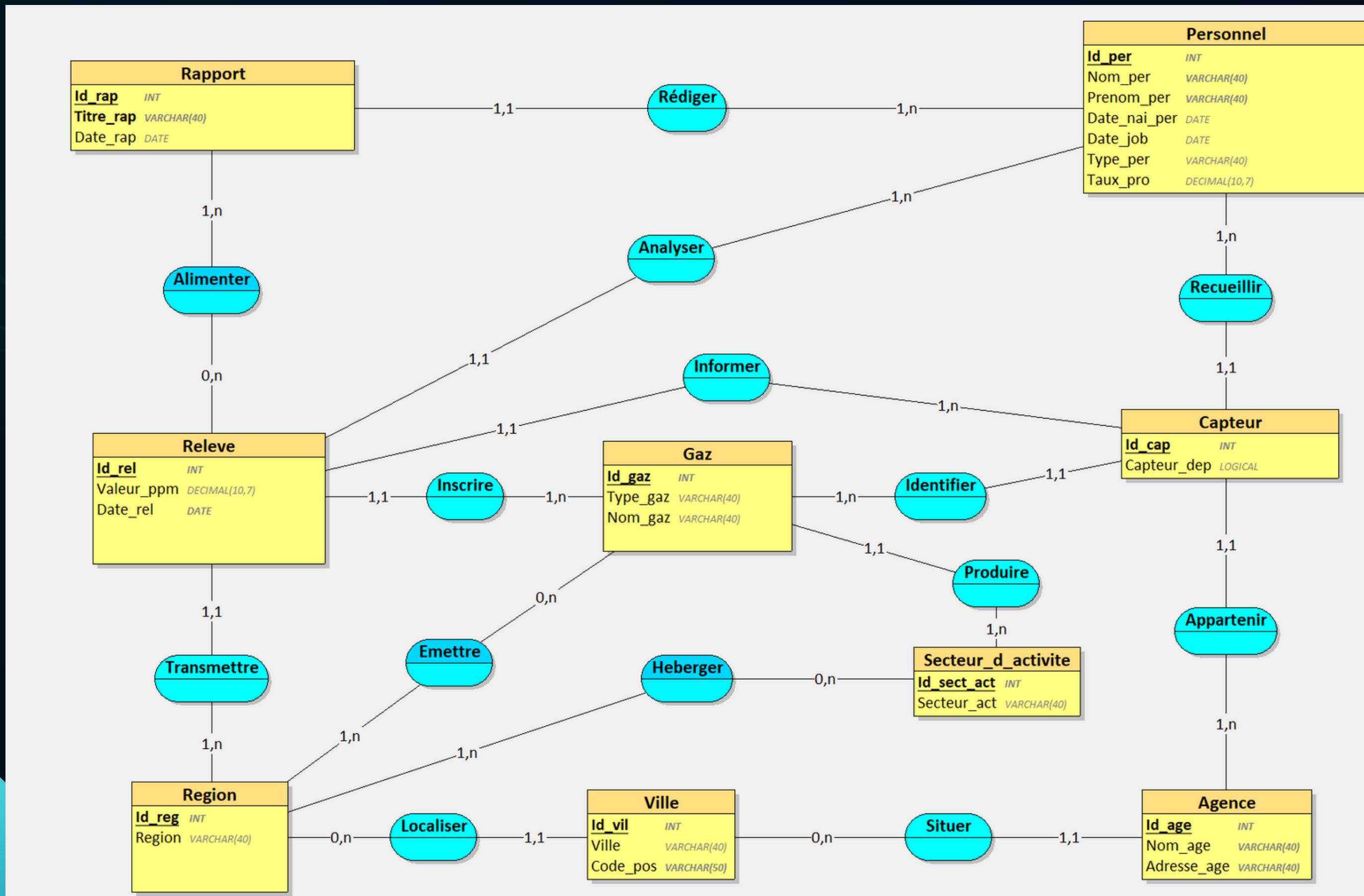
MODÈLE CONCEPTUEL DES DONNÉES

Nous avons élaboré le Modèle Conceptuel de Données (MCD) en utilisant comme point de départ le dictionnaire des données fait précédemment. Nous avons systématiquement structuré ces données en tables distinctes, telles que "Région", "Personnel" et "Agence". Chaque table a été dotée de clés primaires, créant ainsi des identifiants uniques pour chaque enregistrement et permettant leur interconnexion.

Ensuite, nous avons établi les associations entre ces tables en utilisant des verbes à l'infinitif qui reflètent la nature des relations. Par exemple, le verbe "rédiger" a été employé pour l'association entre la table "Rapport" et la table "Personnel".

Par la suite, nous avons identifié les relations entre les entités. Ces relations explicitent la manière dont les entités interagissent les unes avec les autres. Elles peuvent revêtir différentes formes, telles que "un-à-un", "un-à-plusieurs" ou "plusieurs-à-plusieurs". Par exemple, une relation "un-à-plusieurs" a été établie entre les entités "Personnel" et "Rapports", indiquant qu'un membre du personnel peut rédiger plusieurs rapports.

MODÈLE CONCEPTUEL DES DONNÉES



MODÈLE LOGIQUE DES DONNÉES

Pour passer du MCD au MLD, nous avons suivi les règles suivante :

- Une entité du MCD devient une relation, c'est à dire une table

(Chaque entité du MCD est convertie en une table distincte dans le MLD. Par exemple, les entités telles que "Région", "Personnel" et "Agence" deviennent des relations dans le MLD.)

- Son identifiant devient la clé primaire de la relation

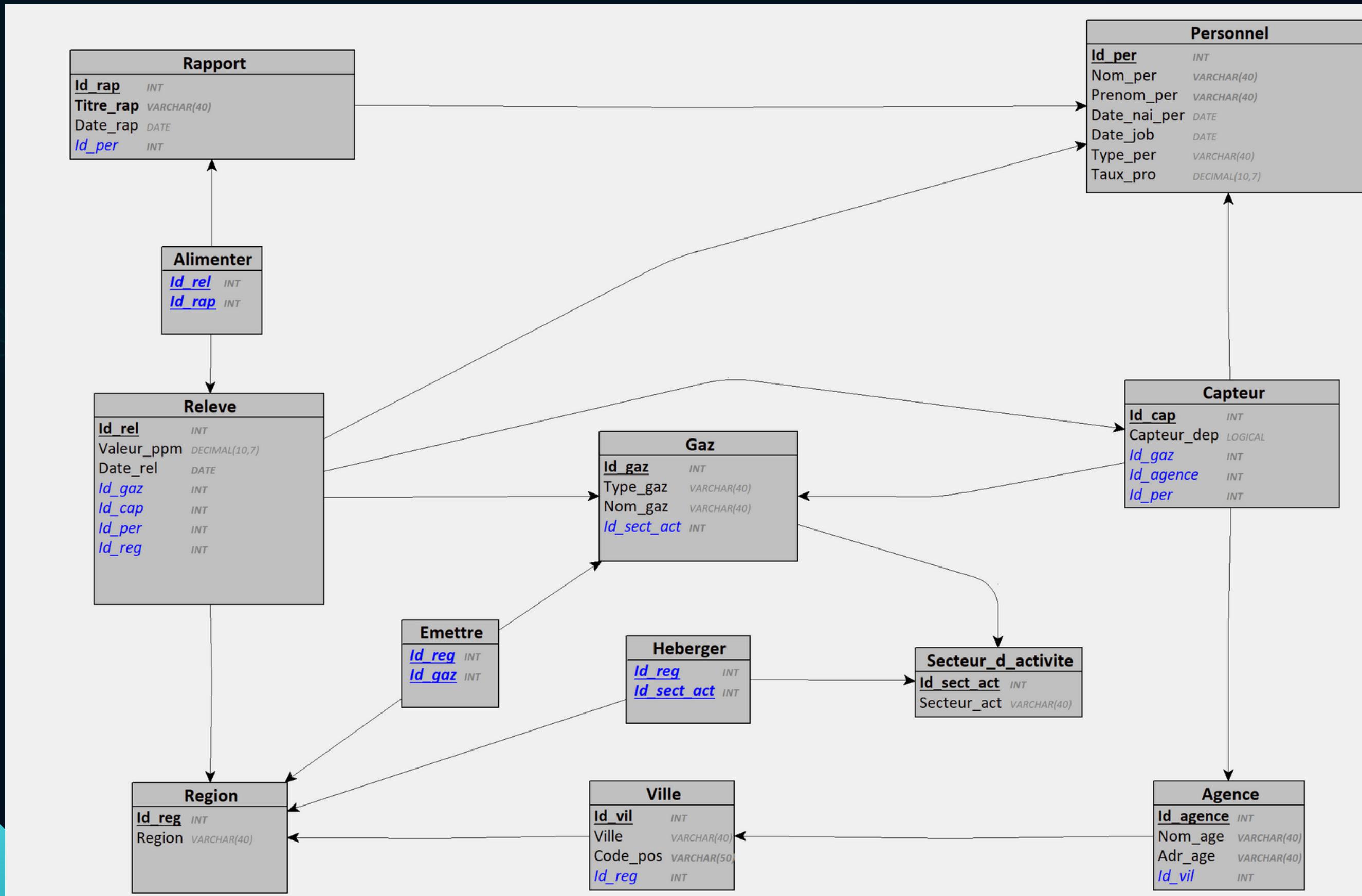
(L'identifiant de chaque entité dans le MCD devient la clé primaire de la relation correspondante dans le MLD. Cette clé primaire garantit l'unicité des enregistrements dans la table. Les autres propriétés deviennent les attributs de la relation).

MODÈLE LOGIQUE DES DONNÉES

Attributs des relations :

- (Les autres propriétés de chaque entité du MCD deviennent les attributs des relations dans le MLD. Par exemple, les attributs spécifiques comme le nom d'une région ou les informations personnelles d'un employé sont représentés en tant qu'attributs dans les tables correspondantes.).
- Une association de type 1:N (c'est à dire qui a les cardinalités maximales positionnées à « 1 » d'une côté de l'association et à « n » de l'autre côté) se traduit par la création d'une clé étrangère dans la relation correspondante à l'entité côté « 1 ». Cette clé étrangère référence la clé primaire de la relation correspondant à l'autre entité.

MODÈLE LOGIQUE DES DONNÉES



MODÈLE LOGIQUE DES DONNÉES

Modèle logique des données version textuel

```
Personnel = (Id_per INT, Nom_per VARCHAR(40), Prenom_per VARCHAR(40), Date_nai_per DATE, Date_job DATE, Type_per VARCHAR(40), Taux_pro DECIMAL(10,7));  
Region = (Id_reg INT, Region VARCHAR(40));  
Rapport = (Id_rap INT, Titre_rap VARCHAR(40), Date_rap DATE, #Id_per);  
Secteur_d_activite = (Id_sect_act INT, Secteur_act VARCHAR(40));  
Ville = (Id_vil INT, Ville VARCHAR(40), Code_pos VARCHAR(50), #Id_reg);  
Gaz = (Id_gaz INT, Type_gaz VARCHAR(40), Nom_gaz VARCHAR(40), #Id_sect_act);  
Agence = (Id_agence INT, Nom_age VARCHAR(40), Adr_age VARCHAR(40), #Id_vil);  
Capteur = (Id_cap INT, Capteur_dep LOGICAL, #Id_gaz, #Id_agence, #Id_per);  
Releve = (Id_rel INT, Valeur_ppm DECIMAL(10,7), Date_rel DATE, #Id_gaz, #Id_cap, #Id_per, #Id_reg);  
Alimenter = (#Id_rel, #Id_rap);  
Emettre = (#Id_reg, #Id_gaz);  
Heberger = (#Id_reg, #Id_sect_act);
```

MODÈLE PHYSIQUE DES DONNÉES

Définition :

Le Modèle Physique des Données (MPD) transforme le Modèle Logique des Données (MLD) en une structure concrète et détaillée prête à être implémentée dans un système de gestion de base de données (SGBD). Il décrit précisément la composition de la base de données, y compris la création des tables, des colonnes avec leurs types de données respectifs, ainsi que les contraintes telles que les clés primaires et étrangères.

MODÈLE PHYSIQUE DES DONNÉES

```
CREATE TABLE Personnel(
    Id_per INT,
    Nom_per VARCHAR(40) NOT NULL,
    Prenom_per VARCHAR(40) NOT NULL,
    Date_nai_per DATE,
    Date_job DATE,
    Type_per VARCHAR(40),
    Taux_pro DECIMAL(10,7),
    PRIMARY KEY(Id_per)
);

CREATE TABLE Region(
    Id_reg INT,
    Region VARCHAR(40),
    PRIMARY KEY(Id_reg)
);

CREATE TABLE Rapport(
    Id_rap INT,
    Titre_rap VARCHAR(40) NOT NULL,
    Date_rap DATE,
    Id_per INT NOT NULL,
    PRIMARY KEY(Id_rap),
    UNIQUE(Titre_rap),
    FOREIGN KEY(Id_per) REFERENCES Personnel(Id_per)
);

CREATE TABLE Secteur_d_activite(
    Id_sect_act INT,
    Secteur_act VARCHAR(40),
    PRIMARY KEY(Id_sect_act)
);
```

```
CREATE TABLE Ville(
    Id_vil INT,
    Ville VARCHAR(40),
    Code_pos VARCHAR(50) NOT NULL,
    Id_reg INT NOT NULL,
    PRIMARY KEY(Id_vil),
    FOREIGN KEY(Id_reg) REFERENCES Region(Id_reg)
);

CREATE TABLE Gaz(
    Id_gaz INT,
    Type_gaz VARCHAR(40),
    Nom_gaz VARCHAR(40),
    Id_sect_act INT NOT NULL,
    PRIMARY KEY(Id_gaz),
    FOREIGN KEY(Id_sect_act) REFERENCES Secteur_d_activite(Id_sect_act)
);

CREATE TABLE Agence(
    Id_agence INT,
    Nom_agé VARCHAR(40) NOT NULL,
    Adr_agé VARCHAR(40) NOT NULL,
    Id_vil INT NOT NULL,
    PRIMARY KEY(Id_agence),
    FOREIGN KEY(Id_vil) REFERENCES Ville(Id_vil)
);

CREATE TABLE Capteur(
    Id_cap INT,
    Capteur_dep LOGICAL,
    Id_gaz INT NOT NULL,
    Id_agence INT NOT NULL,
    id_personnel INT NOT NULL,
    PRIMARY KEY(Id_cap),
    FOREIGN KEY(Id_gaz) REFERENCES Gaz(Id_gaz),
    FOREIGN KEY(Id_agence) REFERENCES Agence(Id_agence),
    FOREIGN KEY(id_personnel) REFERENCES Personnel(id_personnel)
);
```

```
CREATE TABLE Releve(
    Id_rel INT,
    Valeur_ppm DECIMAL(10,7),
    Date_rel DATE NOT NULL,
    Id_gaz INT NOT NULL,
    Id_cap INT NOT NULL,
    id_personnel INT NOT NULL,
    Id_reg INT NOT NULL,
    PRIMARY KEY(Id_rel),
    FOREIGN KEY(Id_gaz) REFERENCES Gaz(Id_gaz),
    FOREIGN KEY(Id_cap) REFERENCES Capteur(Id_cap),
    FOREIGN KEY(id_personnel) REFERENCES Personnel(id_personnel),
    FOREIGN KEY(Id_reg) REFERENCES Region(Id_reg)
);

CREATE TABLE Alimenter(
    Id_rel INT,
    Id_rap INT,
    PRIMARY KEY(Id_rel, Id_rap),
    FOREIGN KEY(Id_rel) REFERENCES Releve(Id_rel),
    FOREIGN KEY(Id_rap) REFERENCES Rapport(Id_rap)
);

CREATE TABLE Emettre(
    Id_reg INT,
    Id_gaz INT,
    PRIMARY KEY(Id_reg, Id_gaz),
    FOREIGN KEY(Id_reg) REFERENCES Region(Id_reg),
    FOREIGN KEY(Id_gaz) REFERENCES Gaz(Id_gaz)
);

CREATE TABLE Heberger(
    Id_reg INT,
    Id_sect_act INT,
    PRIMARY KEY(Id_reg, Id_sect_act),
    FOREIGN KEY(Id_reg) REFERENCES Region(Id_reg),
    FOREIGN KEY(Id_sect_act) REFERENCES Secteur_d_activite(Id_sect_act)
);
```

LES ARBRES ALGÉBRIQUES

Définition :

Dans le contexte des bases de données, un arbre algébrique est une représentation structurée et schématique utilisée pour visualiser et optimiser les requêtes SQL avant leur exécution sur une base de données. Cette représentation prend la forme d'un arbre où chaque noeud correspond à une opération algébrique qui sera appliquée aux données.

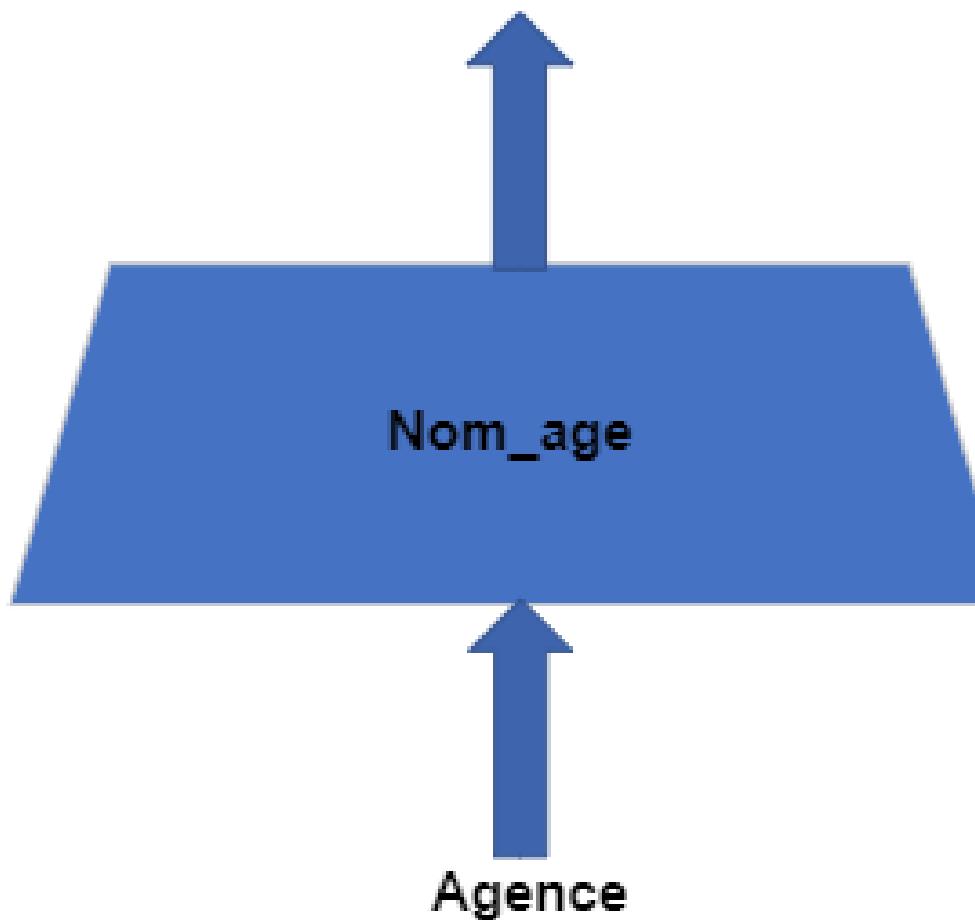
LES ARBRES ALGÉBRIQUES

Dans nos arbres algébriques, nous retrouvons plusieurs types d'opérations :

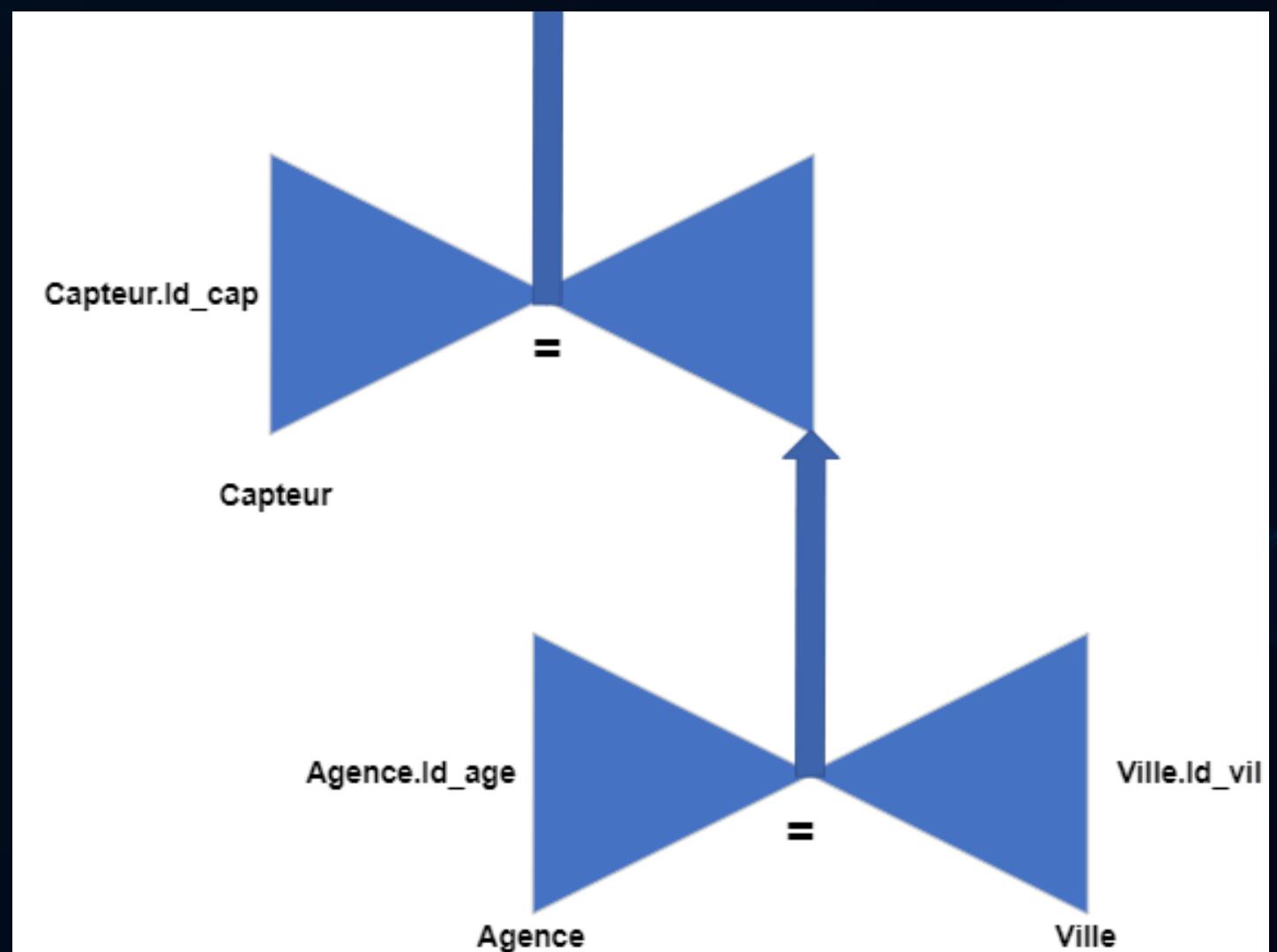
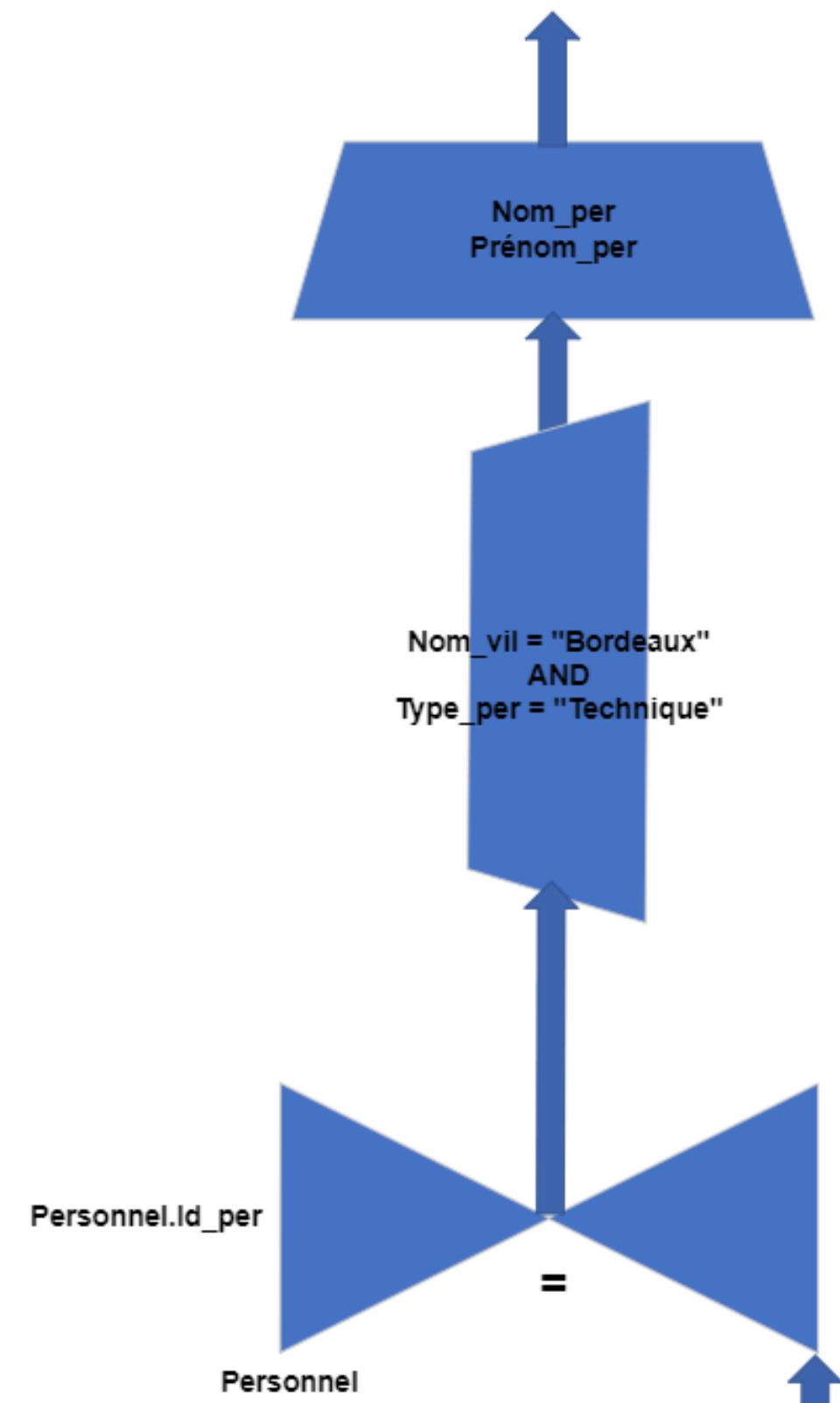
- Les **jointures**, représentées par des doubles triangles, combinent les colonnes provenant de deux tables différentes en les reliant à l'aide d'une condition de correspondance. Cela crée un ensemble de résultats combinés rassemblant ainsi des données cohérentes de plusieurs tables.
- Les **projections**, symbolisées par des trapèzes horizontales, sélectionnent et fournissent un ensemble spécifique de colonnes d'une table. Les résultats de la requête sont restreints aux données de cette sélection ce qui permet de traiter et de réduire les données pour une meilleure lisibilité.
- Les **sélections**, représentées par des trapèzes verticaux, filtrent les lignes d'une table en fonction de conditions spécifiques établies. Seules les données correspondant aux critères sont transmises, ce qui permet d'extraire des données précises d'un ensemble plus large. Ces requêtes sont plus efficaces car elles limitent les données utilisées.

LES ARBRES ALGÉBRIQUES

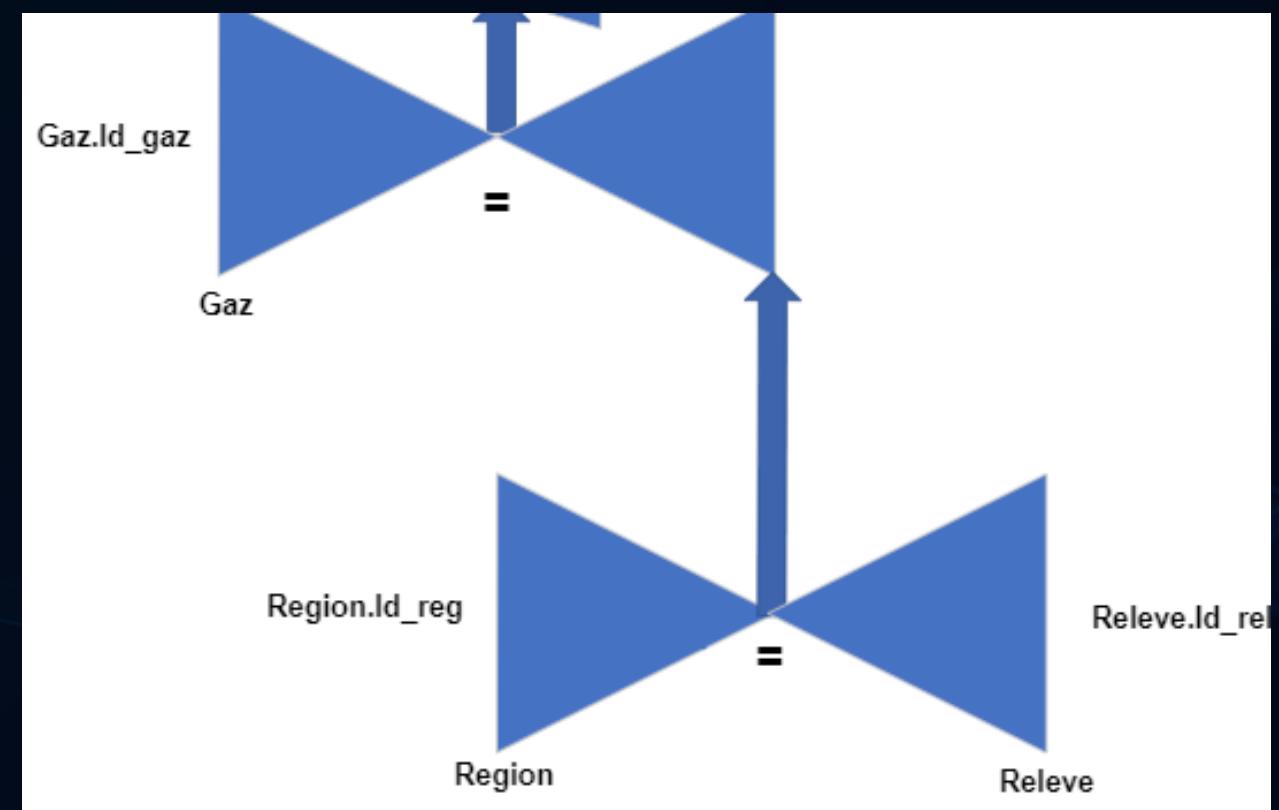
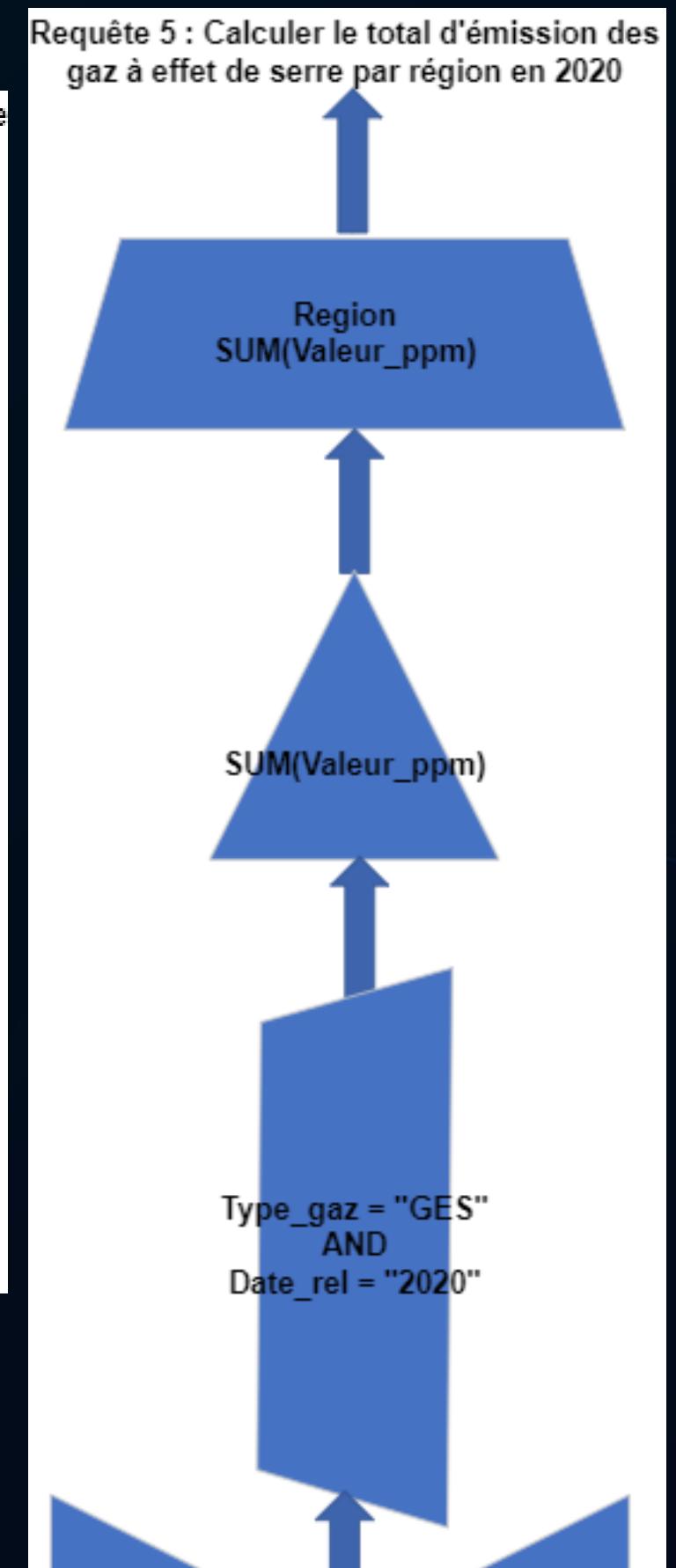
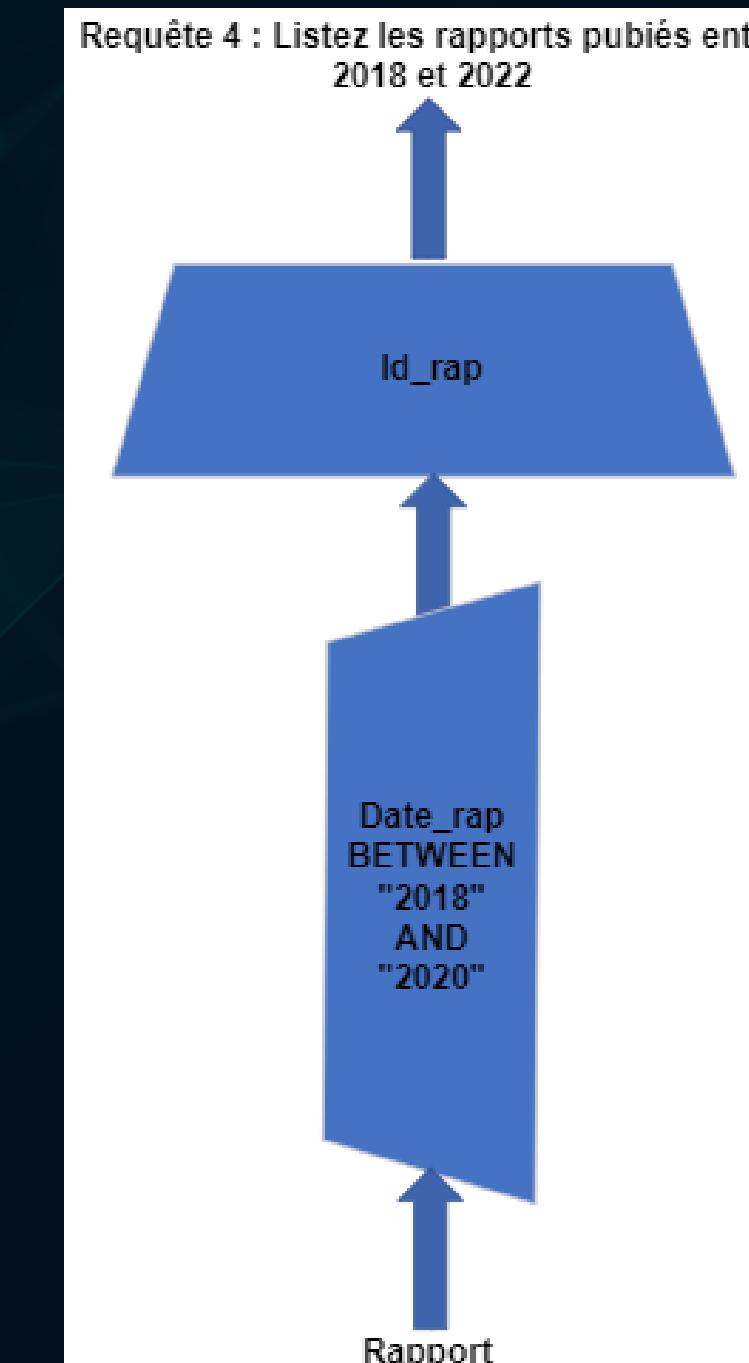
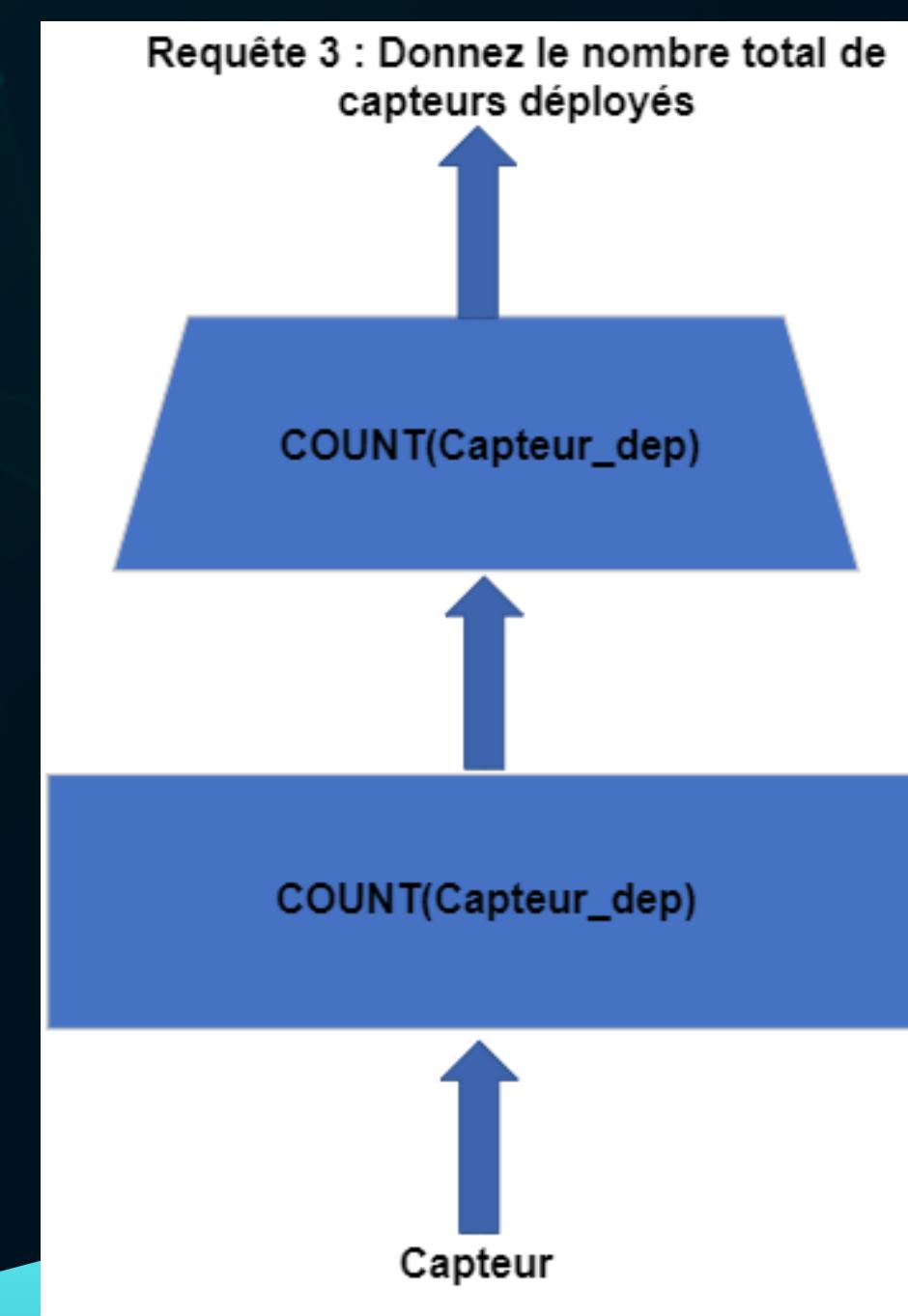
Requête 1 : Listez l'ensemble des agences



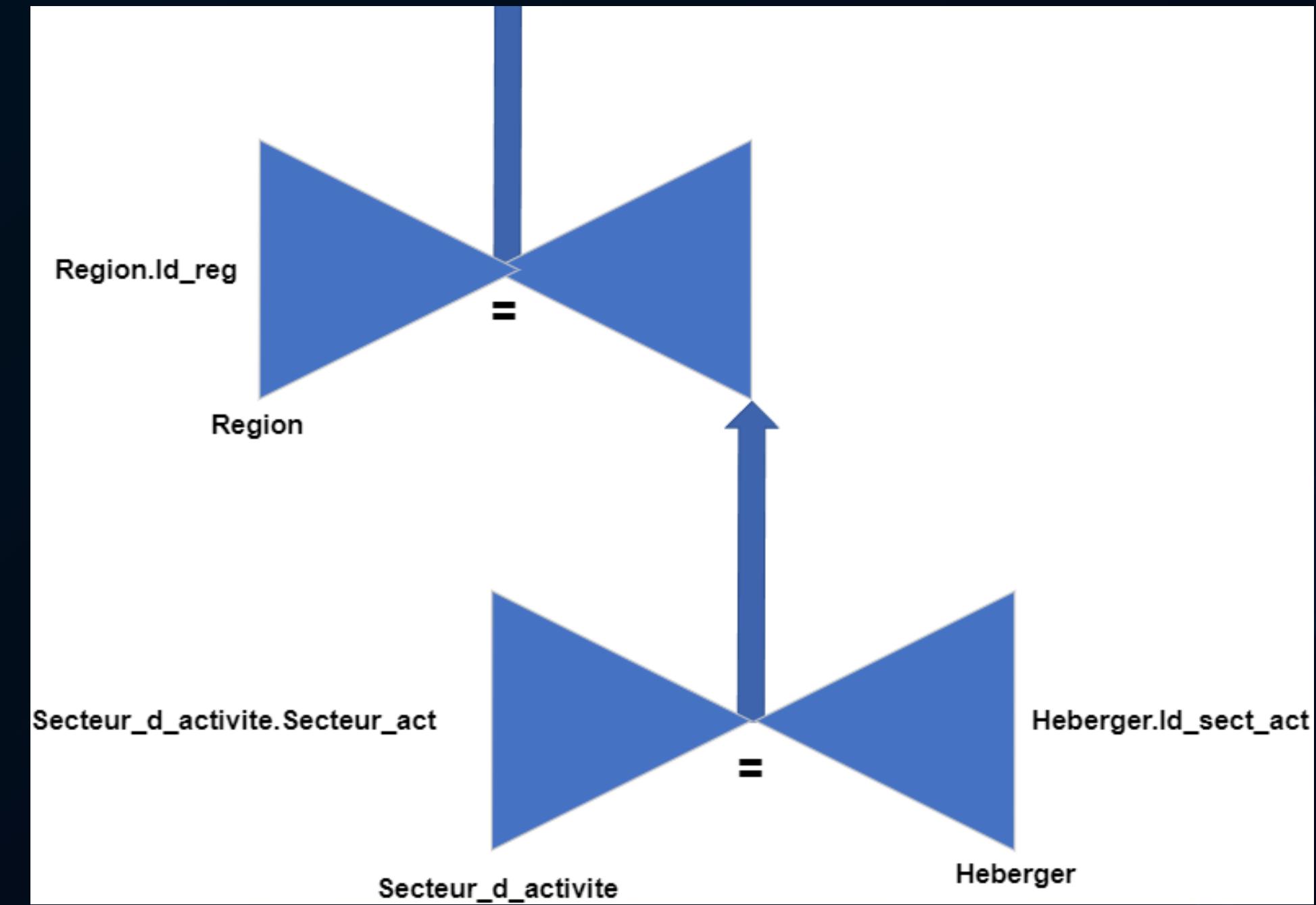
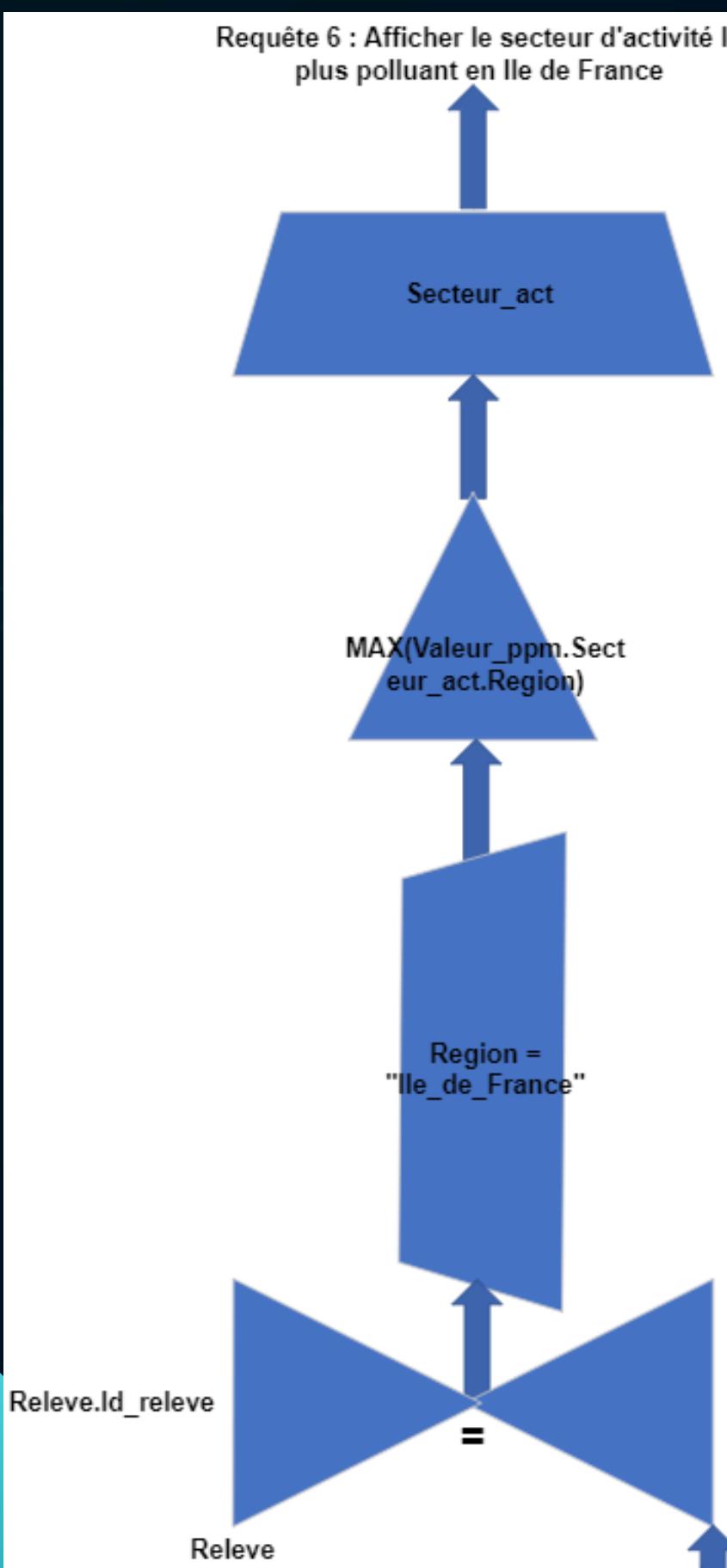
Requête 2 : Listez l'ensemble du personnel technique de l'agence de Bordeaux



LES ARBRES ALGÉBRIQUES



LES ARBRES ALGÉBRIQUES

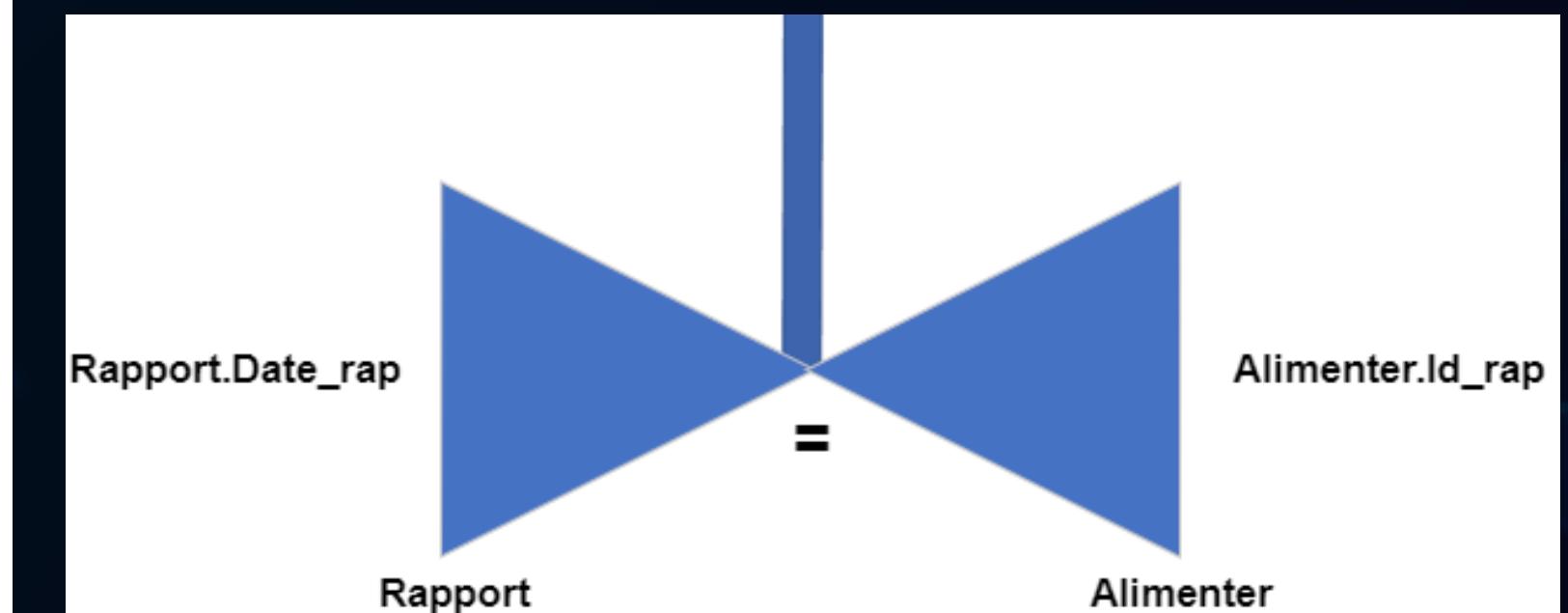
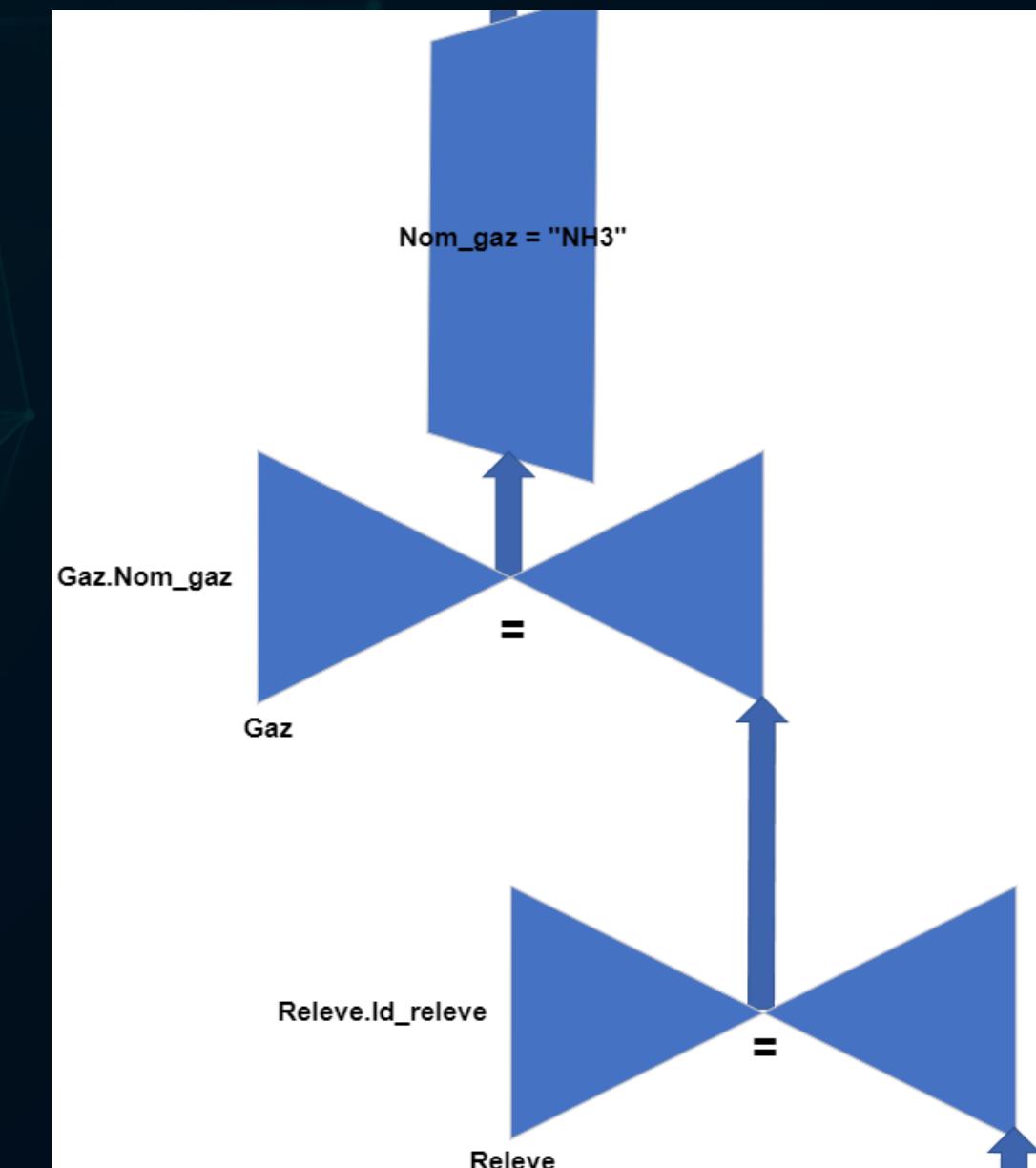


LES ARBRES ALGÉBRIQUES

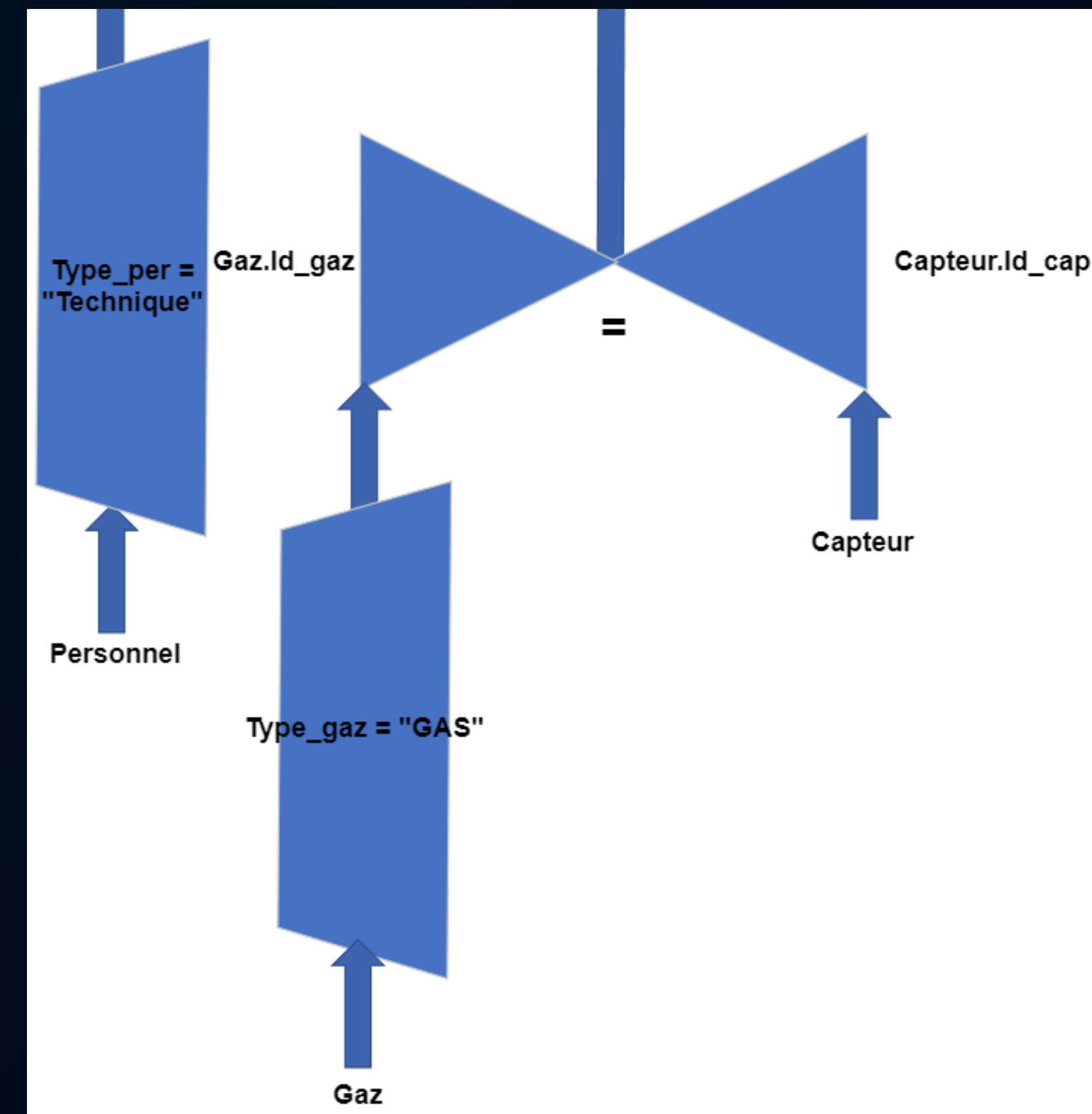
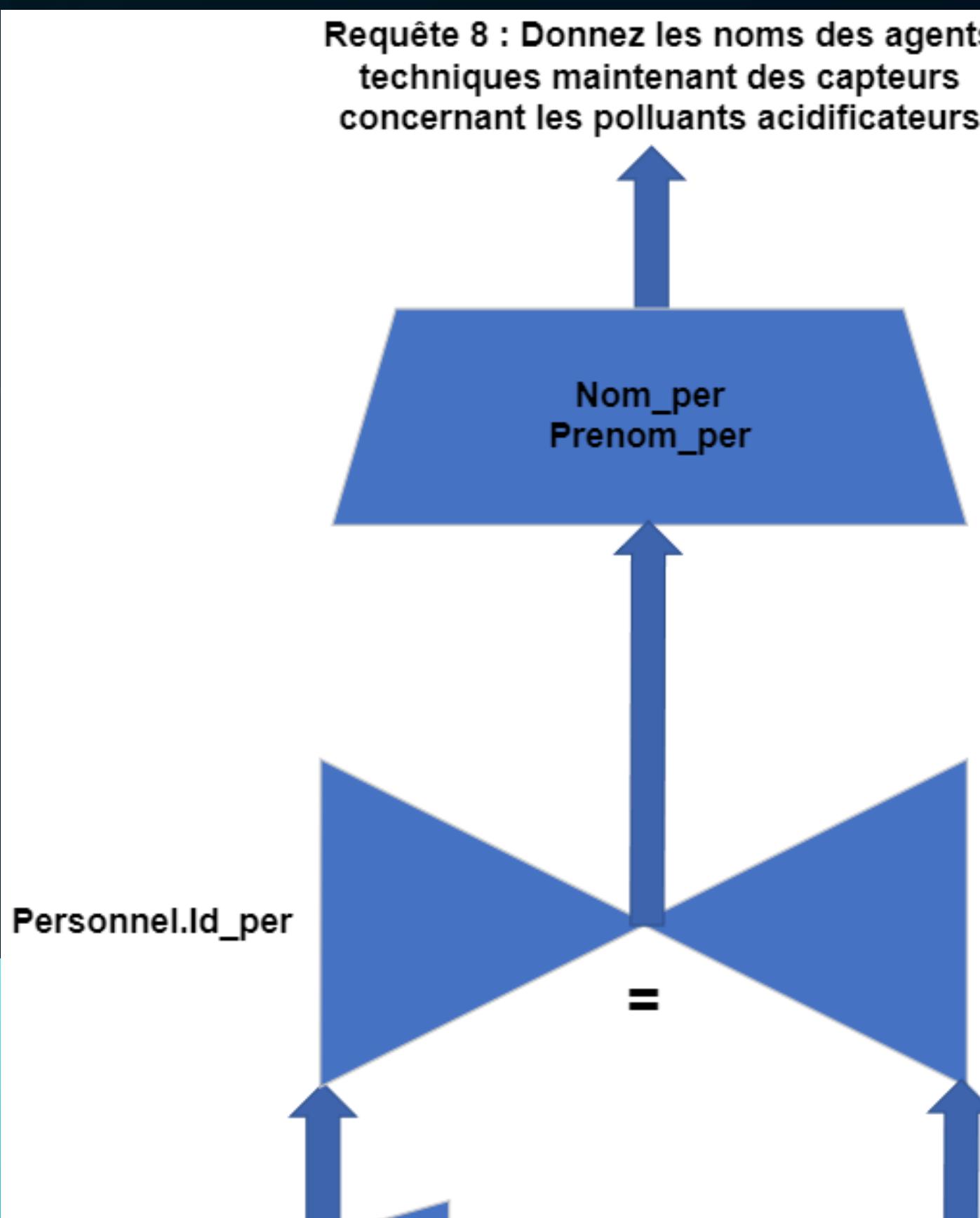
Requête 7 : Classez les rapports concernant des émissions de NH3 par ordre chronologique

`Id_rap
Nom_gaz
Date_rap`

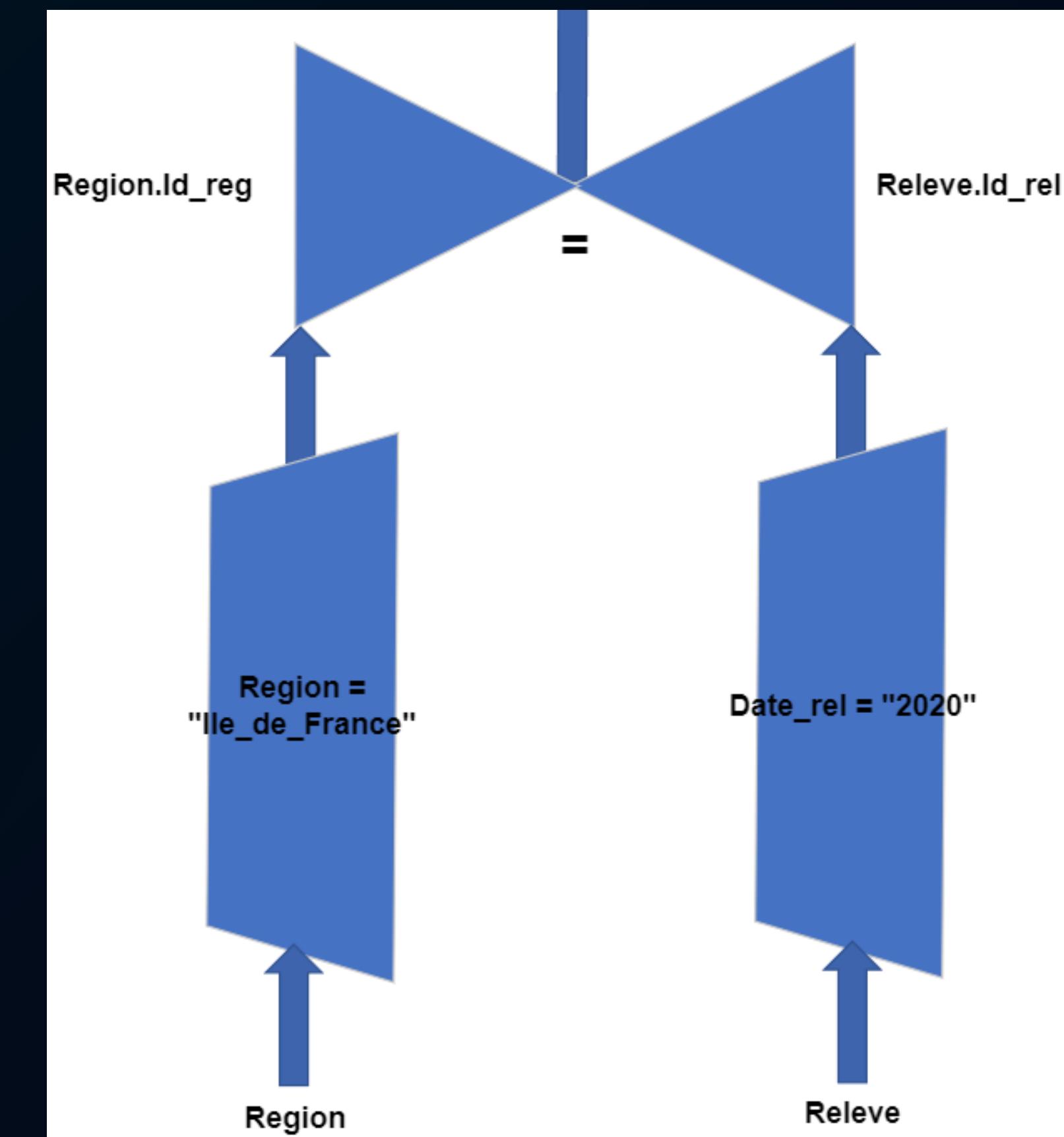
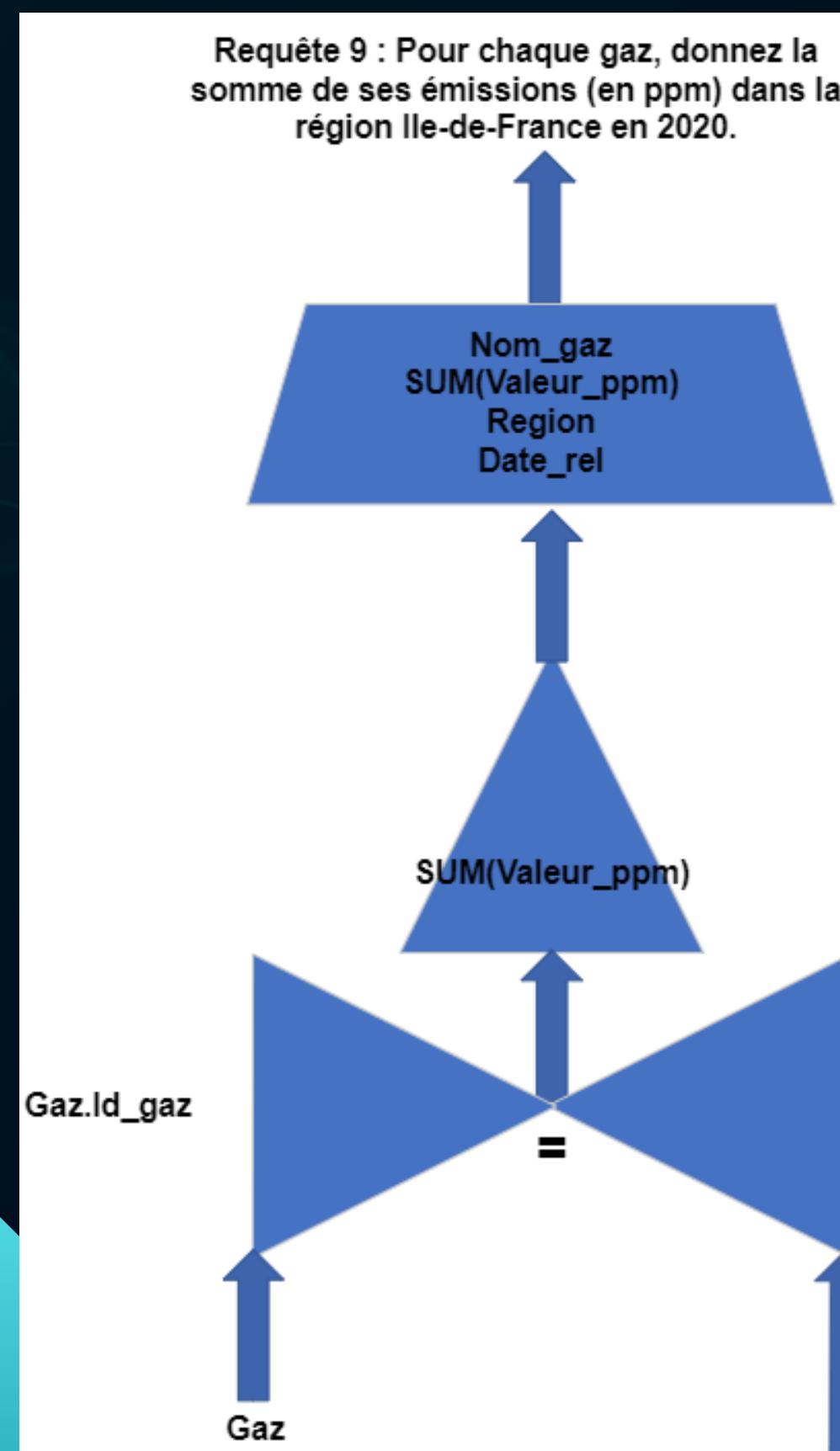
`ORDER BY ASC()`



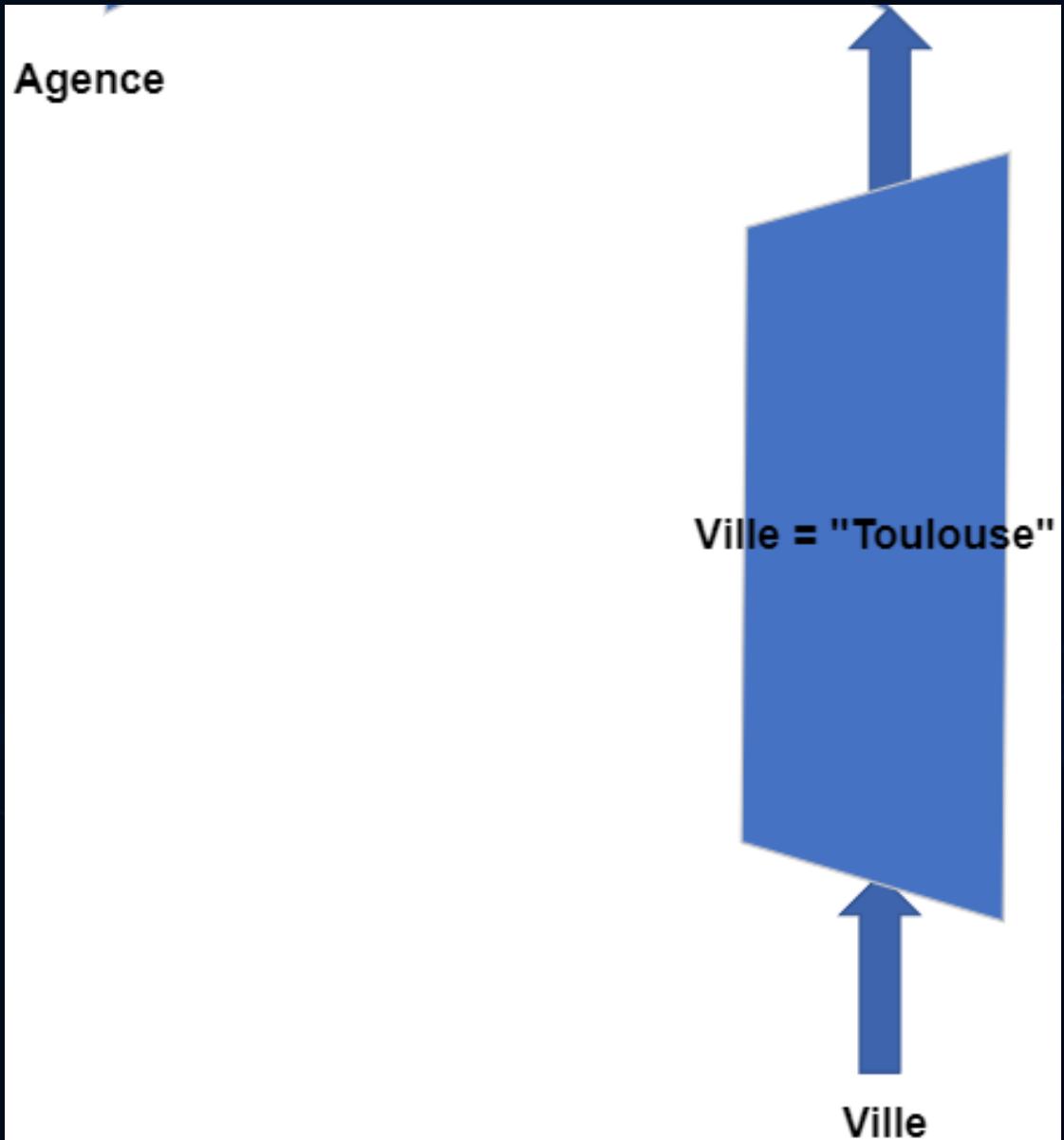
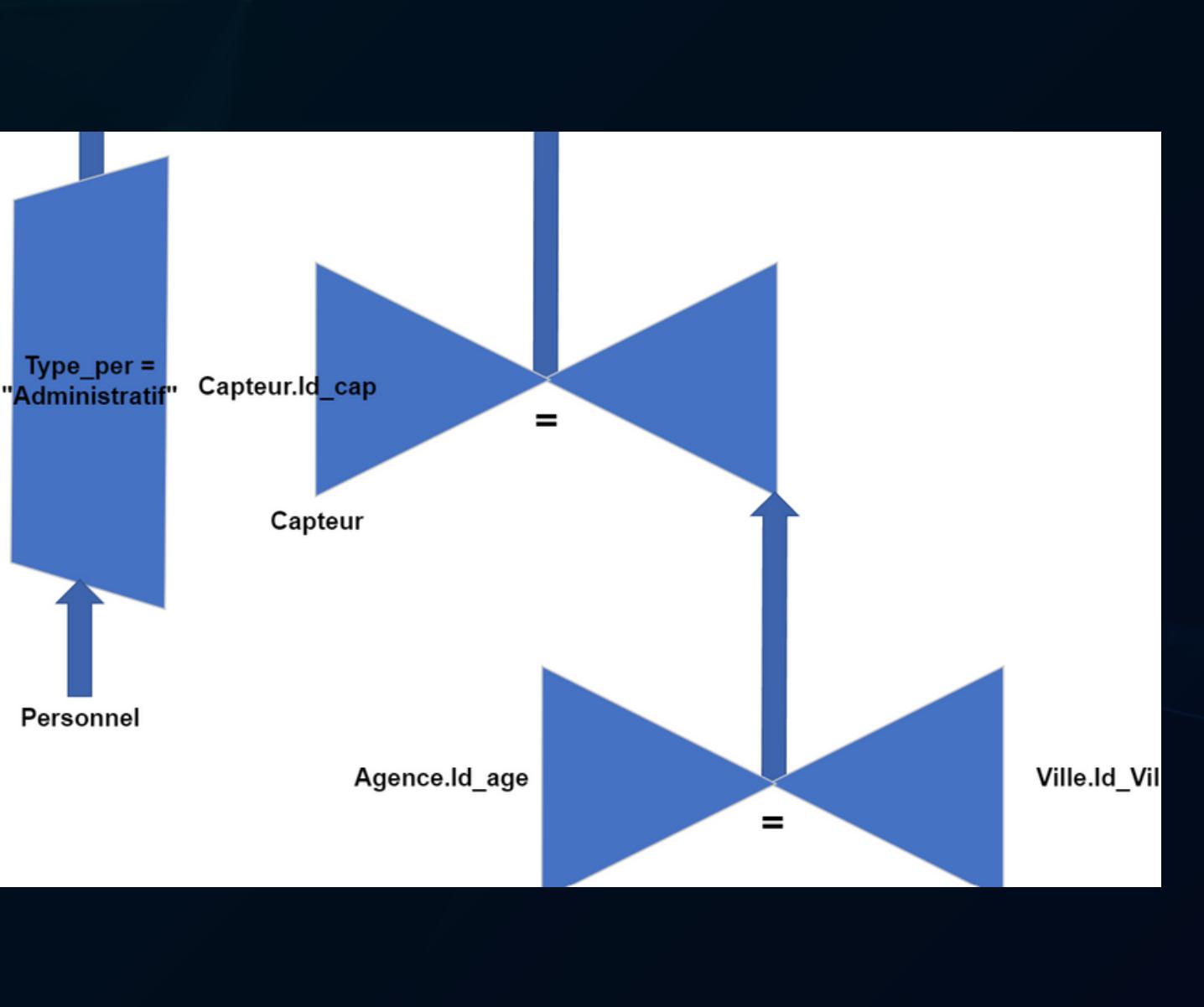
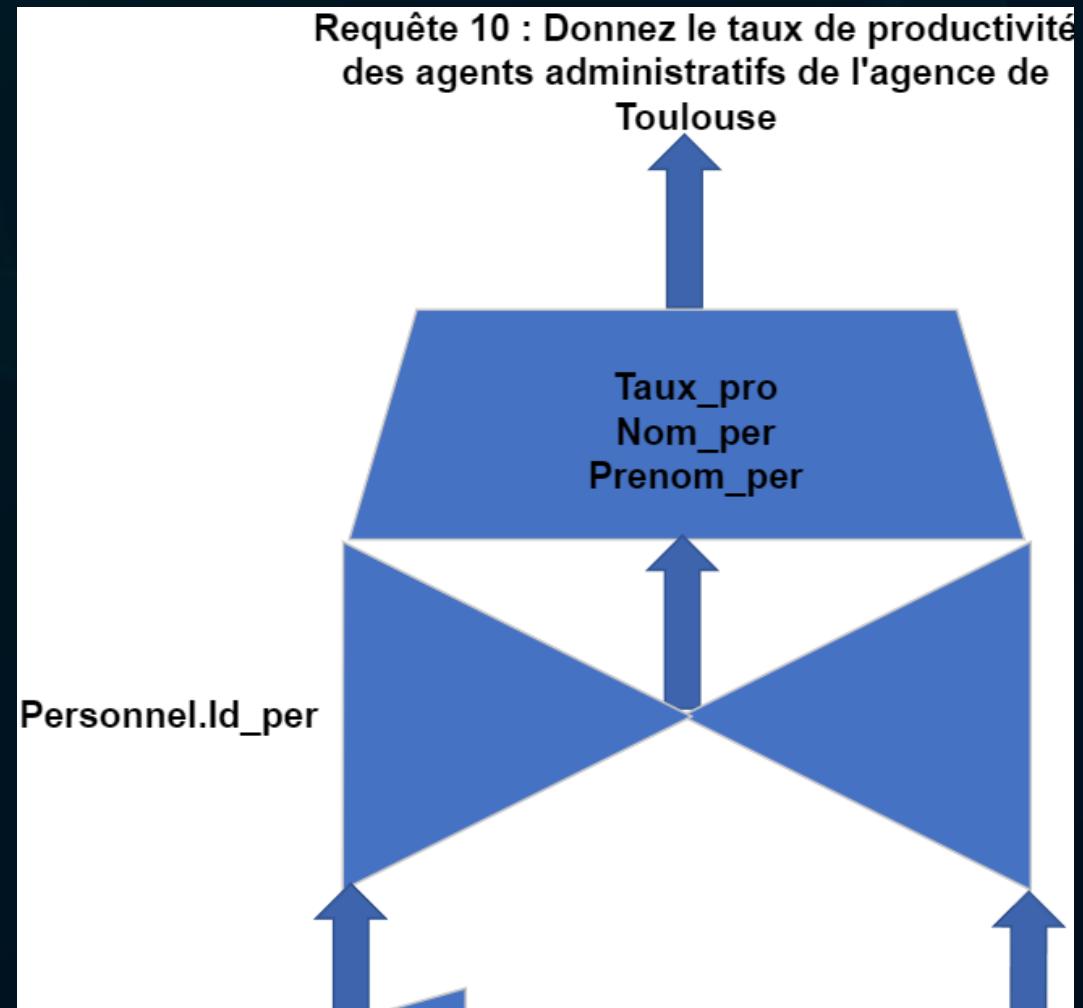
LES ARBRES ALGÉBRIQUES



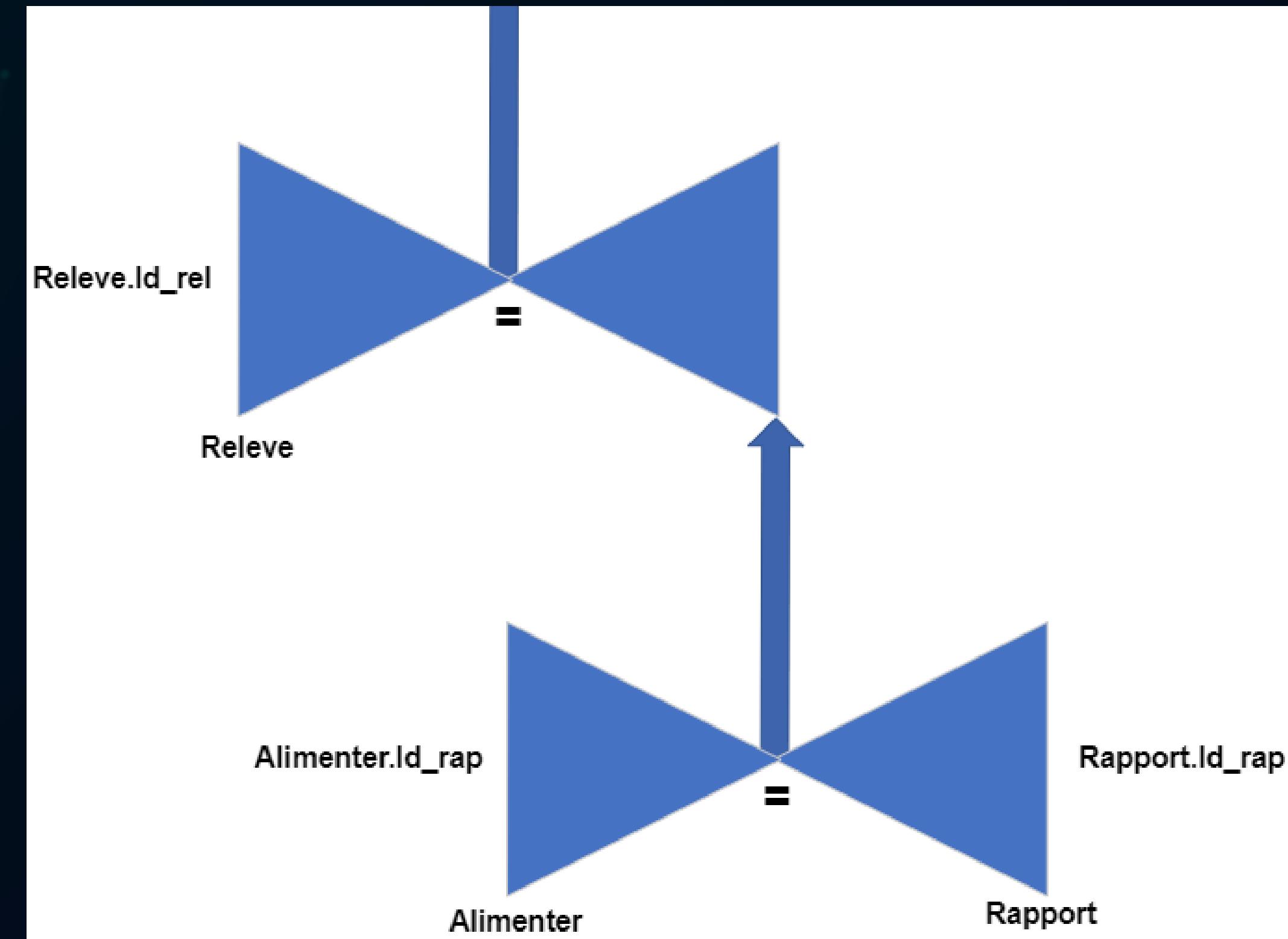
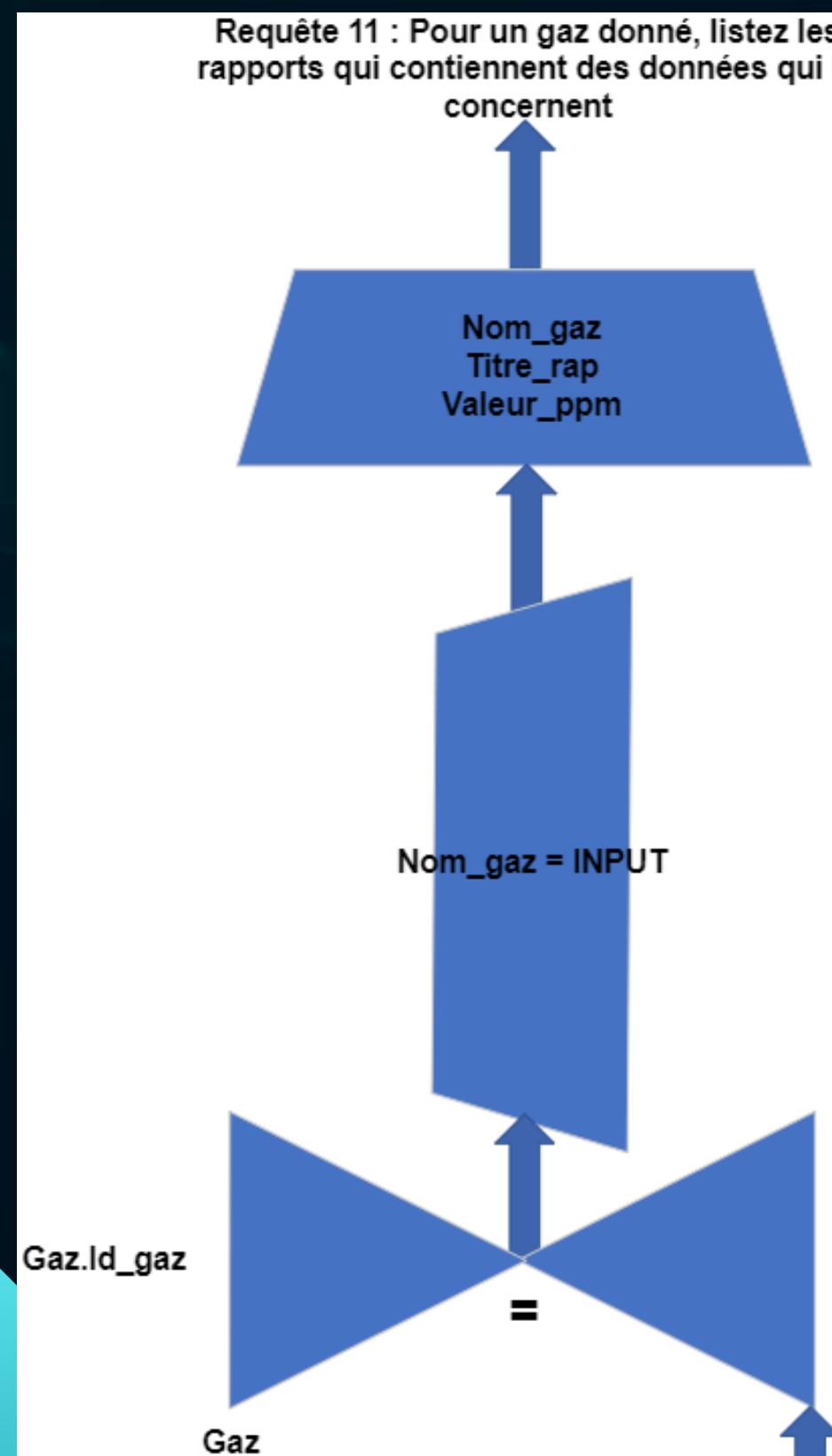
LES ARBRES ALGÉBRIQUES



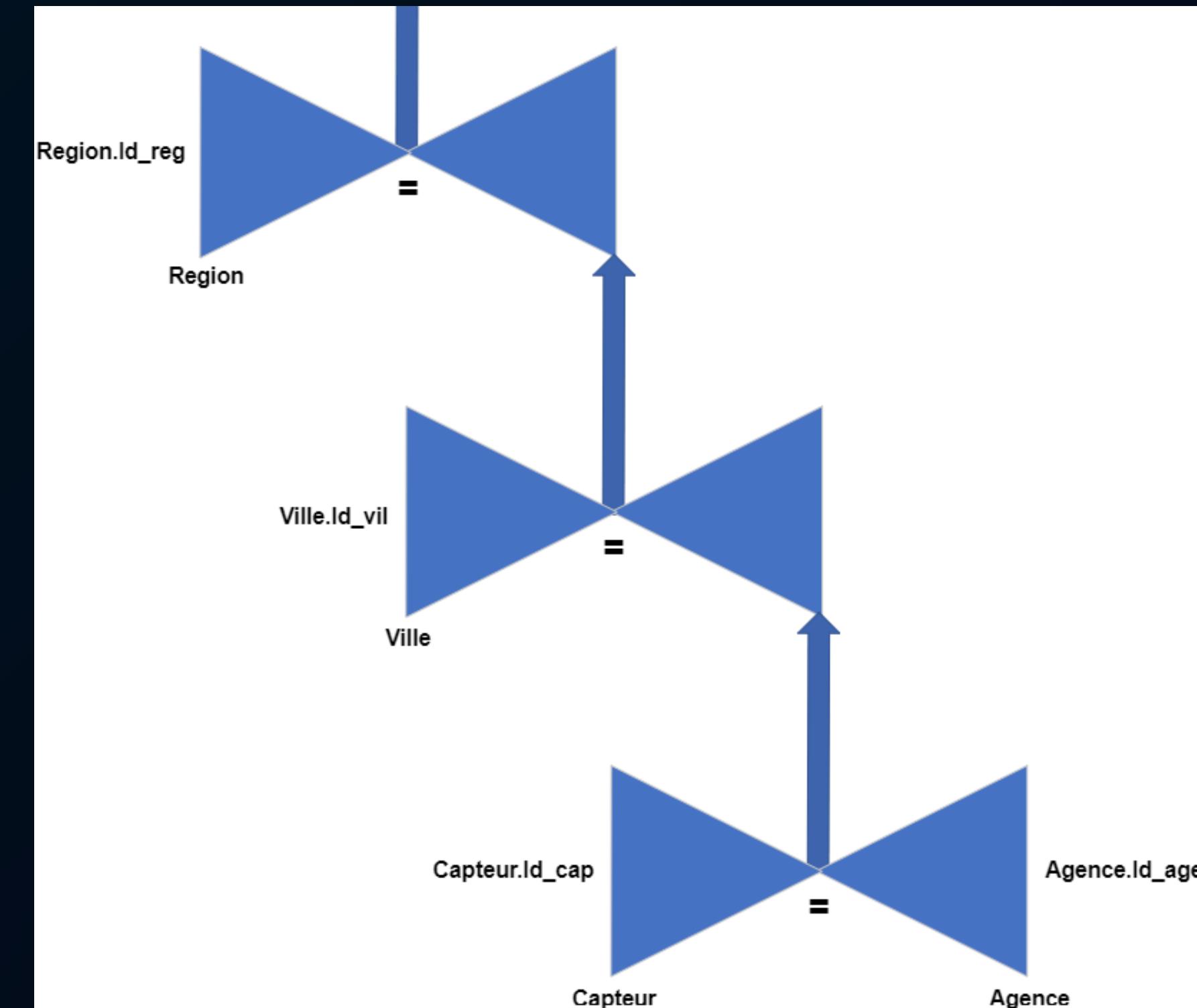
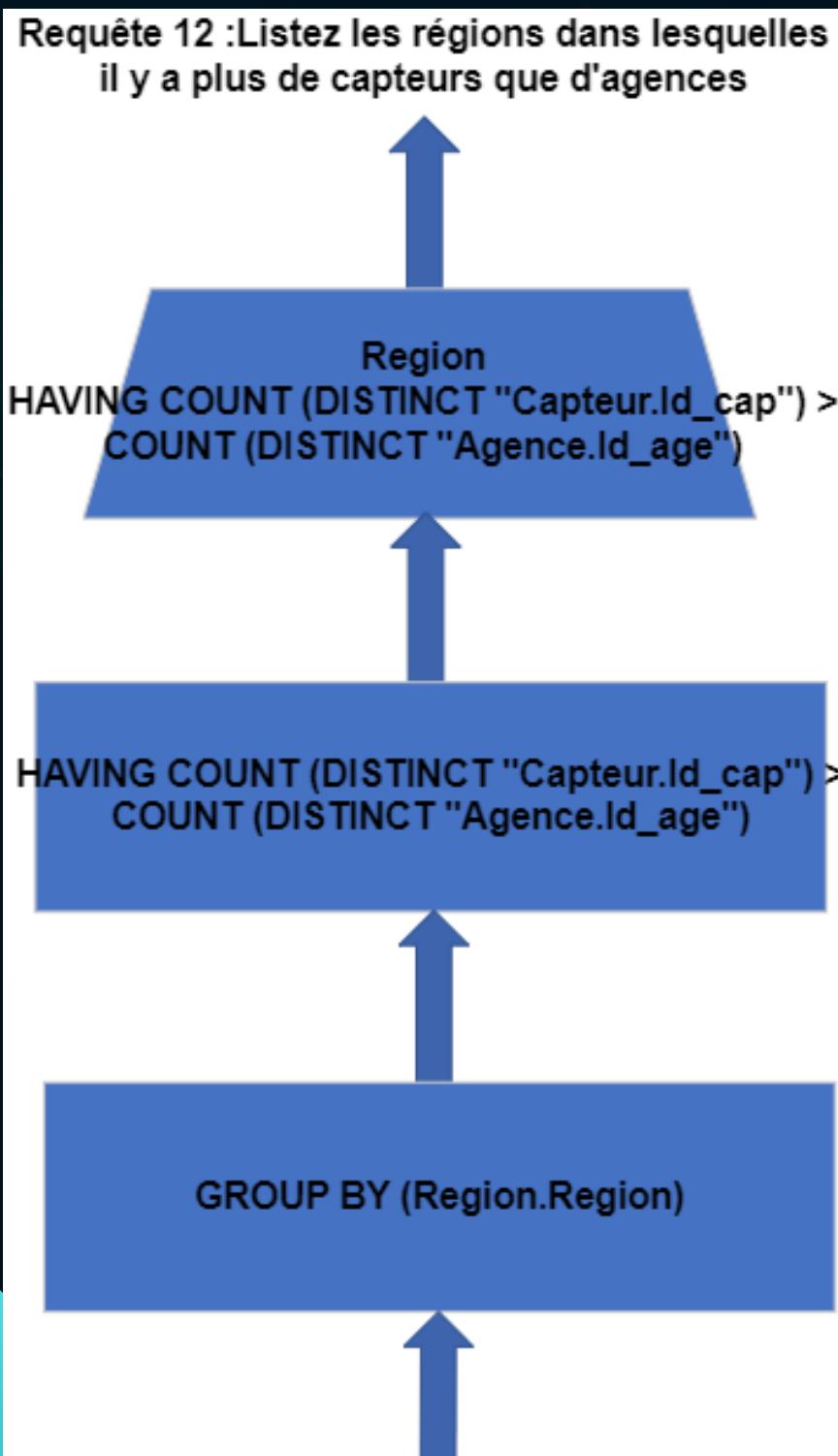
LES ARBRES ALGÉBRIQUES



LES ARBRES ALGÉBRIQUES



LES ARBRES ALGÉBRIQUES



CONCLUSION

Pour conclure, ce livrable a impliqué plusieurs choix de modélisation afin de visualiser les éléments et les liaisons qui composeront notre future base de données. Tout d'abord, nous avons créé un dictionnaire de données pour lister les informations qui regrouperont les données brutes. Ensuite, nous avons établi le Modèle Conceptuel des Données (MCD) pour organiser nos données en différentes entités, avec leurs attributs respectifs et les relations entre eux. Cette étape a été suivie par la conception du Modèle Logique des Données (MLD), qui offre une vision plus concrète de l'organisation et du stockage effectifs des données dans la base de données. Finalement, nous avons développé le Modèle Physique des Données (MPD), qui détaille la structure physique et les aspects techniques qui seront implémentés dans la base de données. Nous avons également créé des arbres algébriques pour structurer les requêtes SQL, permettant ainsi une exécution plus optimisée.

Grâce à toutes ces étapes, nous avons pu atteindre les objectifs de ce livrable, qui contribuent à l'efficacité globale du système de gestion pour notre futur base de données.