

PROJECT : DONNÉES ET MODÉLISATION

LIVRABLE 01



MEMBRE DU GROUPE :

RAMI MOHAMED AMINE

MOBAREK MALIK

ACHIR NASSIM

SOMMAIRE :

INTRODUCTION :

Dictionnaire de donnes :

Conception :

- MCD (Modèle Conceptuel de Données).

- MLD (Modèle Logique de Données).

- MPD (Modèle Physique de Données).

Requêtes :

Conclusion :

INTRODUCTION :

Le présent livrable représente une étape cruciale dans le développement du projet en cours. À travers ce document, les étudiants fournissent une analyse détaillée ainsi que la conception des données nécessaires à la mise en œuvre du système.

Dans ce livrable, nous avons inclus le fruit de notre travail comprenant cinq éléments clés : le dictionnaire de données, le modèle conceptuel de données (MCD), le modèle logique de données (MLD), le modèle physique de données (MPD) et les arbres algébriques des requêtes requises pour le projet.

Tous ces éléments sont regroupés dans un document unique au format PDF pour assurer une présentation cohérente et facile à consulter. Nous avons accordé une attention particulière à la lisibilité de chaque schéma afin de faciliter la compréhension et l'interprétation de notre travail par le Pilote.

De plus, ce document contient une section explicative détaillant les choix de modélisation opérés par notre groupe. Chaque décision a été prise en tenant compte des exigences du projet ainsi que des meilleures pratiques en matière de conception de bases de données.

Il convient de souligner que le Modèle Conceptuel de Données (MCD) est conforme à la 3ème forme normale, garantissant ainsi une structure optimale et évitant les anomalies de données potentielles.

Dictionnaire de donnes :

| Nom | Type | Taille | Commentaire |
|------------------------|----------------------|--------|---|
| id_Agence | Numérique | | identifiant de l'agence |
| Nom_Agence | Chaîne de caractères | 100 | Nom de l'agence |
| id_Personnel | Numérique | | identifiant du personnel |
| Nom | Chaîne de caractères | 100 | Nom du personnel |
| Prénom | Chaîne de caractères | 100 | Prenom du personnel |
| Date_de_naissance | DATE | | Date de naissance du personnel |
| Date_de_prise_du_poste | DATE | | Date de prise du poste du personnel |
| Adresse_postale | Chaîne de caractères | 255 | Adresse postale du personnel |
| Num_de_telephone | Numérique | | Numéro de téléphone du personnel |
| id_Capteur | Numérique | | identifiant du capteur |
| Nom_Capteur | Chaîne de caractères | 50 | Nom du Capteur |
| id_Rapport | Numérique | | identifiant du rapport |
| Titre | Chaîne de caractères | 255 | Titre du Rapport |
| Date_de_parution | DATE | | Date de parution du rapport |
| id_Gaz | Numérique | | identifiant du Gaz |
| Nom_de_Gaz | Chaîne de caractères | 50 | Nom du gaz polluant |
| Classification | Chaîne de caractères | 50 | gaz polluants sont classifiés comme "acidificateurs" serre" |
| id_Ville | Numérique | | identifiant de la ville |
| Nom_de_Ville | Chaîne de caractères | 255 | Nom de la Ville |
| id_Secteur | Numérique | | identifinant du Secteur d'activité |
| Nom_de_Secteur | Chaîne de caractères | 255 | Nom du Secteur d'activité Concernés |
| id_Région | Numérique | | identifiant de la région |
| Nom_de_Région | Chaîne de caractères | 255 | Nom de la région |
| id_Données | Numérique | 255 | identifaint de la donnée |
| Date_relevé | DATE | | Date de relevé des ensembles de données |
| Mesure_ppm | Numérique | | Mesures de quantité d'air en ppm |

La table décrite dans le dictionnaire de données comprend les informations suivantes :

ID Agence : Un numéro unique qui identifie chaque agence dans le système.

Nom Agence : Le nom de l'agence, stocké sous forme de chaîne de caractères avec une longueur maximale de 100 caractères.

ID Personnel : Identifiant unique pour chaque membre du personnel. Nom Personnel : Le nom du personnel, stocké sous forme de chaîne de caractères avec une longueur maximale de 100 caractères.

Prénom Personnel : Le prénom du personnel, stocké sous forme de chaîne de caractères avec une longueur maximale de 100 caractères.

Date de naissance : La date de naissance du personnel, stockée au format DATE.

Date de prise du poste : La date à laquelle le personnel a pris son poste, stockée au format DATE.

Adresse postale : L'adresse postale du personnel, stockée sous forme de chaîne de caractères avec une longueur maximale de 255 caractères.

Numéro de téléphone : Le numéro de téléphone du personnel, stocké sous forme numérique.

ID Capteur : Identifiant unique pour chaque capteur dans le système.

Nom Capteur : Le nom du capteur, stocké sous forme de chaîne de caractères avec une longueur maximale de 50 caractères.

ID Rapport : Un numéro unique qui identifie chaque rapport dans le système.

Titre : Le titre du rapport, stocké sous forme de chaîne de caractères avec une longueur maximale de 255 caractères.

Date de parution : La date de parution du rapport, stockée au format DATE.

ID Gaz : Identifiant unique pour chaque type de gaz polluant.

Nom de Gaz : Le nom du gaz polluant, stocké sous forme de chaîne de caractères avec une longueur maximale de 50 caractères.

Classification : La classification du gaz polluant, stockée sous forme de chaîne de caractères avec une longueur maximale de 50 caractères, pouvant être "acidifiants" ou "gaz à effet de serre".

ID Ville : Identifiant unique pour chaque ville dans le système.

Nom de Ville : Le nom de la ville, stocké sous forme de chaîne de caractères avec une longueur maximale de 255 caractères.

ID Secteur : Identifiant unique pour chaque secteur d'activité concerné.

Nom de Secteur : Le nom du secteur d'activité concerné, stocké sous forme de chaîne de caractères avec une longueur maximale de 255 caractères.

ID Région : Identifiant unique pour chaque région dans le système.

Nom de Région : Le nom de la région, stocké sous forme de chaîne de caractères avec une longueur maximale de 255 caractères.

ID Données : Identifiant unique pour chaque ensemble de données.

Date relevée : La date à laquelle les données ont été relevées, stockée au format DATE.

Mesures en ppm : Les mesures de quantité d'air en parties par million (ppm), stockées sous forme numérique.

Conception :

MCD (Modèle Conceptuel de Données) :

Un MCD, ou Modèle Conceptuel de Données, est une représentation abstraite et simplifiée de la structure des données d'un système d'information, souvent utilisé dans le cadre de la conception de bases de données.

Le but principal d'un MCD est de décrire les entités importantes dans le domaine d'application du système, ainsi que les relations qui existent entre ces entités. Il met l'accent sur la logique et la structure des données, indépendamment de toute considération technique ou physique de mise en œuvre.

En général, un MCD est constitué de plusieurs éléments :

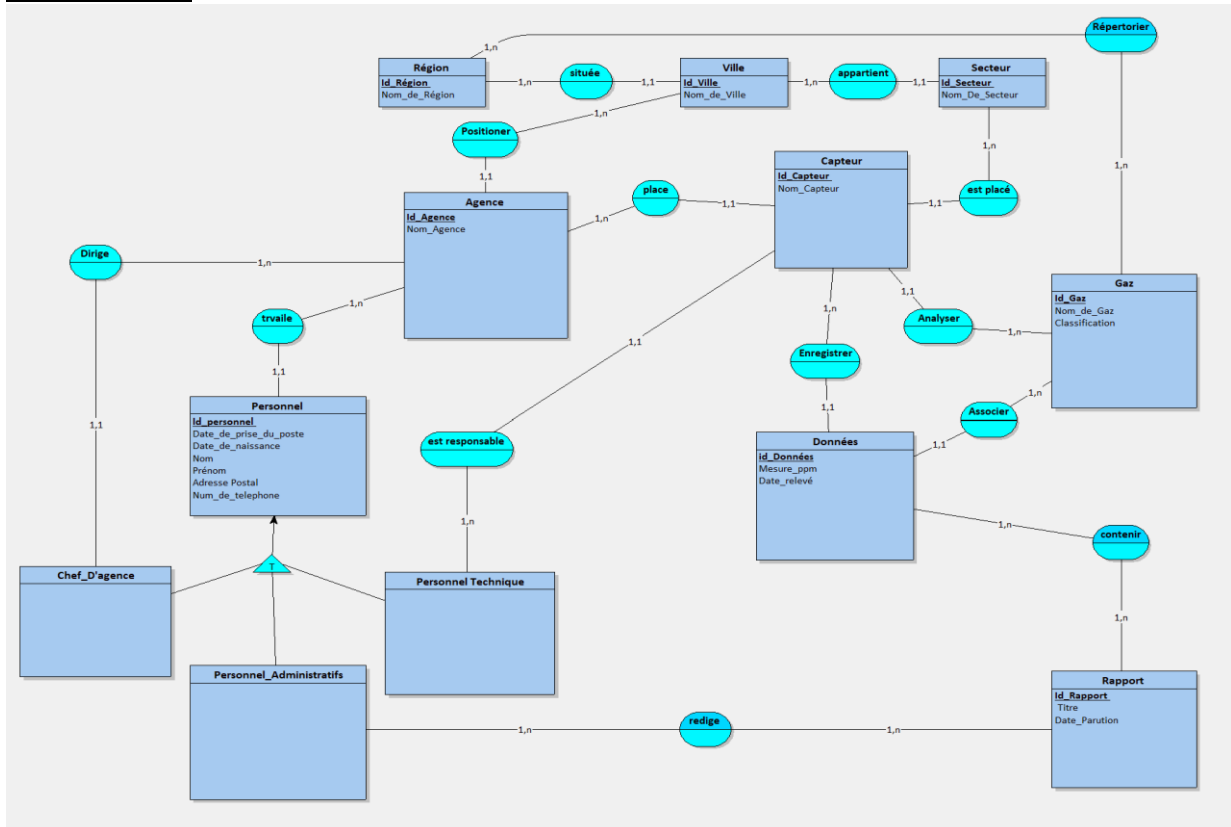
Entités : Ce sont des objets, des concepts ou des choses du monde réel sur lesquels l'entreprise souhaite conserver des informations. Par exemple, dans un système de gestion d'une bibliothèque, les entités pourraient inclure des livres, des auteurs et des emprunteurs.

Attributs : Ce sont les propriétés ou les caractéristiques des entités. Chaque entité peut avoir plusieurs attributs qui décrivent différentes informations à son sujet. Par exemple, pour l'entité "Livres", les attributs pourraient inclure le titre, l'auteur et l'année de publication.

Relations : Ce sont les associations ou les liens entre différentes entités. Elles décrivent comment les entités sont connectées les unes aux autres et peuvent être de différents types, comme les relations un-à-un, un-à-plusieurs ou plusieurs-à-plusieurs.

Clés primaires et étrangères : Les clés primaires sont des attributs qui identifient de manière unique chaque occurrence d'une entité. Les clés étrangères sont des attributs qui établissent des liens entre différentes entités en utilisant les clés primaires.

Notre MCD :



Description de la structure des données sous forme de Modèle Conceptuel de Données (MCD) :

Entités :

Agence :

Id_Agence (INT, PK)

Nom_Agence (VARCHAR(50))

Id_Ville_ (INT, FK référençant Ville_)

Personnel :

Id_personnel_ (INT, PK)

Date_de_prise_du_poste (DATE)

Date_de_naissance_ (DATE, NOT NULL)

Nom (VARCHAR(50))

Prénom_ (VARCHAR(50))

Adresse_Postal_ (VARCHAR(50))

Num_de_telephone (VARCHAR(50))

Id_Agence (INT, FK référençant Agence_)

Rapport :

Id_Rapport_ (INT, PK)

_Titre (VARCHAR(50))

Date_Parution (DATE)

Secteur :

Id_Secteur_ (INT, PK)

Nom_De_Secteur_ (VARCHAR(50))

Id_Ville_ (INT, FK référençant Ville_)

Personnel_Technique :

Id_personnel_ (INT, PK, FK référençant Personnel_)

Chef_D_agence :

Id_personnel_ (INT, PK, FK référençant Personnel_)

Id_Agence (INT, FK référençant Agence_)

Personnel Administratifs :

Id_personnel_ (INT, PK, FK référençant Personnel_)

Gaz :

Id_Gaz (INT, PK)

Nom_de_Gaz (VARCHAR(20))

Classification_ (VARCHAR(50))

Région :

Id_Région_ (INT, PK)

Nom_de_Région (VARCHAR(50))

Ville :

Id_Ville_ (INT, PK)

Nom_de_Ville_ (VARCHAR(40))

Id_Région_ (INT, FK référençant Région_)

Capteur :

Id_Capteur_ (INT, PK)

Nom_Capteur_ (VARCHAR(50))

Id_personnel_ (INT, FK référençant Personnel_Technique)

Id_Secteur_ (INT, FK référençant Secteur_)

Id_Agence (INT, FK référençant Agence_)

Id_Gaz (INT, FK référençant Gaz)

Données :

id_Données (INT, PK)

Mesure_ppm (DOUBLE)

Date_relevé (VARCHAR(50))

Id_Gaz (INT, FK référençant Gaz)

Id_Capteur_ (INT, FK référençant Capteur)

Relations :

Contenir :

Id_Rapport_ (INT, FK référençant Rapport_)

id_Données (INT, FK référençant Données)

(PK: Id_Rapport_, id_Données)

Redige_ :

Id_personnel_ (INT, FK référençant Personnel_Administratifs)

Id_Rapport_ (INT, FK référençant Rapport_)

(PK: Id_personnel_, Id_Rapport_)

Répertorier :

Id_Gaz (INT, FK référençant Gaz)

Id_Région_ (INT, FK référençant Région_)

(PK: Id_Gaz, Id_Région_)

Choix de modélisation :

Lors de la conception de notre modèle conceptuel de données (MCD), nous avons cherché à simplifier autant que possible les composants du dictionnaire de données pour faciliter la réalisation des requêtes et des opérations algébriques ultérieures. Nous avons veillé à ce que le modèle soit normalisé jusqu'à la troisième forme normale (3NF).

Pour simplifier la structure, nous avons regroupé les différentes catégories d'agents (chef d'agence, agent technique et agent administratif) en un seul attribut nommé "profession", qui fait partie de l'entité "personnel". Cette approche permet de réduire

la complexité du MCD en évitant la nécessité d'utiliser un mécanisme d'héritage pour l'entité "personnel".

L'entité "personnel" contient les attributs communs à toutes les catégories d'agents, tandis que les entités spécialisées (Chef_D'agence, Personnel_Technique, Personnel_Administratifs) ajoutent des attributs spécifiques à chaque catégorie.

Cet héritage permet de réduire la redondance des données et de maintenir une structure plus claire et plus facile à gérer. Il facilite également la réalisation des requêtes et des opérations algébriques ultérieures en permettant une approche plus uniforme pour traiter les différents types d'agents.

La première relation, "Contenir", établit un lien entre les rapports et les données. Elle signifie qu'un rapport peut contenir plusieurs données, et une donnée peut être associée à plusieurs rapports. En d'autres termes, c'est une relation de type "n:m" entre les entités Rapport et Données.

La seconde relation, "Redige_", relie le personnel administratif aux rapports. Elle indique qu'un membre du personnel administratif peut rédiger plusieurs rapports, mais un rapport est rédigé par un seul membre du personnel administratif. Cette relation est de type "n:1" entre les entités Personnel_Administratifs et Rapport.

Enfin, la troisième relation, "Répertoire", établit une association entre les gaz et les régions. Elle signifie qu'un gaz peut être répertorié dans plusieurs régions, et une région peut contenir plusieurs types de gaz. Cette relation est également de type "n:m" entre les entités Gaz et Région.

Ces relations permettent de définir les interactions et les liens entre les différentes entités de votre système, contribuant ainsi à la structuration et à la compréhension de votre modèle conceptuel de données.

Grâce à l'utilisation de l'outil Looping, nous avons réussi à élaborer un MCD qui respecte les principes de la troisième forme normale, comprenant six entités et huit associations. Ce modèle servira de base pour la création du modèle logique de données (MLD), puis du modèle physique de données (MPD). Ensuite, nous pourrons générer les opérations algébriques nécessaires pour la mise en place de notre base de données.

MLD (Modèle Logique de Données) :

Représentation intermédiaire entre le Modèle Conceptuel de Données (MCD) et le Modèle Physique de Données (MPD) dans le processus de conception de bases de données.

Alors que le MCD se concentre sur la structure conceptuelle des données en utilisant des concepts tels que les entités, les attributs et les relations, le MLD se rapproche davantage de la manière dont ces concepts seront implémentés dans une base de données relationnelle spécifique.

Le MLD traduit les entités, les attributs, les relations et les contraintes du MCD en termes de tables, de colonnes, de clés primaires, de clés étrangères et d'autres éléments spécifiques à un système de gestion de base de données relationnelle (SGBDR).

Caractéristiques typiques d'un MLD :

Tables : Chaque entité du MCD devient une table dans le MLD.

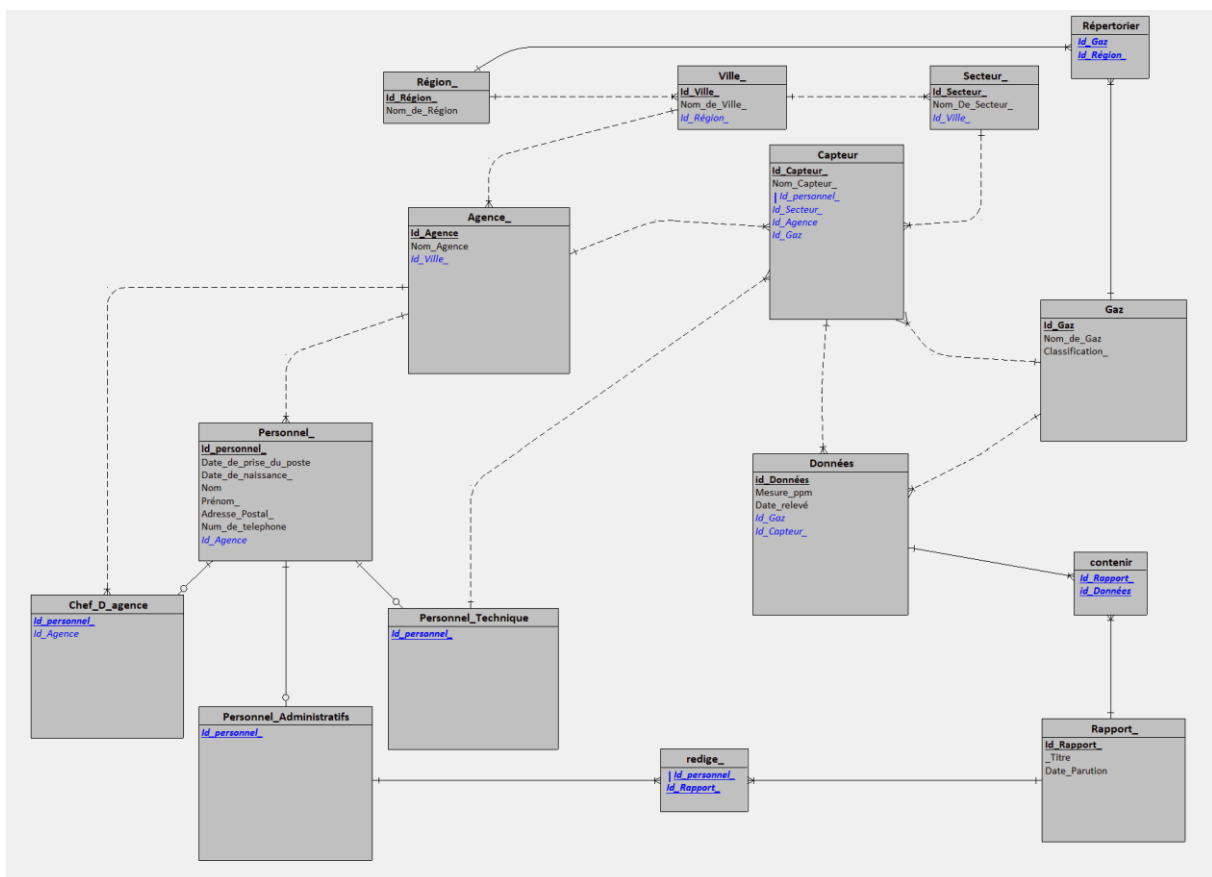
Colonnes : Les attributs du MCD deviennent des colonnes dans les tables du MLD.

Clés primaires : Les clés primaires sont définies pour chaque table pour garantir l'unicité des enregistrements.

Clés étrangères : Les relations entre les entités du MCD deviennent des contraintes de clés étrangères dans le MLD pour assurer l'intégrité référentielle entre les tables.

Contraintes : Les contraintes d'intégrité, telles que les contraintes de clés étrangères et les contraintes de domaine, sont spécifiées pour garantir la cohérence des données.

Notre MLD :



MLD TEXTUEL :

Gaz = (Id_Gaz INT, Nom_de_Gaz VARCHAR(20), Classification_ VARCHAR(50));

Région_ = (Id_Région INT, Nom_de_Région VARCHAR(50));

Ville_ = (Id_Ville_ *INT*, Nom_de_Ville_ *VARCHAR(40)*, #Id_Région_);
Agence_ = (Id_Agence *INT*, Nom_Agence *VARCHAR(50)*, #Id_Ville_);
**Personnel_ = (Id_personnel *INT*, Date_de_prise_du_poste *DATE*, Date_de_naissance_ *DATE*,
 Nom *VARCHAR(50)*, Prénom_ *VARCHAR(50)*, Adresse_Postal_ *VARCHAR(50)*, Num_de_telephone *VARCHAR(50)*,
 #Id_Agence);**
Rapport_ = (Id_Rapport *INT*, _Titre *VARCHAR(50)*, Date_Parution *DATE*);
Secteur_ = (Id_Secteur *INT*, Nom_De_Secteur_ *VARCHAR(50)*, #Id_Ville_);
Personnel_Technique = (#Id_personnel);
Chef_D_agence = (#Id_personnel , #Id_Agence);
Personnel_Administratifs = (#Id_personnel);
**Capteur = (Id_Capteur *INT*, Nom_Capteur_ *VARCHAR(50)*, #(#Id_personnel_), #Id_Secteur_,
 #Id_Agence, #Id_Gaz);**
**Données = (id_Données *INT*, Mesure_ppm *DOUBLE*, Date_relevé *VARCHAR(50)*, #Id_Gaz,
 #Id_Capteur_);**
contenir = (#Id_Rapport , #id Données);
redige_ = (#(#Id_personnel), #Id_Rapport);
Répertorier = (#Id_Gaz, #Id Région);
Explication :

Entités :

Gaz :

Attributs : Id_Gaz (identifiant du gaz), Nom_de_Gaz (nom du gaz), Classification_ (classification du gaz).

Région :

Attributs : Id_Région_ (identifiant de la région), Nom_de_Région (nom de la région).

Ville :

Attributs : Id_Ville_ (identifiant de la ville), Nom_de_Ville (nom de la ville).

Relations : Une ville est associée à une région (#Id_Région_).

Agence :

Attributs : Id_Agence (identifiant de l'agence), Nom_Agence (nom de l'agence).

Relations : Une agence est située dans une ville (#Id_Ville_).

Personnel :

Attributs : Id_personnel_ (identifiant du personnel), Date_de_prise_du_poste (date de prise de poste), Date_de_naissance_ (date de naissance), Nom (nom), Prénom_ (prénom), Adresse_Postal_ (adresse postale), Num_de_telephone (numéro de téléphone).

Relations : Le personnel travaille pour une agence (#Id_Agence).

Rapport :

Attributs : Id_Rapport_ (identifiant du rapport), _Titre (titre du rapport), Date_Parution (date de parution).

Secteur :

Attributs : Id_Secteur_ (identifiant du secteur), Nom_De_Secteur_ (nom du secteur).

Relations : Un secteur est situé dans une ville (#Id_Ville_).

Personnel Technique :

Relations : Ce sont les membres du personnel qui jouent un rôle technique dans l'organisation.

Chef D agence :

Relations : Ce sont les membres du personnel qui jouent un rôle de chef dans une agence spécifique.

Personnel Administratifs :

Relations : Ce sont les membres du personnel qui jouent un rôle administratif dans l'organisation.

Capteur :

Attributs : Id_Capteur_ (identifiant du capteur), Nom_Capteur_ (nom du capteur).

Relations : Un capteur est attribué à un personnel (#Id_personnel_), est situé dans un secteur (#Id_Secteur_), est associé à une agence (#Id_Agence) et mesure un type de gaz (#Id_Gaz).

Données :

Attributs : id_Données (identifiant des données), Mesure_ppm (mesure en parties par million), Date_relevé (date de relevé).

Relations : Les données sont associées à un type de gaz (#Id_Gaz) et à un capteur (#Id_Capteur_).

Relations :

Contenir :

Relations : Les rapports contiennent des données.

Redige _ :

Relations : Les membres du personnel rédigent des rapports.

Répertorier :

Relations : Les types de gaz sont répertoriés dans les régions.

MPD (Modèle Physique de Données) :

Est une représentation concrète et détaillée de la structure des données d'une base de données, telles qu'elles seront stockées physiquement et implémentées dans un système de gestion de base de données (SGBD) spécifique. Contrairement au MCD et au MLD qui se concentrent sur la structure logique des données, le MPD prend en compte les aspects physiques de stockage et d'optimisation des données pour un SGBD particulier.

MPD inclut généralement :

Schéma de base de données : Le MPD détaille la structure de la base de données en spécifiant les tables, les colonnes et les contraintes de chaque table.

Types de données : Il précise les types de données utilisés pour chaque colonne dans les tables de la base de données, tels que VARCHAR, INT, DATE, etc.

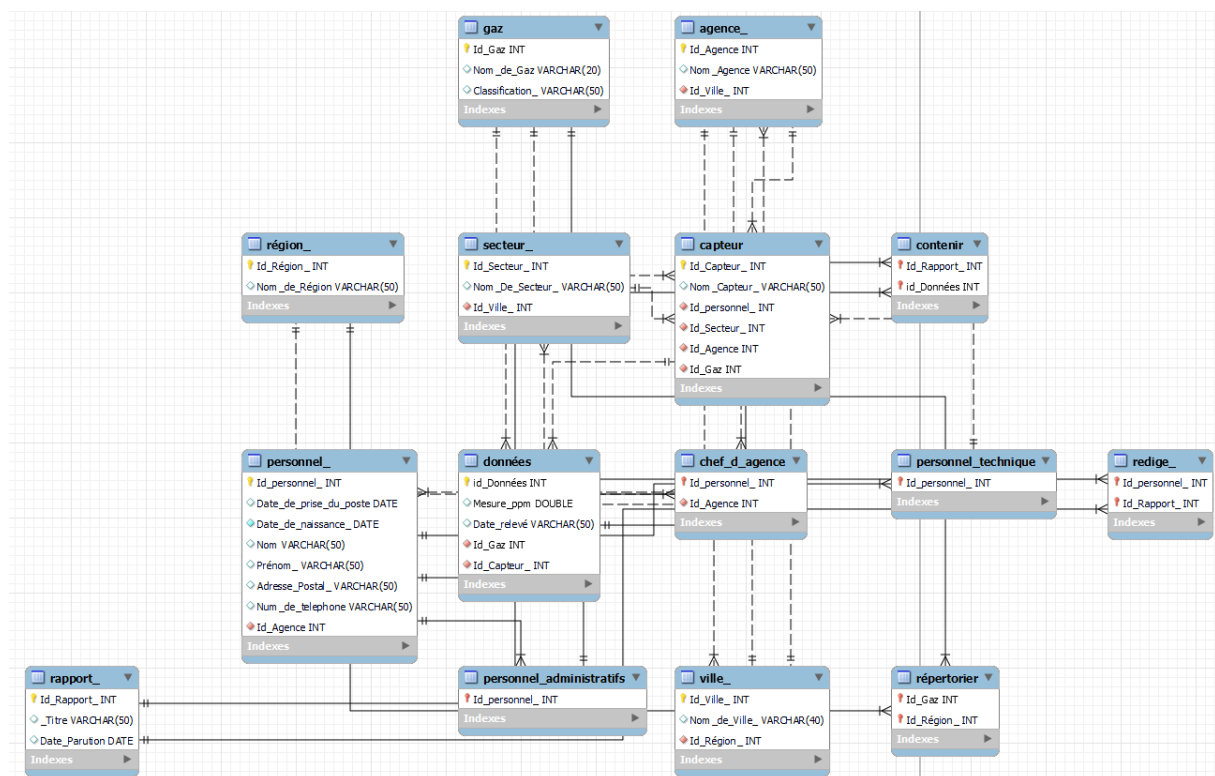
Index : Le MPD peut spécifier les index utilisés pour accélérer les opérations de recherche et de jointure dans la base de données.

Clés primaires et étrangères : Il inclut les définitions des clés primaires et étrangères pour garantir l'intégrité des données et les relations entre les différentes tables.

Contraintes : Le MPD peut inclure des contraintes supplémentaires telles que des contraintes de vérification, des contraintes de valeurs uniques, etc.

Configuration spécifique au SGBD : Il peut également inclure des paramètres de configuration spécifiques au SGBD utilisé, tels que la taille de la mémoire tampon, la configuration du cache, etc.

Notre MPD :



En forme de code :

```
1  CREATE TABLE Gaz(  
2      Id_Gaz INT,  
3      Nom_de_Gaz VARCHAR(20),  
4      Classification_ VARCHAR(50),  
5      PRIMARY KEY(Id_Gaz)  
6  );  
7  
8  CREATE TABLE Région_  
9      Id_Région_ INT,  
10     Nom_de_Région VARCHAR(50),  
11     PRIMARY KEY(Id_Région_)  
12 );  
13  
14 CREATE TABLE Ville_  
15     Id_Ville_ INT,  
16     Nom_de_Ville_ VARCHAR(40),  
17     Id_Région_ INT NOT NULL,  
18     PRIMARY KEY(Id_Ville_),  
19     FOREIGN KEY(Id_Région_) REFERENCES Région_(Id_Région_)  
20 );  
21  
22 CREATE TABLE Agence_  
23     Id_Agence INT,  
24     Nom_Agence VARCHAR(50),  
25     Id_Ville_ INT NOT NULL,  
26     PRIMARY KEY(Id_Agence),  
27     FOREIGN KEY(Id_Ville_) REFERENCES Ville_(Id_Ville_)  
28 );  
29  
30 CREATE TABLE Personnel_  
31     Id_personnel_ INT,  
32     Date_de_prise_du_poste DATE,  
33     Date_de_naissance_ DATE NOT NULL,  
34     Nom VARCHAR(50),  
35     Prénom_ VARCHAR(50),  
36     Adresse_Postal_ VARCHAR(50),  
37     Num_de_telephone VARCHAR(50),  
38     Id_Agence INT NOT NULL,  
39     PRIMARY KEY(Id_personnel_),  
40     FOREIGN KEY(Id_Agence) REFERENCES Agence_(Id_Agence)  
41 );  
42  
43 CREATE TABLE Rapport_  
44     Id_Rapport_ INT,  
45     _Titre VARCHAR(50),  
46     Date_Parution DATE,  
47     PRIMARY KEY(Id_Rapport_)  
48 );  
49  
50 CREATE TABLE Secteur_  
51     Id_Secteur_ INT,  
52     Nom_De_Secteur_ VARCHAR(50),  
53     Id_Ville_ INT NOT NULL,  
54     PRIMARY KEY(Id_Secteur_),  
55     FOREIGN KEY(Id_Ville_) REFERENCES Ville_(Id_Ville_)  
56 );
```

```

50 • ⊖ CREATE TABLE Secteur_(
51     Id_Secteur_ INT,
52     Nom_De_Secteur_ VARCHAR(50),
53     Id_Ville_ INT NOT NULL,
54     PRIMARY KEY(Id_Secteur_),
55     FOREIGN KEY(Id_Ville_) REFERENCES Ville_(Id_Ville_)
56 );
57
58 • ⊖ CREATE TABLE Personnel_Technique(
59     Id_personnel_ INT,
60     PRIMARY KEY(Id_personnel_),
61     FOREIGN KEY(Id_personnel_) REFERENCES Personnel_(Id_personnel_)
62 );
63
64 • ⊖ CREATE TABLE Chef_D_agence(
65     Id_personnel_ INT,
66     Id_Agence INT NOT NULL,
67     PRIMARY KEY(Id_personnel_),
68     FOREIGN KEY(Id_personnel_) REFERENCES Personnel_(Id_personnel_),
69     FOREIGN KEY(Id_Agence) REFERENCES Agence_(Id_Agence)
70 );
71
72 • ⊖ CREATE TABLE Personnel_Administratifs(
73     Id_personnel_ INT,
74     PRIMARY KEY(Id_personnel_),
75     FOREIGN KEY(Id_personnel_) REFERENCES Personnel_(Id_personnel_)
76 );
77
78 • ⊖ CREATE TABLE Capteur(
79     Id_Capteur_ INT,
80     Nom_Capteur_ VARCHAR(50),
81     Id_personnel_ INT NOT NULL,
82     Id_Secteur_ INT NOT NULL,
83     Id_Agence INT NOT NULL,
84     Id_Gaz INT NOT NULL,
85     PRIMARY KEY(Id_Capteur_),
86     FOREIGN KEY(Id_personnel_) REFERENCES Personnel_Technique(Id_personnel_),
87     FOREIGN KEY(Id_Secteur_) REFERENCES Secteur_(Id_Secteur_),
88     FOREIGN KEY(Id_Agence) REFERENCES Agence_(Id_Agence),
89     FOREIGN KEY(Id_Gaz) REFERENCES Gaz(Id_Gaz)
90 );

```



```

92 • ○ CREATE TABLE Données(
93     id_Données INT,
94     Mesure_ppm DOUBLE,
95     Date_relevé VARCHAR(50),
96     Id_Gaz INT NOT NULL,
97     Id_Capteur_ INT NOT NULL,
98     PRIMARY KEY(id_Données),
99     FOREIGN KEY(Id_Gaz) REFERENCES Gaz(Id_Gaz),
100    FOREIGN KEY(Id_Capteur_) REFERENCES Capteur(Id_Capteur_)
101 );
102
103 • ○ CREATE TABLE contenir(
104     Id_Rapport_ INT,
105     id_Données INT,
106     PRIMARY KEY(Id_Rapport_, id_Données),
107     FOREIGN KEY(Id_Rapport_) REFERENCES Rapport_(Id_Rapport_),
108     FOREIGN KEY(id_Données) REFERENCES Données(id_Données)
109 );
110
111 • ○ CREATE TABLE redige_(
112     Id_personnel_ INT,
113     Id_Rapport_ INT,
114     PRIMARY KEY(Id_personnel_, Id_Rapport_),
115     FOREIGN KEY(Id_personnel_) REFERENCES Personnel_Administratifs(Id_personnel_),
116     FOREIGN KEY(Id_Rapport_) REFERENCES Rapport_(Id_Rapport_)
117 );
118
119 • ○ CREATE TABLE Répertorier(
120     Id_Gaz INT,
121     Id_Région_ INT,
122     PRIMARY KEY(Id_Gaz, Id_Région_),
123     FOREIGN KEY(Id_Gaz) REFERENCES Gaz(Id_Gaz),
124     FOREIGN KEY(Id_Région_) REFERENCES Région_(Id_Région_)
125 );

```

Requêtes :

Les requêtes illustrent plusieurs aspects de l'utilisation des bases de données dans le domaine de l'environnement :

Surveillance et suivi : Les requêtes telles que la liste des agences, le personnel technique et le nombre total de capteurs déployés permettent de suivre l'état et la portée du système de surveillance environnementale.

Analyse des données historiques : Les requêtes sur les rapports publiés entre certaines années et les émissions par région dans une année donnée permettent d'analyser les tendances historiques et d'évaluer les performances environnementales dans le temps.

Identification des problèmes : Les requêtes telles que l'identification du secteur le plus polluant ou des gaz émis dans une région spécifique permettent d'identifier les sources potentielles de problèmes environnementaux.

Suivi des performances du personnel : Les requêtes sur la productivité du personnel administratif et technique permettent de surveiller les performances individuelles et d'identifier les domaines où des améliorations peuvent être nécessaires.

Analyse détaillée : Les requêtes sur les rapports contenant des données spécifiques à un gaz ou à une région permettent une analyse détaillée des mesures et des analyses effectuées, ce qui peut être utile pour des études spécialisées ou des rapports de recherche.

En combinant ces requêtes et en interprétant les résultats, les gestionnaires environnementaux peuvent obtenir des informations précieuses pour prendre des décisions éclairées, mettre en œuvre des politiques efficaces et contribuer à la protection et à la préservation de l'environnement.

Exemple de requêtes :

Listez l'ensemble du personnel technique de l'agence de Bordeaux : Cette requête récupère tous les membres du personnel ayant un rôle technique et qui sont affectés à l'agence de Bordeaux, ce qui pourrait être utile pour la gestion des ressources humaines et des affectations.

Comment avoir la solution :

L'algèbre relationnelle est utilisée de manière générale pour exprimer les solutions de requêtes :

Projection (π) : Utilisée pour sélectionner les colonnes spécifiques d'une relation.

Sélection (σ) : Utilisée pour filtrer les lignes d'une relation en fonction de certaines conditions.

Jointure (\bowtie) : Utilisée pour combiner deux relations en fonction d'une condition de correspondance entre les valeurs de leurs attributs.

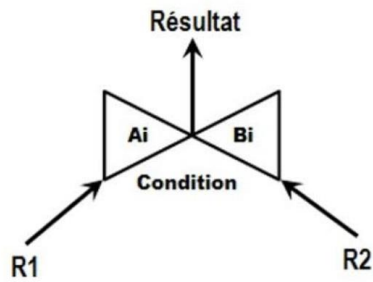
Groupement (Γ) : Utilisée pour regrouper les lignes d'une relation en fonction des valeurs d'un ou plusieurs attributs, souvent suivie d'une opération d'agrégation.

Agrégation (Σ) : Utilisée pour calculer des valeurs agrégées telles que la somme, la moyenne, le comptage, etc.

Tri (τ) : Utilisée pour trier les lignes d'une relation en fonction des valeurs d'un ou plusieurs attributs.

Intersection (\cap), Union (\cup), Différence ($-$) : Utilisées pour effectuer des opérations ensemblistes sur les relations.

Division (\div) : Utilisée pour sélectionner des tuples dans une relation qui sont associés à tous les tuples d'une autre relation.



Note : Si l'opérateur est = alors c'est une **Equi-jointure**
Sinon c'est une **Inéqui-jointure**

R1

| N | Nom | TEL |
|---|-------|-----------|
| 1 | ALI | 021324354 |
| 2 | AHMED | 021554354 |
| 3 | ALI | 034456543 |

R2

| Num | Nom | Adresse |
|-----|-------|---------|
| 0 | ALI | Alger |
| 2 | AHMED | Blida |
| 5 | MALIK | Oran |

R1 ⋈ R2
N >= Num

| N | Nom | TEL | Num | Nom | Adresse |
|---|-------|-----------|-----|-------|---------|
| 1 | ALI | 021324354 | 0 | ALI | Alger |
| 2 | AHMED | 021554354 | 0 | ALI | Alger |
| 2 | AHMED | 021554354 | 2 | AHMED | Blida |
| 3 | ALI | 034456543 | 0 | ALI | Alger |
| 3 | ALI | 034456543 | 2 | AHMED | Blida |

14

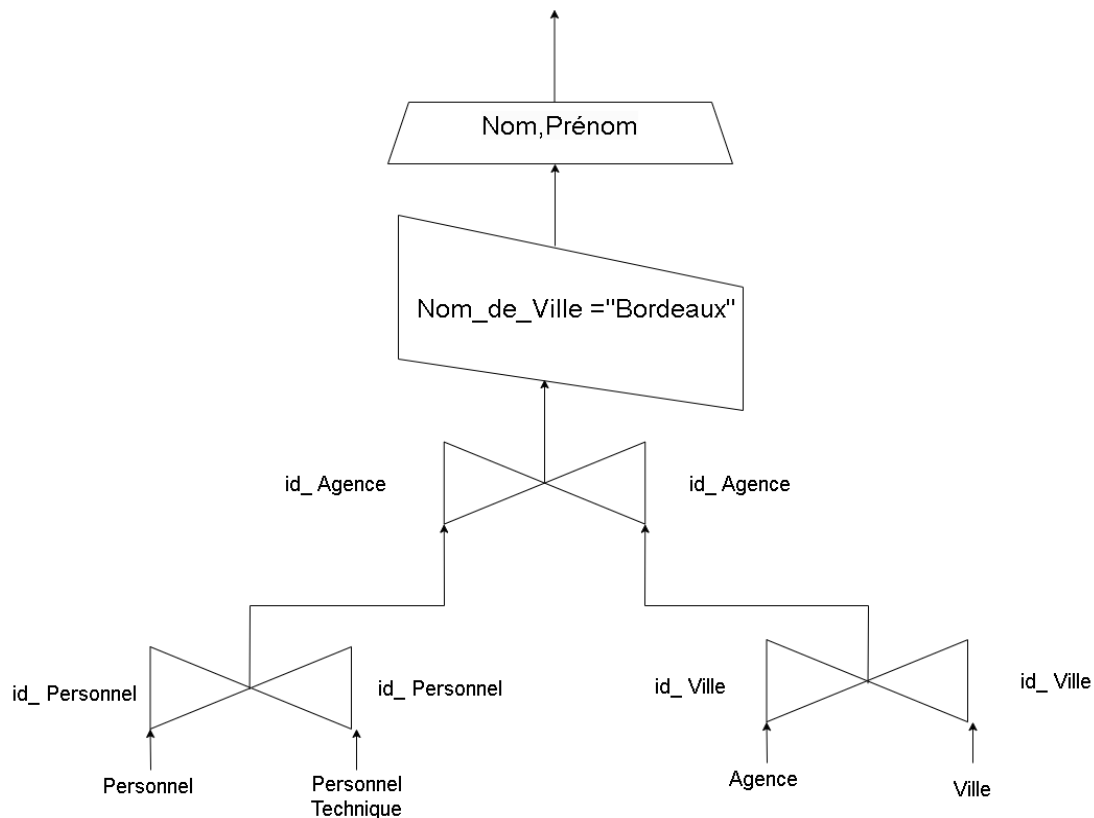
Fonction :

π [Nom, Prénom] σ (Nom_de_Ville = "Bordeaux") Personnel \bowtie Personnel technique : [Personnel]. ID

personnel = [Personnel technique]. ID personnel. Agence \bowtie Ville :

[Agence]. Id_Ville = [Ville]. Id_Ville. Personnel \bowtie Agence : [Agence].id_Agence = [Personnel].id_Agence

Arbre algébrique :



Conclusion :

En conclusion, le processus complet de conception et de mise en œuvre d'une base de données, du dictionnaire de données à la réalisation des requêtes avec des solutions arborescentes algébriques et algébriques relationnelles, représente une approche méthodique et structurée pour assurer le succès d'un projet. Chaque étape joue un rôle crucial dans la création d'une base de données robuste, performante et bien conçue.

Le dictionnaire de données établit une base solide en définissant clairement les éléments de données nécessaires et en assurant une compréhension commune entre les membres de l'équipe. Le modèle conceptuel de données (MCD) abstrait les concepts clés du système, tandis que le modèle logique de données (MLD) les transforme en un schéma détaillé prêt à être implémenté. Le modèle physique de données (MPD) finalise la conception en spécifiant les détails de l'implémentation dans un système de gestion de base de données particulier.

Les requêtes avec des solutions arborescentes algébriques et algébriques relationnelles offrent une perspective complète sur la manipulation des données, permettant une optimisation efficace et une compréhension approfondie des performances du système. Ces deux approches offrent une flexibilité dans la formulation des requêtes et garantissent une exécution efficace des opérations.

En combinant ces étapes, nous avons créé un système de base de données cohérent, bien documenté et capable de répondre aux besoins fonctionnels et de performance de l'application. Ce processus méthodique fournit une base solide pour le développement de systèmes de base de données fiables et évolutifs, essentiels pour la réussite des projets dans un large éventail de domaines.