

Rapport de Projet

Rapport rédigé suite à un projet effectué en :

L3 Informatique

Dans



FrencGrais

Conception d'une base de données pour un entrepôt

Sous la supervision de

M. THIBAUT BERNARD & M. STEPHANE CORMIER

Par

Souhail Mtarfi

Corentin Favrel

Année : 2022/2023

A l'Université de Reims Champagne-Ardenne



**UNIVERSITÉ
DE REIMS
CHAMPAGNE-ARDENNE**

Table des matières

Introduction	3
I. Analyse	4
II. Cahier des charges	4
III. Dictionnaire de données	5
a. Les entités	5
b. Les relations	7
IV. Modèle conceptuel de données	9
V. Modèle relationnel	10
VI. Modèle logique de données	11
VII. Modèle conceptuel de traitement	11
a. L'inventaire	11
c. Le contrôle	12
VIII. Normalisation	12
IX. Export de la base de données	16
a. Création des tables	16
b. L'ajout des clés étrangères	17
d. insertion des données	18
e. suppression des tables	21
X. Requêtes SQL	21
XI. Programme PL/SQL	23
Conclusion	26

Introduction

Le but de ce projet est de mettre en place un système de gestion entièrement informatisé à la suite de la construction d'un nouvel entrepôt de stockage ultramoderne.

Ce dossier comportera une introduction explicative afin de bien comprendre le but de ce projet, un dictionnaire de données sera présent afin d'identifier toutes les entités présentes mais aussi toutes les relations entre elles.

Un schéma sous forme de MCD a été mis en place afin d'avoir une vue simplifiée sur les différentes entités et relations.

Pour finir il y aura une partie concernant les requêtes SQL réalisés ainsi que 6 programmes PL/SQL pour différents types de requêtes.

Nous savons que les produits seront uniquement alimentaires. Les **produits** arriveront en **lots de produits** et auront une **fiche descriptive**. Chaque **produit** sera réglementé par un **cadre administratif**.

Les **lots de produits** ouvriront automatiquement un **Dossier de lot**, et auront obligatoirement des **conditions de stockage**. Les **lots de produits** subissent un contrôle par semaine, un **produit** sera choisi dans un **lot de produits**. Ce sera au **Technicien de stockage** de prélever un **produit** dans un **lot de produit**. Il devra ensuite remplir une **fiche d'inventaire**.

Les **conditions de stockage** seront précisées par le **cadre administratif** qui lui-même définit les **contrôles**.

I. Analyse

Le supermarché FrencGrais vient de construire un nouvel entrepôt de stockage à la pointe de la technologie. Il est conçu pour être un endroit où tous les produits sont stockés puis distribués dans les différents magasins. Pour concevoir cet entrepôt, ils devront créer une base de données pouvant stocker toutes les informations sur ce qui est stocké dans l'entrepôt.

Nous discuterons de la manière dont une entreprise procéderait pour concevoir une telle base de données pour son entrepôt, y compris le type de données qu'elle devrait inclure et le type de méthodes de stockage qu'elle devrait utiliser.

Les produits auront un type correspondant à une catégorie, comme : produits laitiers, fruits, boissons, surgelés...

La fiche descriptive appartiendra au produit, il y sera indiqué les informations du produit tels que le poids, le volume, une description et une date limite de consommation.

Un lot de produit correspond à tous les produits arrivant le même jour dans une même livraison .

Le dossier de lot est associé au lot de produit et il y sera indiqué la quantité de produits, la date d'entrée et la date de retrait.

Les conditions de stockage seront présentes dès l'arrivée d'un lot de produit et il y sera indiqué la durée de conservation maximum et la température minimum et maximum de stockage.

Le cadre administratif réglementera les produits et sera donc associés aux conditions de stockage.

Des contrôles seront effectués sur des produits où il sera indiqué l'origine du produit, la présence ou non d'OGM (défini par un type logique, KO pour non, OK pour oui), la présence de colorants sera vérifiée comme dans les cas des OGM. Il y aura une vérification de la date limite, un résultat pour dire si le produit est valide ou non et il sera également possible de mettre un commentaire pour plus de précision sur le contrôle.

Le technicien de stockage effectuera les contrôles, le technicien sera donc défini par son nom et prénom.

La fiche inventaire sera remplie par le technicien, il y sera précisé la date ainsi que le statut du lot de produit.

II. Cahier des charges

Le supermarché FrencGrais vient de construire un nouvel entrepôt de stockage à la pointe de la technologie. Le nouvel entrepôt est conçu pour stocker 2 000 palettes de marchandises et a une superficie de 3 600 mètres carrés. On y accède par une grande porte coulissante de 12 mètres de large et 16 mètres de haut. Le bâtiment est équipé d'un système de chauffage central qui peut être réglé en fonction de la météo.

L'entrepôt est conçu pour résister aux effets du changement climatique et des catastrophes naturelles telles que les tremblements de terre, les inondations ou les incendies. Le bâtiment a été construit avec du béton armé et des poutres en acier pour une protection supplémentaire contre les tremblements de terre. Il dispose également d'un système de détection automatique d'incendie, d'un système de gicleurs et de systèmes de ventilation séparés pour l'extraction de la fumée en cas d'incendie ou d'inhalation de fumée provenant de fuites chimiques.

III. Dictionnaire de données

a. Les entités

Produit		
Attribut	Type et taille	Exemple
IdProduit	int	23
typeProduit	VarChar(50)	Laitier

FicheDescriptive		
Attribut	Type et taille	Exemple
IdFicheD	int	12
poids	decimal(15,3)	12,548
volume	int	20
description	VarChar(250)	Produit laitier chocolat
dateLimiteConsommation	Date	20/03/2023

LotProduit		
Attribut	Type et taille	Exemple
IdLot	int(10)	93

DossierLot		
Attribut	Type et taille	Exemple
IdDossier	int(10)	83
dateEntree	Datetime	18/08/2021
dateRetrait	Datetime	28/07/2022
quantité	int	38

ConditionStockage		
Attribut	Type et taille	Exemple
IdConditionStockage	int	73
dureeConservation	int	15
temperatureMin	decimal(3,2)	6,50
temperatureMax	decimal(3,2)	12,50

CadreAdministratif		
Attribut	Type et taille	Exemple
IdCadreAdministratif	int	13
tempsConservation	int	9
temperatureMax	decimal(3,2)	6,50
temperatureMin	decimal(3,2)	12,50

Contrôle		
Attribut	Type et taille	Exemple
IdControle	int	65
origine	VarChar(50)	France
OGM	Logical	1
colorants	Logical	0
dateVerif	Date	20/11/2022
resultat	VarChar(250)	Résultat positif au test

commentaire	VarChar(250)	Rien à signaler sur ce produit
-------------	--------------	--------------------------------

TechnicienStockage		
Attribut	Type et taille	Exemple
IdTechnicien	int	12
nomTechnicien	VarChar(50)	Dupont
prenomTechnicien	VarChar(50)	Jean

FicheInventaire		
Attribut	Type et taille	Exemple
IdFicheI	int	45
dateInventaire	Date	20/04/2021
statut	VarChar(50)	Valide

b. Les relations

Arrive		
Entité 1	Entité 2	Description
Produit(1,n)	LotProduit(1,1)	Les produits arrivent en lot de produit

Possède		
Entité 1	Entité 2	Description
Produit(1,n)	FicheDescriptive(1,1)	Les produits possèdent une fiche descriptive

AvoirCondition		
Entité 1	Entité 2	Description
ConditionStockage(1,n)	LotProduit(1,1)	Les lots de produit ont des conditions de stockages

Réglementé		
Entité 1	Entité 2	Description
Produit(1,n)	CadreAdministratif(1,1)	Les produits sont réglementés par un cadre administratif

Précise		
Entité 1	Entité 2	Description
ConditionStockage(1,n)	CadreAdministratif(1,1)	Les cadres administratif précise les conditions de stockages

Prélève		
Entité 1	Entité 2	Description
TechnicienStockage(1,n)	LotProduit(1,n)	Les techniciens de stockage prélève des lots de produits

Rempli		
Entité 1	Entité 2	Description
TechnicienStockage(1,n)	FicheInventaire(1,1)	Les techniciens de stockages remplissent les fiches inventaires

Créer		
Entité 1	Entité 2	Description
FicheInventaire(1,1)	LotProduit(1,n)	Les lots de produits créé une fiche inventaire

Applique		
Entité 1	Entité 2	Description
LotProduit(1,n)	CadreAdministratif(1,1)	Un cadre administratif est appliqué au lot de produit

OuvrirDossier			
Entitée 1		Entitée 2	Description
LotProduit(1,n)		DossierLot(1,1)	Un lot de produit fait ouvrir un dossier de lot

AvoirControle		
Entitée 1	Entitée 2	Description
LotProduit(0,n)	Controle(1,1)	Un lot de produit à obligatoirement un contrôle

Définit		
Entitée 1	Entitée 2	Description
CadreAdministratif(1,n)	Controle(1,n)	Un contrôle est défini par un cadre administratif

IV. Modèle conceptuel de données

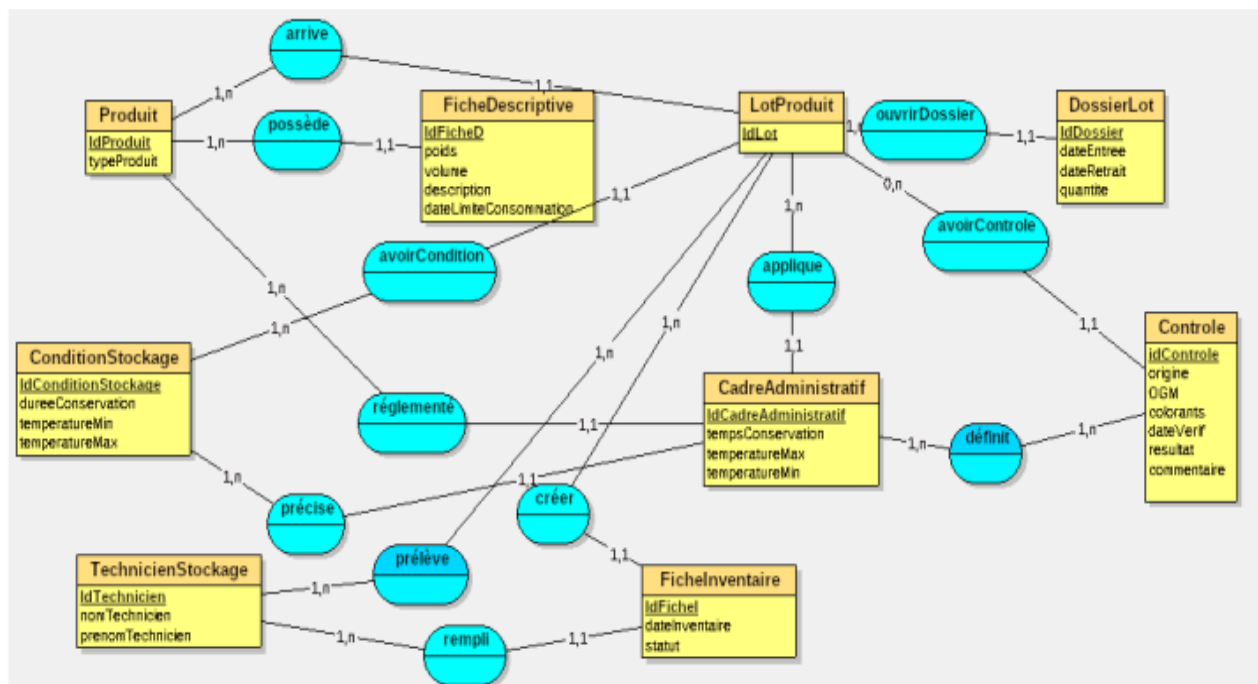


Figure 1 : Modèle conceptuel de données

V. Modèle relationnel

Produit = (IdProduit INT , typeProduit VARCHAR(50));

FicheDescriptive = (IdFicheD INT , poids DECIMAL(15,3) , volume INT , description VARCHAR(250) , dateLimiteConsommation DATE , #IdProduit);

ConditionStockage = (IdConditionStockage INT , dureeConservation INT , temperatureMin DECIMAL(3,2) , temperatureMax DECIMAL(3,2));

TechnicienStockage = (IdTechnicien INT , nomTechnicien VARCHAR(50) , prenomTechnicien VARCHAR(50));

LotProduit = (IdLot INT , #IdConditionStockage, #IdProduit);

DossierLot = (IdDossier INT , dateEntree DATE , dateRetrait DATE , quantite INT , #IdLot);

FicheInventaire = (IdFicheI INT , dateInventaire DATE , statut VARCHAR(50) , #IdTechnicien, #IdLot);

CadreAdministratif = (IdCadreAdministratif INT , tempsConservation INT , temperatureMax DECIMAL(3,2) , temperatureMin DECIMAL(3,2) , #IdConditionStockage, #IdLot, #IdProduit);

Contrôle = (idControle INT , origine VARCHAR(50) , OGM LOGICAL , colorants LOGICAL , dateVerif DATE , resultat VARCHAR(250) , commentaire VARCHAR(250) , #IdLot);

défini = (#IdCadreAdministratif, #idControle);

prélève = (#IdLot, #IdTechnicien);

VI. Modèle logique de données

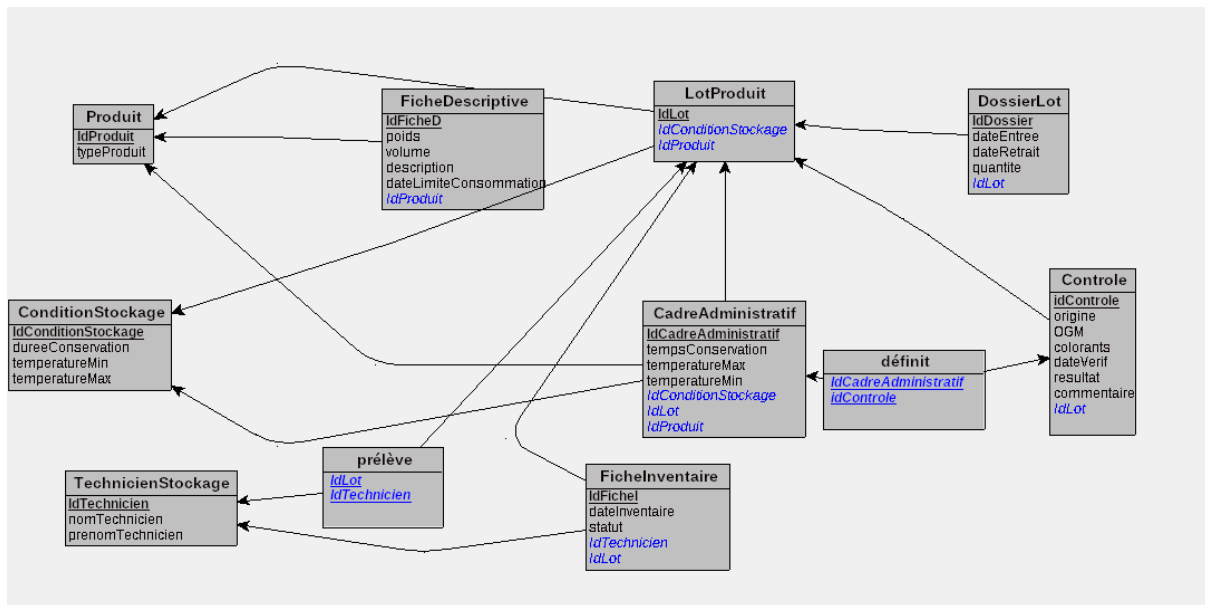


Figure 2 : Modèle logique de données

VII. Modèle conceptuel de traitement

a. L'inventaire

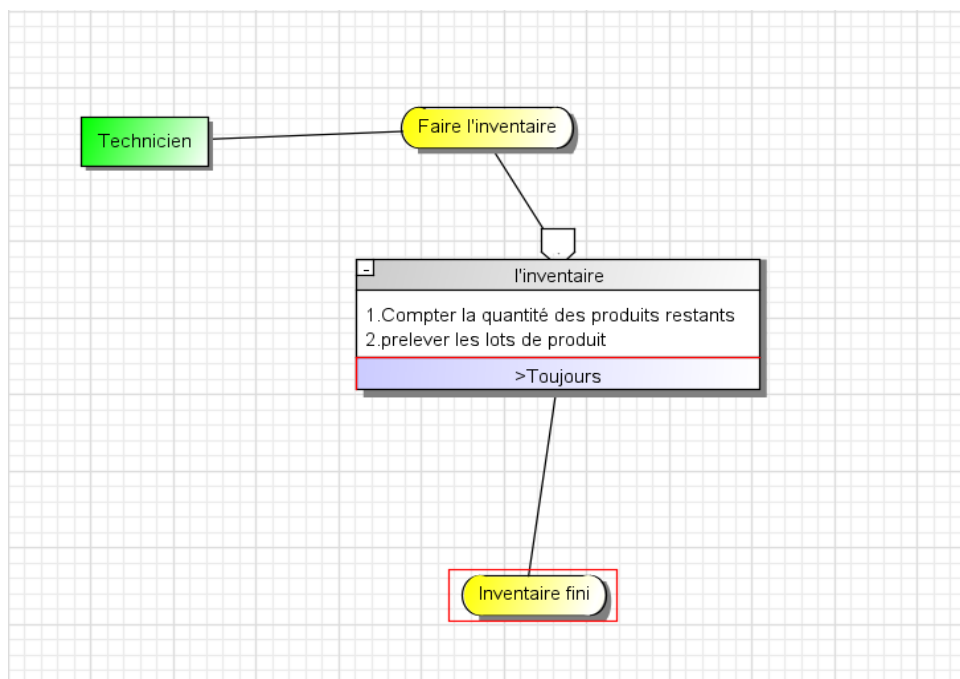


Figure 3 : Modèle conceptuel de traitement pour l'inventaire

c. Le contrôle

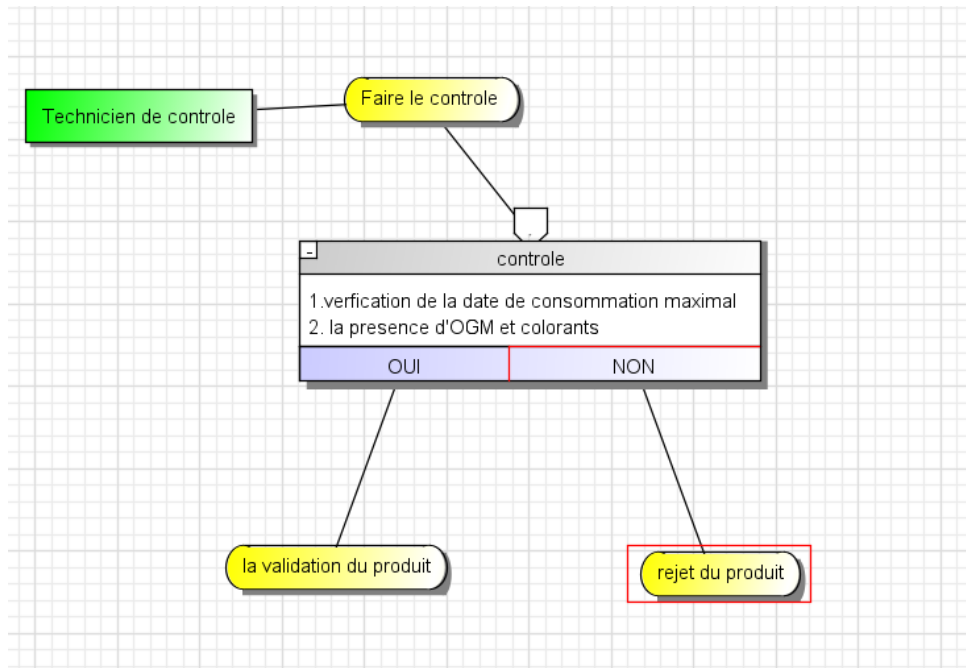


Figure 4 : Modèle conceptuel de traitement pour le contrôle

VIII. Normalisation

```

Produit(IdProduit, typeProduit)
Dépendances Fonctionnelles :
    F:{   IdProduit -> typeProduit
          typeProduit -> IdProduit
        }

calcul de la fermeture transitive de Produit : (Produit)+ pour
trouver la clé

(Produit)+ : typeProduit
    
```

Tous les attributs sont atomiques donc la relation est en 1NF.
 Aucun attribut non clé ne dépend d'une partie de la clé donc la relation est en 2NF.
 Aucun attribut qui n'appartient pas à la clé, ne dépend d'aucun autre attribut non-clé.
 La relation est donc en 3NF.

```

FicheDescriptive(IdFicheD, poids, volume, description,
dateLimiteConsommation)
Dépendances Fonctionnelles :
    F:{   IdFicheD -> poids
          IdFicheD -> volume
        }
    
```

```

        IdFicheD -> description
        IdFicheD -> dateLimiteConsommation
        poids, volume, description, dateLimiteConsommation ->
IdFicheD
    }

```

calcul de la fermeture transitive de FicheDescriptive :
(FicheDescriptive)+ pour trouver la clé

(IdFicheD)+ : poids, volume, description, dateLimiteConsommation

Tous les attributs sont atomiques donc la relation est en 1NF.

Aucun attribut non clé ne dépend d'une partie de la clé donc la relation est en 2NF.

Aucun attribut qui n'appartient pas à la clé, ne dépend d'aucun autre attribut non-clé.

La relation est donc en 3NF.

DossierLot(IdDossier, dateEntree, dateRetrait, quantité)

Dépendances Fonctionnelles :

```

F:{   IdDossier -> dateEntree
      IdDossier -> dateRetrait
      IdDossier -> quantité
      dateEntree, dateRetrait, quantité -> IdDossier
    }

```

calcul de la fermeture transitive de DossierLot : (DossierLot)+
pour trouver la clé

(DossierLot)+ : dateEntree, dateRetrait, quantité

Tous les attributs sont atomiques donc la relation est en 1NF.

Aucun attribut non clé ne dépend d'une partie de la clé donc la relation est en 2NF.

Aucun attribut qui n'appartient pas à la clé, ne dépend d'aucun autre attribut non-clé.

La relation est donc en 3NF.

ConditionStockage(IdConditionStockage, dureeConservation,
temperatureMin, temperatureMax)

Dépendances Fonctionnelles :

```

F:{   IdConditionStockage -> dureeConservation
      IdConditionStockage -> temperatureMin
      IdConditionStockage -> temperatureMax
      dureeConservation, temperatureMin, temperatureMax ->
IdConditionStockage
    }

```

calcul de la fermeture transitive de ConditionStockage :
(ConditionStockage)+ pour trouver la clé

```
(ConditionStockage)+ : dureeConservation, temperatureMin,  
temperatureMax
```

Tous les attributs sont atomiques donc la relation est en 1NF.

Aucun attribut non clé ne dépend d'une partie de la clé donc la relation est en 2NF.

Aucun attribut qui n'appartient pas à la clé, ne dépend d'aucun autre attribut non-clé.

La relation est donc en 3NF.

```
CadreAdministratif(IdCadreAdministratif, tempsConservation,  
temperatureMax, temperatureMin)
```

Dépendances Fonctionnelles :

```
F:{  IdCadreAdministratif -> tempsConservation  
      IdCadreAdministratif -> temperatureMax  
      IdCadreAdministratif -> temperatureMin  
      tempsConservation, temperatureMax, temperatureMin ->  
      IdCadreAdministratif  
}
```

calcul de la fermeture transitive de CadreAdministratif :
(CadreAdministratif)+ pour trouver la clé

```
(CadreAdministratif)+ : tempsConservation, temperatureMax,  
temperatureMin
```

Tous les attributs sont atomiques donc la relation est en 1NF.

Aucun attribut non clé ne dépend d'une partie de la clé donc la relation est en 2NF.

Aucun attribut qui n'appartient pas à la clé, ne dépend d'aucun autre attribut non-clé.

La relation est donc en 3NF.

```
Contrôle(idContrôle, origine, OGM, colorants, dateVerif, resultat,  
commentaire)
```

Dépendances Fonctionnelles :

```
F:{  idContrôle -> origine  
      idContrôle -> OGM  
      idContrôle -> colorants  
      idContrôle -> dateVerif  
      idContrôle -> resultat  
      idContrôle -> commentaire  
      origine, OGM, colorants, dateVerif, resultat,  
      commentaire -> idContrôle  
}
```

calcul de la fermeture transitive de Contrôle : (Contrôle)+ pour
trouver la clé

```
(Contrôle)+ : origine, OGM, colorants, dateVerif, resultat,  
commentaire
```

Tous les attributs sont atomiques donc la relation est en 1NF.
Aucun attribut non clé ne dépend d'une partie de la clé donc la relation est en 2NF.
Aucun attribut qui n'appartient pas à la clé, ne dépend d'aucun autre attribut non-clé.
La relation est donc en 3NF.

TechnicienStockage(IdTechnicien, nomTechnicien, prenomTechnicien)

Dépendances Fonctionnelles :

```
F:{  IdTechnicien -> nomTechnicien
      IdTechnicien -> prenomTechnicien
      nomTechnicien, prenomTechnicien -> IdTechnicien
}
```

calcul de la fermeture transitive de TechnicienStockage :
(TechnicienStockage)+ pour trouver la clé

(TechnicienStockage)+ : nomTechnicien, prenomTechnicien

Tous les attributs sont atomiques donc la relation est en 1NF.
Aucun attribut non clé ne dépend d'une partie de la clé donc la relation est en 2NF.
Aucun attribut qui n'appartient pas à la clé, ne dépend d'aucun autre attribut non-clé.
La relation est donc en 3NF.

FicheInventaire(IdFicheI, dateInventaire, statut)

Dépendances Fonctionnelles :

```
F:{  IdFicheI -> dateInventaire
      IdFicheI -> statut
      dateInventaire, statut -> IdFicheI
}
```

calcul de la fermeture transitive de FicheInventaire :
(FicheInventaire)+ pour trouver la clé

(FicheInventaire)+ : dateInventaire, statut

Tous les attributs sont atomiques donc la relation est en 1NF.
Aucun attribut non clé ne dépend d'une partie de la clé donc la relation est en 2NF.
Aucun attribut qui n'appartient pas à la clé, ne dépend d'aucun autre attribut non-clé.
La relation est donc en 3NF.

Toutes les relations sont en 3NF donc nous pouvons dire que notre base de données est en 3ème forme normale

IX. Export de la base de données

a. Création des tables

```

CREATE TABLE Produit
(
    Idproduit NUMBER(10) PRIMARY KEY,
    typeProduit VARCHAR2(50) NOT NULL
);
CREATE TABLE FicheDescriptive
(
    IdFicheD NUMBER(10) PRIMARY KEY,
    IdProduit NUMBER(10) NOT NULL,
    poids NUMBER(15,3) NOT NULL,
    volume NUMBER(10) NOT NULL,
    dateLimiteConsommation DATE NOT NULL,
    description_ VARCHAR2(250) NOT NULL
);

Create table LotProduit
(
    IdLot NUMBER(10) PRIMARY KEY,
    IdProduit NUMBER(10) NOT NULL,
    IdConditionStockage NUMBER(10) NOT NULL
);
create table DossierLot
(
    IdDossier NUMBER(10) PRIMARY KEY,
    IdLot NUMBER(10) NOT NULL,
    dateEntree DATE NOT NULL,
    dateRetrait DATE NOT NULL,
    quantite NUMBER(10) NOT NULL
);
Create table ConditionStockage
(
    IdConditionStockage NUMBER(10) PRIMARY KEY,
    dureeConservation NUMBER(10) NOT NULL,
    temperatureMin NUMBER(10) NOT NULL,
    temperatureMax NUMBER(10) NOT NULL
);

create table FicheInventaire
(
    IdFicheI NUMBER(10) PRIMARY KEY,
    IdLot NUMBER(10) NOT NULL,
    dateInventaire DATE NOT NULL,
    statut VARCHAR2(50) NOT NULL,
    IdTechnicien NUMBER(10) NOT NULL
);
create table CadreAdministratif
(
    IdCadreAdministratif NUMBER(10) PRIMARY KEY,
    IdLot NUMBER(10) NOT NULL,
    IdCondition NUMBER(10) NOT NULL,
    IdProduit NUMBER(10) NOT NULL,

```



```

        tempsConservation NUMBER(10) NOT NULL,
        temperatureMax NUMBER(10) NOT NULL,
        temperatureMin NUMBER(10) NOT NULL
    );
    Create table Controle
    (
        IdControle NUMBER(10) PRIMARY KEY,
        IdLot NUMBER(10) NOT NULL,
        origine VARCHAR2(50) NOT NULL,
        OGM NUMBER(1) NOT NULL,
        colorants NUMBER(1) NOT NULL,
        dateVerif DATE NOT NULL,
        resultat VARCHAR2(250) NOT NULL,
        commentaire VARCHAR2(250) NOT NULL
    );
    create table TechnicienStockage
    (
        IdTechnicien NUMBER(10) PRIMARY KEY,
        nomTechnicien VARCHAR2(50) NOT NULL,
        prenomTechnicien VARCHAR2(50) NOT NULL
    );
    create table definit
    (
        IdCadreAdministratif NUMBER(10) NOT NULL,
        IdControle NUMBER(10) NOT NULL
    );
    create table preleve
    (
        IdLot NUMBER(10) NOT NULL,
        IdTechnicien NUMBER(10) NOT NULL
    );

```

b. L'ajout des clés étrangères

```

Alter table LotProduit add constraint fk_lotproduit_produit foreign key
(IdProduit) references Produit(IdProduit);

Alter table LotProduit add constraint fk_lotproduit_conditionstockage
foreign key (IdConditionStockage) references
ConditionStockage(IdConditionStockage);

Alter table DossierLot add constraint fk_dossierlot_lotproduit foreign
key (IdLot) references LotProduit(IdLot);

Alter table FicheInventaire add constraint fk_ficheinventaire_lotproduit
foreign key (IdLot) references LotProduit(IdLot);

Alter table FicheInventaire add constraint
fk_ficheinventaire_technicienstockage foreign key (IdTechnicien)
references TechnicienStockage(IdTechnicien);

```

```

Alter table CadreAdministratif add constraint
fk_cadreadministratif_lotproduit foreign key (IdLot) references
LotProduit(IdLot);

Alter table CadreAdministratif add constraint
fk_cadreadministratif_conditionstockage foreign key (IdCondition)
references ConditionStockage(IdCondition);

Alter table CadreAdministratif add constraint
fk_cadreadministratif_produit foreign key (IdProduit) references
Produit(IdProduit);

Alter table Controle add constraint fk_controle_lotproduit foreign key
(IdLot) references LotProduit(IdLot);

Alter table definit add constraint fk_definit_cadreadministratif foreign
key (IdCadreAdministratif) references
CadreAdministratif(IdCadreAdministratif);

Alter table definit add constraint fk_definit_controle foreign key
(IdControle) references Controle(IdControle);

Alter table preleve add constraint fk_preleve_lotproduit foreign key
(IdLot) references LotProduit(IdLot);

Alter table preleve add constraint fk_preleve_technicienstockage foreign
key (IdTechnicien) references TechnicienStockage(IdTechnicien);

```

d. insertion des données

```

INSERT INTO Produit VALUES (1, 'Fruits');
INSERT INTO Produit VALUES (2, 'Légumes');
INSERT INTO Produit VALUES (3, 'Viandes');
INSERT INTO Produit VALUES (4, 'Laitages');

INSERT INTO FicheDescriptive VALUES
(1, 1, 100, 120, to_date('21/01/2022', 'dd/mm/yyyy'), 'Orange de marque Belle
France');
INSERT INTO FicheDescriptive VALUES
(5, 1, 80, 120, to_date('12/02/2022', 'dd/mm/yyyy'), 'pomme de marque Belle
France');
INSERT INTO FicheDescriptive VALUES
(2, 2, 80, 100, to_date('18/02/2022', 'dd/mm/yyyy'), 'pomme de terre de marque
Belle France');
INSERT INTO FicheDescriptive VALUES
(6, 2, 80, 100, to_date('03/02/2022', 'dd/mm/yyyy'), 'Carotte de marque Belle
France');
INSERT INTO FicheDescriptive VALUES
(3, 3, 100, 120, to_date('30/11/2022', 'dd/mm/yyyy'), 'poulet de marque Belle
France');

```

```

INSERT INTO FicheDescriptive VALUES
(7,3,100,120,to_date('07/11/2022','dd/mm/yyyy'),'veau de marque Belle
France');
INSERT INTO FicheDescriptive VALUES
(4,4,100,120,to_date('18/01/2024','dd/mm/yyyy'),'Fromage cheddar de
marque la vache qui rit');
INSERT INTO FicheDescriptive VALUES
(8,4,100,120,to_date('08/04/2023','dd/mm/yyyy'),'Yaourt de marque la
vache qui rit');

INSERT INTO ConditionStockage VALUES (1,21,5,15);
INSERT INTO ConditionStockage VALUES (2,90,5,20);
INSERT INTO ConditionStockage VALUES (3,360,-10,5);
INSERT INTO ConditionStockage VALUES (4,250,5,15);
INSERT INTO ConditionStockage VALUES (5,21,5,15);
INSERT INTO ConditionStockage VALUES (6,90,5,20);
INSERT INTO ConditionStockage VALUES (7,360,-10,5);
INSERT INTO ConditionStockage VALUES (8,600,5,15);

INSERT INTO LotProduit VALUES (1,1,1);
INSERT INTO LotProduit VALUES (2,2,2);
INSERT INTO LotProduit VALUES (3,3,3);
INSERT INTO LotProduit VALUES (4,4,4);
INSERT INTO LotProduit VALUES (5,1,5);
INSERT INTO LotProduit VALUES (6,2,6);
INSERT INTO LotProduit VALUES (7,3,7);
INSERT INTO LotProduit VALUES (8,4,8);

--on mis que la date de retrait c'est la date limite de consommation
INSERT INTO DossierLot VALUES
(1,1,to_date('01/01/2022','dd/mm/yyyy'),to_date('21/01/2022','dd/mm/yyyy')
),120);
INSERT INTO DossierLot VALUES
(2,2,to_date('01/12/2021','dd/mm/yyyy'),to_date('18/02/2022','dd/mm/yyyy')
),180);
INSERT INTO DossierLot VALUES
(3,3,to_date('13/01/2022','dd/mm/yyyy'),to_date('07/11/2022','dd/mm/yyyy')
),100);
INSERT INTO DossierLot VALUES
(4,4,to_date('06/06/2022','dd/mm/yyyy'),to_date('08/04/2023','dd/mm/yyyy')
),100);
INSERT INTO DossierLot VALUES
(5,5,to_date('01/01/2022','dd/mm/yyyy'),to_date('12/02/2022','dd/mm/yyyy')
),120);
INSERT INTO DossierLot VALUES
(6,6,to_date('01/12/2021','dd/mm/yyyy'),to_date('03/02/2022','dd/mm/yyyy')
),180);
INSERT INTO DossierLot VALUES
(7,7,to_date('13/01/2022','dd/mm/yyyy'),to_date('07/11/2022','dd/mm/yyyy')
),100);
INSERT INTO DossierLot VALUES
(8,8,to_date('06/06/2022','dd/mm/yyyy'),to_date('08/04/2023','dd/mm/yyyy')
),100);

INSERT INTO TechnicienStockage VALUES (1,'Doe','John');
INSERT INTO TechnicienStockage VALUES (2,'Topa','Caline');
INSERT INTO TechnicienStockage VALUES (3,'Dupont','Quentin');

```

```

INSERT INTO FicheInventaire VALUES
(1,1,to_date('01/01/2022','dd/mm/yyyy'),'OK',1);
INSERT INTO FicheInventaire VALUES
(2,2,to_date('01/12/2021','dd/mm/yyyy'),'OK',1);
INSERT INTO FicheInventaire VALUES
(3,3,to_date('13/01/2022','dd/mm/yyyy'),'OK',1);
INSERT INTO FicheInventaire VALUES
(4,4,to_date('06/06/2022','dd/mm/yyyy'),'OK',2);
INSERT INTO FicheInventaire VALUES
(5,5,to_date('01/01/2022','dd/mm/yyyy'),'OK',2);
INSERT INTO FicheInventaire VALUES
(6,6,to_date('01/12/2021','dd/mm/yyyy'),'OK',2);
INSERT INTO FicheInventaire VALUES
(7,7,to_date('13/01/2022','dd/mm/yyyy'),'OK',3);
INSERT INTO FicheInventaire VALUES
(8,8,to_date('06/06/2022','dd/mm/yyyy'),'OK',3);
INSERT INTO FicheInventaire VALUES
(9,4,to_date('06/06/2022','dd/mm/yyyy'),'OK',1);

INSERT INTO CadreAdministratif VALUES (1,1,1,1,21,15,5);
INSERT INTO CadreAdministratif VALUES (2,2,2,2,90,20,5);
INSERT INTO CadreAdministratif VALUES (3,3,3,3,360,5,-10);
INSERT INTO CadreAdministratif VALUES (4,4,4,4,250,15,5);
INSERT INTO CadreAdministratif VALUES (5,5,1,1,21,15,5);
INSERT INTO CadreAdministratif VALUES (6,6,2,2,90,20,5);
INSERT INTO CadreAdministratif VALUES (7,7,3,3,360,5,-10);
INSERT INTO CadreAdministratif VALUES (8,8,4,4,600,15,5);

INSERT INTO Controle VALUES
(1,1,'Maroc',0,0,to_date('20/01/2022','dd/mm/yyyy'),'OK','les oranges
sont de bonne qualité');
INSERT INTO Controle VALUES
(2,2,'Italie',0,0,to_date('20/02/2022','dd/mm/yyyy'),'KO','Il y a des
légumes abimés ils sont deppassés la date limite');
INSERT INTO Controle VALUES
(3,3,'Espagne',1,0,to_date('07/07/2022','dd/mm/yyyy'),'OK','les viandes
sont de bonne qualité mais le poulet est OGM');
INSERT INTO Controle VALUES
(4,4,'France',0,0,to_date('01/01/2022','dd/mm/yyyy'),'OK','le fromage est
de bonne qualité');
INSERT INTO Controle VALUES
(5,5,'Ukraine',1,1,to_date('20/01/2022','dd/mm/yyyy'),'OK','les pommes
sont de bonne qualité mais elles sont OGM et elles contient des
colorants');
INSERT INTO Controle VALUES
(6,6,'Italie',0,0,to_date('20/02/2022','dd/mm/yyyy'),'KO','Il y a des
légumes abimés ils sont deppassés la date limite');
INSERT INTO Controle VALUES
(7,7,'Espagne',1,0,to_date('07/07/2022','dd/mm/yyyy'),'OK','le veau
contient 1 OGM');
INSERT INTO Controle VALUES
(8,8,'France',0,1,to_date('01/01/2022','dd/mm/yyyy'),'OK','le yaourt
contient des colorants');
INSERT INTO Controle VALUES
(9,4,'France',0,0,to_date('25/03/2022','dd/mm/yyyy'),'OK','le fromage est
encore bon !!');

```

```
INSERT INTO definit VALUES (1,1);
INSERT INTO definit VALUES (2,2);
INSERT INTO definit VALUES (1,3);
INSERT INTO definit VALUES (2,4);
INSERT INTO definit VALUES (1,5);
INSERT INTO definit VALUES (2,6);
INSERT INTO definit VALUES (6,7);
INSERT INTO definit VALUES (8,8);
```

```
INSERT INTO preleve VALUES (1,1);
INSERT INTO preleve VALUES (2,2);
INSERT INTO preleve VALUES (3,1);
INSERT INTO preleve VALUES (4,1);
INSERT INTO preleve VALUES (5,2);
INSERT INTO preleve VALUES (6,2);
INSERT INTO preleve VALUES (7,3);
INSERT INTO preleve VALUES (8,3);
```

```
select * from produit;
select * from lotProduit;
select * from DossierLot;
select * from TechnicienStockage;
select * from FicheInventaire;
select * from CadreAdministratif;
select * from Controle;
select * from definit;
select * from preleve;
```

e. suppression des tables

```
DROP TABLE produit;
DROP TABLE lot;
DROP TABLE DossierLot;
DROP TABLE TechnicienStockage;
DROP TABLE FicheInventaire;
DROP TABLE CadreAdministratif;
DROP TABLE Controle;
DROP TABLE definit;
DROP TABLE preleve;
```

X. Requêtes SQL

--Le dossier de lot des produits d'origine française où il n'y a pas de présences d'OGM ni de colorants, et le nom du technicien ayant fait l'inventaire est John Doe.

```
SELECT *
FROM DossierLot
WHERE IdLot IN (SELECT IdLot FROM Controle WHERE origine='France' AND
OGM=0 AND colorants=0)
AND IdLot IN (SELECT IdLot FROM FicheInventaire WHERE IdTechnicien IN
(SELECT IdTechnicien FROM TechnicienStockage WHERE nomTechnicien='Doe'
AND prenomTechnicien='John'));
```

-- La fiche descriptive des produits dont la température Max de stockage est supérieure à 10°C et la durée de conservation est inférieure à 200 jours par ordre décroissant de la date limite de consommation.

```
SELECT *
FROM FicheDescriptive
WHERE IdProduit IN (SELECT IdProduit
FROM LotProduit
where IdConditionStockage IN (
SELECT IdConditionStockage
FROM ConditionStockage
WHERE temperatureMax >10 AND dureeConservation<200));
```

-- La liste des produit dont la durée de conservation est supérieure à la durée de conservation moyenne des produits et la températureMin de stockage est inférieure à la températureMin de stockage moyenne des produits.

```
SELECT *
FROM FicheDescriptive
WHERE IDProduit
IN (SELECT IdProduit
FROM LotProduit
WHERE IdConditionStockage IN (
SELECT IdConditionStockage
FROM ConditionStockage
WHERE dureeConservation > (SELECT
AVG(dureeConservation) FROM ConditionStockage)
AND temperatureMin <
(SELECT AVG(temperatureMin) FROM ConditionStockage)));
```

--le nom de technicien qui a fait l'inventaire d'un produit qui est un OGM et qui contient des colorants

```
SELECT nomTechnicien,prenomTechnicien
FROM TechnicienStockage
WHERE IdTechnicien IN (SELECT IdTechnicien
FROM FicheInventaire
WHERE IdLot IN (SELECT IdLot
FROM Controle
```

```
WHERE OGM=1 AND
colorants=1));
```

-- La fiche descriptive des produits qui reste plus de 50 jours avant la date limite de consommation par rapport à la date actuelle et qui ont un contrôle OK.

```
SELECT *
FROM FicheDescriptive
WHERE IDProduit IN (SELECT IdProduit
                    FROM LotProduit
                    WHERE IdLot IN (SELECT IdLot
                                    FROM Controle
                                    WHERE dateLimiteConsommation >
SYSDATE+50 AND resultat='OK'));
```

--Les nom des techniciens de stockage commençant par un 'T' ayant fait l'inventaire d'un produit qui est un OGM et qui contient des colorants.

```
SELECT nomTechnicien,prenomTechnicien
FROM TechnicienStockage
WHERE IdTechnicien IN (SELECT IdTechnicien
                      FROM FicheInventaire
                      WHERE IdLot IN (SELECT IdLot
                                      FROM Controle
                                      WHERE OGM=1 AND
colorants=1))
AND nomTechnicien LIKE 'T%';
```

XI. Programme PL/SQL

-- Procédure pl/sql qui utilise un curseur calculant le pourcentage de produits qui ont un contrôle OK par rapport au nombre total de produits.

```
CREATE OR REPLACE PROCEDURE pourcentageProduitOK
IS
    CURSOR c1 IS
    SELECT count(*)
    FROM Controle
    WHERE resultat='OK';
    nbProduitOK NUMBER;
    nbProduit NUMBER;
BEGIN
    SELECT COUNT(*) INTO nbProduit FROM Controle;
    OPEN c1;
    LOOP
```

```

    FETCH c1 INTO nbProduitOK;
    EXIT WHEN c1%NOTFOUND;
    END LOOP;
    DBMS_OUTPUT.PUT_LINE('le pourcentage de produits qui ont un
    controle OK est : '||nbProduitOK/nbProduit*100||'%');
END pourcentageProduitOK;

call pourcentageProduitOK

```

--Un curseur qui parcourt les produits qui ont un contrôle KO et qui affiche le nom du technicien ayant fait l'inventaire.

```

DECLARE
    CURSOR c1 IS
    SELECT nomTechnicien,prenomTechnicien
    FROM TechnicienStockage
    WHERE IdTechnicien IN (SELECT IdTechnicien
                           FROM FicheInventaire
                           WHERE IdLot IN (SELECT IdLot
                                           FROM Controle
                                           WHERE
resultat='KO'));
    nomTechnicien_c VARCHAR2(20);
    prenomTechnicien_c VARCHAR2(20);
BEGIN
    OPEN c1;
    LOOP
    FETCH c1 INTO nomTechnicien_c,prenomTechnicien_c;
    EXIT WHEN c1%NOTFOUND;
    DBMS_OUTPUT.PUT_LINE('le nom du technicien qui a fait
    l''inventaire est : '||nomTechnicien_c||' '||prenomTechnicien_c);
    END LOOP;
END ;

```

--Un trigger qui vérifie que la date de retrait n'est pas nulle sinon il met une valeur par défaut qui ne peut pas être antérieure à la date d'entrée.

```

CREATE OR REPLACE TRIGGER dateRetrait
BEFORE INSERT ON DossierLot
FOR EACH ROW
BEGIN
    IF :new.dateRetrait < :new.dateEntree THEN
    DBMS_OUTPUT.PUT_LINE('la date retrait ne peut pas être antérieure
    à la date d''entrée on a mis 31/12/2100 par défaut');
    :new.dateRetrait:=to_date('31/12/2100','dd/mm/yyyy');
    END IF;
END dateRetrait;

insert into DossierLot
values (9,4,to_date('01/01/2019','dd/mm/yyyy'),to_date('01/01/2018','dd/mm
/yyyy'),10);

```


--Une fonction qui prend en paramètre une température et qui retourne la température en degré Fahrenheit

```
CREATE OR REPLACE FUNCTION tempFahrenheit(temp NUMBER)
RETURN NUMBER
IS
    tempF NUMBER;
BEGIN
    tempF:=temp*1.8+32;
    RETURN tempF;
END tempFahrenheit;

DECLARE
    temp NUMBER;
BEGIN
    temp:=tempFahrenheit(10);
    DBMS_OUTPUT.PUT_LINE('la température en degré Fahrenheit est :
'||temp);
END;
```

-- Une procédure permettant pour chaque produit, d'obtenir la température de stockage maximal.

```
CREATE OR REPLACE Procedure tempMaxProduit
IS
    CURSOR c1 IS
    SELECT temperatureMax
    FROM ConditionStockage
    WHERE IdConditionStockage IN (SELECT IdConditionStockage
                                FROM LotProduit
                                WHERE IdProduit IN (SELECT
                                                    FROM
                                                    FicheDescriptive));
    temp NUMBER;
BEGIN
    OPEN c1;
    LOOP
    FETCH c1 INTO temp;
    EXIT WHEN c1%NOTFOUND;
    DBMS_OUTPUT.PUT_LINE('la temperature de stockage maximal des
produits est : '||temp);
    END LOOP;
END tempMaxProduit;

call tempMaxProduit
```

Conclusion

Cette section de conclusion est la dernière partie de notre projet de conception de base de données. Dans cette section, nous résumerons les points clés que nous souhaitons mettre en évidence et fournirons une conclusion sur la manière de concevoir une base de données pour un entrepôt.

Dans ce projet, nous avons appris que la conception d'une base de données pour l'entrepôt nécessite de nombreuses considérations. Pour s'assurer que l'entrepôt n'a aucun problème à gérer son système de stockage, il est important de considérer tous les types de données et la manière dont elles sont utilisées. Nous vous recommandons d'utiliser des bases de données relationnelles car elles sont faciles à utiliser et sont très flexibles avec les changements d'exigences.