

Analyse et conception

3.1 Introduction

Dans ce chapitre, nous aborderons une description générale de notre application selon la démarche de la méthode générique de développement de logiciel ou Processus Unifié (UP).

La phase de conception du système est une étape fondamentale précède l'implémentation, elle permet d'acquérir la description technique d'application tel que : les détails des divers diagrammes, scénarios à implémenter dans la phase suivante, l'identification des acteurs, ...

Le rôle de la phase conception est de définir les choix techniques d'un projet d'une manière compréhensible par les développeurs afin de les implémenter.

Cette phase dans le cadre de notre étude contient la conception de l'entrepôt de données actif qui compose certaines étapes déterminées comme suite :

- Analyser les sources de données;
- Extraire la collection des données nécessaires pour stocker dans l'entrepôt de données;
- Concevoir un modèle multidimensionnel pour organiser les données extrait;

- Déterminer l'ensemble des règles d'analyses qui répondent aux besoins décisionnel.

3.2 Domaine d'application

Au cours de l'étude de cas que nous présenterons, nous étudions l'efficacité des règles d'analyse au sein d'un entrepôt de données actif pour contrôler la qualité des services de processus de livraison des produits à distance.

3.2.1 Processus de Livraison à distance

Le concept de l'action de "livrer" est de remettre quelque chose à quelqu'un dans un moment du temps.

La livraison est le fait de transporter des produits par un réseau de transport (par terre, air ou mer) à l'aide des véhicules spécialisés dans le transport d'objets, et d'assurer leur acheminement jusqu'à destination. La personne responsable de la livraison des produits est le chauffeur-livreur. Ce qui suit, nous mentionnons les tâches de routine qu'un chauffeur-livreur est censé effectuer pour assurer le flux de travail.

Afin que le chauffeur-livreur prépare sa tournée pour satisfaire les engagements qui concerne la livraison il doit :

- ◇ Respecter les règles définies et les consignes nécessaires tel que : le temps de contacter avec le fournisseur, les documents nécessaires qu'il doit avoir en sa possession, ...
- ◇ S'assure de garantir le bon état des produits pendant : le chargement de son véhicule, le transport et la sécurité de son déplacement.
- ◇ Respecter les règles de conditionnement : camion réfrigéré, ...
- ◇ Respecter les règles du code de la route : vitesse, documents, poids total, ...
- ◇ Effectuer toutes les formalités douanières nécessaires aux cas de passage aux frontières (transport international).
- ◇ Contrôler que le produit remis est conforme à la commande (la qualité, la quantité et les documents d'accompagnement, ...).

- ◇ Informé le client d'avance en cas de difficulté de livraison (un retard ou une anomalie).
- ◇ L'entretien de son véhicule.
- ◇ Si le chauffeur-livreur ne peut pas régler un problème seul, il doit informer les responsables pour la prise des mesures correctrices : cas d'un litige avec le client, un doute à l'entretien de véhicule, ...

3.2.2 Problèmes

Les principaux problèmes auxquels peut être confronté le processus de livraison des produits sont :

- Le retard de livraison et non-respect de la date convenue;
- Indisponibilité du produit, soit temporairement, ou bien la production du produit a été définitivement arrêtée par le fabricant;
- La qualité des produits : les achats détériorés lors de la livraison, des avaries sont découvertes (humidité, fuites, casses, pourritures...);
- La quantité reçue est supérieure ou inférieure à la commande;
- Double livraison : le chauffeur-livreur a fait une livraison double par erreur;
- Un litige entre le client et le livreur;
- Le produit livré n'est pas conforme à la commande effectuée. Le produit est différent;
- Le client a rétracté sa commande et n'a pas signalé l'annulation.

3.3 Architecture générale du système

Le schéma suivant montre l'architecture générale du système que nous avons proposé.

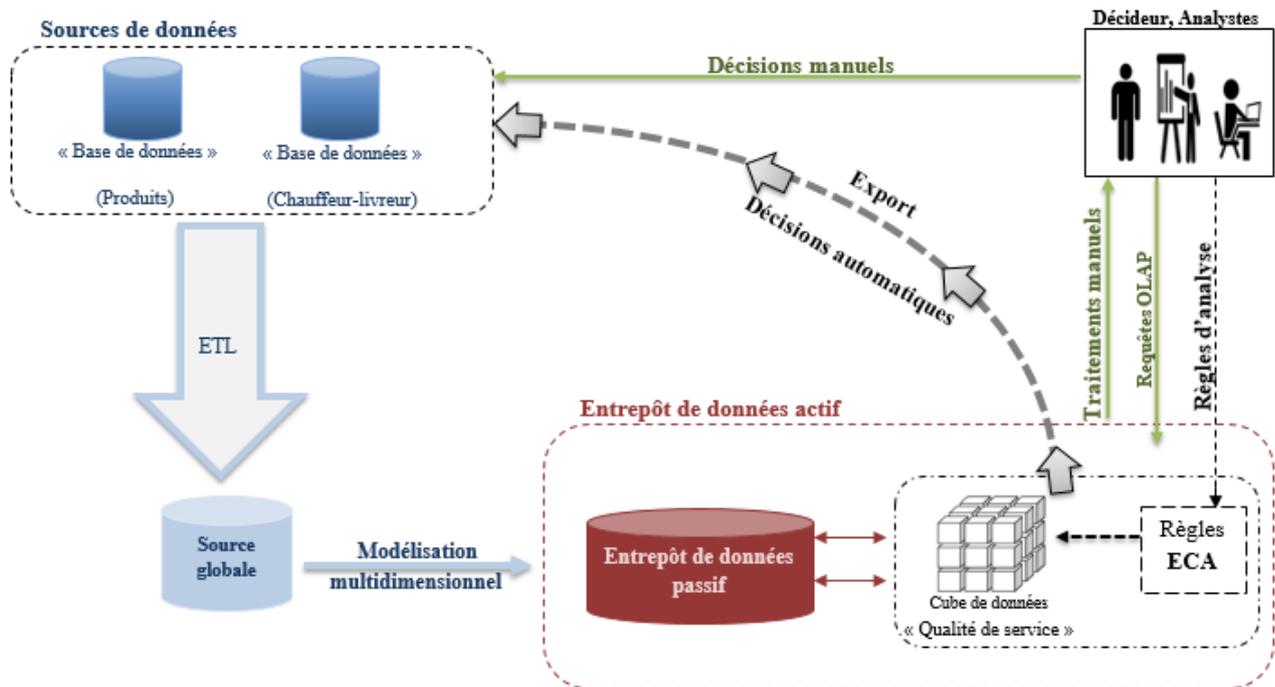


Figure 3.1: L'architecture générale du système.

- **Sources de données:** Pour collecter les données nécessaires à notre analyse, nous avons utilisé deux bases de données sont : la base de données « produits » et la base de données « chauffeur-livreur ».
- **Source globale :** c'est une source virtuelle de données contient l'intégration des sources de données utilisent à la construction de l'entrepôt de données.
- **L'entrepôt de données actif** construit par :
 - ◇ **Un entrepôt de données (passif) :** qui contient les données manipulées et représentées sous forme multidimensionnelle.
 - ◇ **Un cube de données :** le but de ce cube est d'organiser les données qui déterminent le mesure d'intérêt pour l'analyse en une ou plusieurs dimensions.
 - ◇ **Règles ECA :** ce sont les règles d'analyse qui seront utilisées pour analyser les données du cube.
- **Décisions automatiques :** tâches de décision routine.
- **Décisions manuels :** tâches de décision non-routine.

3.4 Spécification des besoins

Lors de cette phase, on va présenter les besoins fonctionnels ainsi que les besoins non fonctionnels pour bien exprimer les tâches à réaliser et aussi de comprendre les fonctionnalités que le système va fournir.

3.4.1 Besoins non fonctionnels

Les besoins non fonctionnels concernent les contraintes et les choix dimensionnant la conception du système, ils concernent d'une manière indirecte le résultat et le rendement de l'utilisateur.

Nous intéressons de répondre aux exigences suivantes :

- **Temps de livraison** : Le chauffeur-livreur doit respecter les délais nécessaires pour livrer la commande selon le convenu.
- **Qualité du produit** : veillez à sécuriser le transport des produits pour éviter tout dommage pendant le processus de transport.
- **Disponibilité du produit** : Signaler la pénurie de produits sur le marché de la production avec la possibilité de les remplacer par un produit similaire ou de l'abandonner complètement.
- **Disponibilité des chauffeurs-livreurs** : prendre en compte les commandes et s'assurer de les livrer avant délais, en plus de s'assurer que le nombre de livreur est proportionnel au nombre de demandes pour éviter les cas de demandes sans chauffeur livreur ou chauffeur livreur inactif depuis longtemps.
- **Quantité de produits** : engagement sur la quantité requise sans augmentation ni diminution.
- **Client inactif** : signaler les clients inactifs depuis une période donnée.

3.4.2 Besoins fonctionnels

Il s'agit des besoins fonctionnels spécifiant un comportement d'entrée /sortie du système. On déduit d'après notre étude les besoins fonctionnels suivants :

- ◇ Gérer les chauffeurs-livreurs (Ajouter, Modifier, Notifier, Supprimer).
- ◇ Gérer les clients (Ajouter, Modifier, Notifier, Supprimer).
- ◇ Gérer les produits (Ajouter, Modifier, Supprimer).
- ◇ Consulter l'état des bases de données source.

3.5 Description du contexte

Cette étape comporte les activités suivantes :

- Identification des acteurs.
- Réalisation des diagrammes de contexte.

3.5.1 Identification des acteurs

Un acteur est un élément actif qui interagit et joue un rôle (prend des décisions et des initiatives) par rapport au système. Le tableau ci-dessous identifier les acteurs de notre système et leurs rôles principaux.

Acteur	Rôle
Administrateur (Décideur)	<ul style="list-style-type: none"> ▪ Il gère les chauffeurs livreurs ainsi que les clients et les produits. ▪ Améliorez la qualité du service.
Chauffeur-livreur	<ul style="list-style-type: none"> ▪ Préparer une tournée intégré pour assurer la qualité du travail.
Client	<ul style="list-style-type: none"> ▪ Passer une commande.

Table 3.1: Rôles principaux des acteurs du système.

3.5.2 Réalisation des diagrammes de contexte

Les diagrammes permettre la modélisation des projets. On utilise deux types de diagramme pour la modélisation de notre application, le diagramme de cas d'utilisation et le diagramme de classe.

3.5.2.1 Diagrammes de cas d'utilisation

Le diagramme de cas d'utilisation est un diagramme de comportement nous permet de montrer l'interaction entre les acteurs et le système et de représenter toutes les fonctionnalités que le système doit fournir [28]. Nous allons présenter trois diagramme des cas d'utilisation, Administrateur, Produits(Clients) et Chauffeurs_livreur (Chauffeur_livreur et Client).

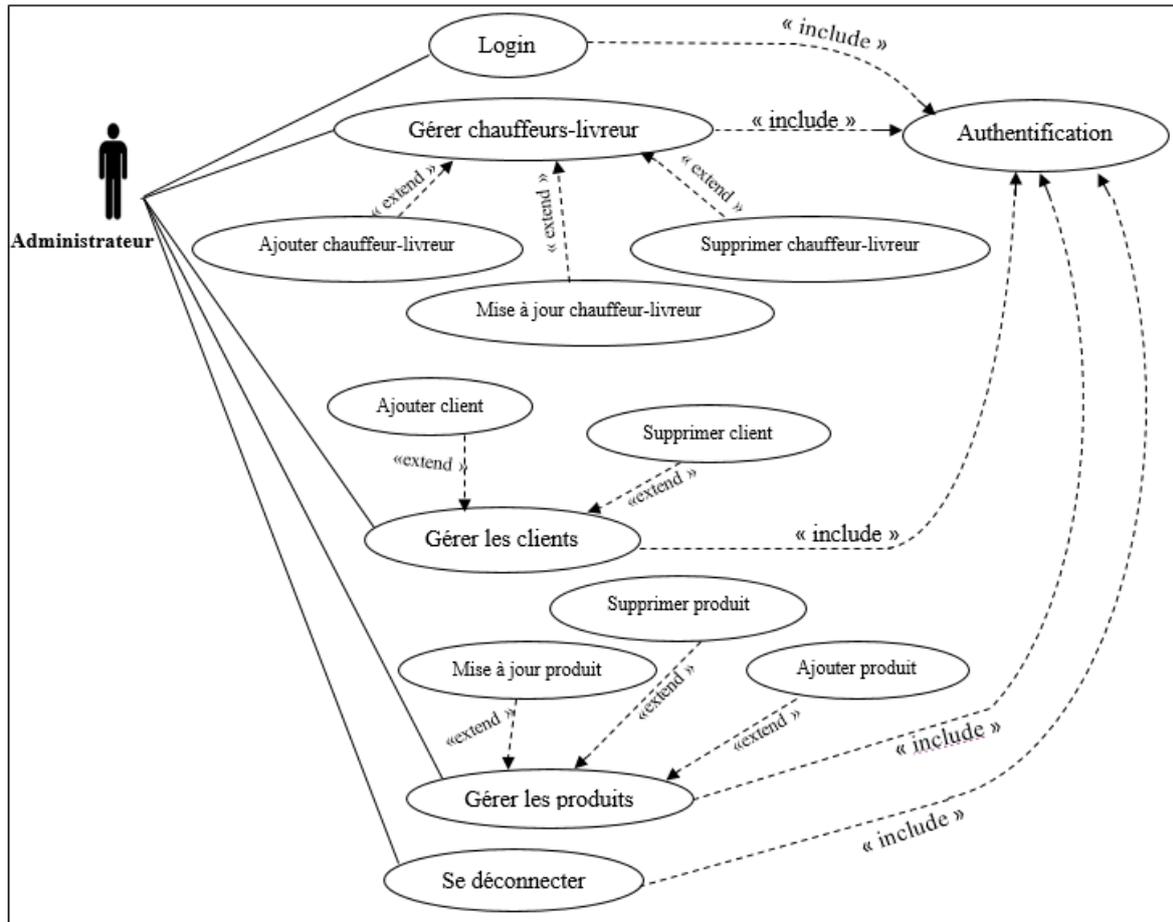


Figure 3.2: Diagramme de cas d'utilisateur du système « **Administrateur** »

Par la suite, on va présenter les diagrammes de cas d'utilisateur des sources de données du système. Notre source de données contenir deux bases de données : la base de données produits et la base de données chauffeur-livreur.

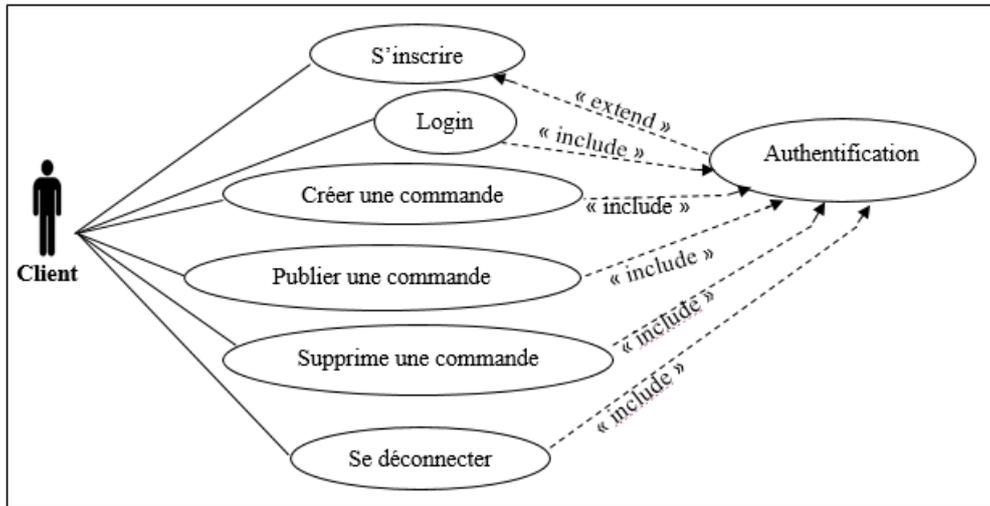


Figure 3.3: Diagramme de cas d'utilisateur « Produits »

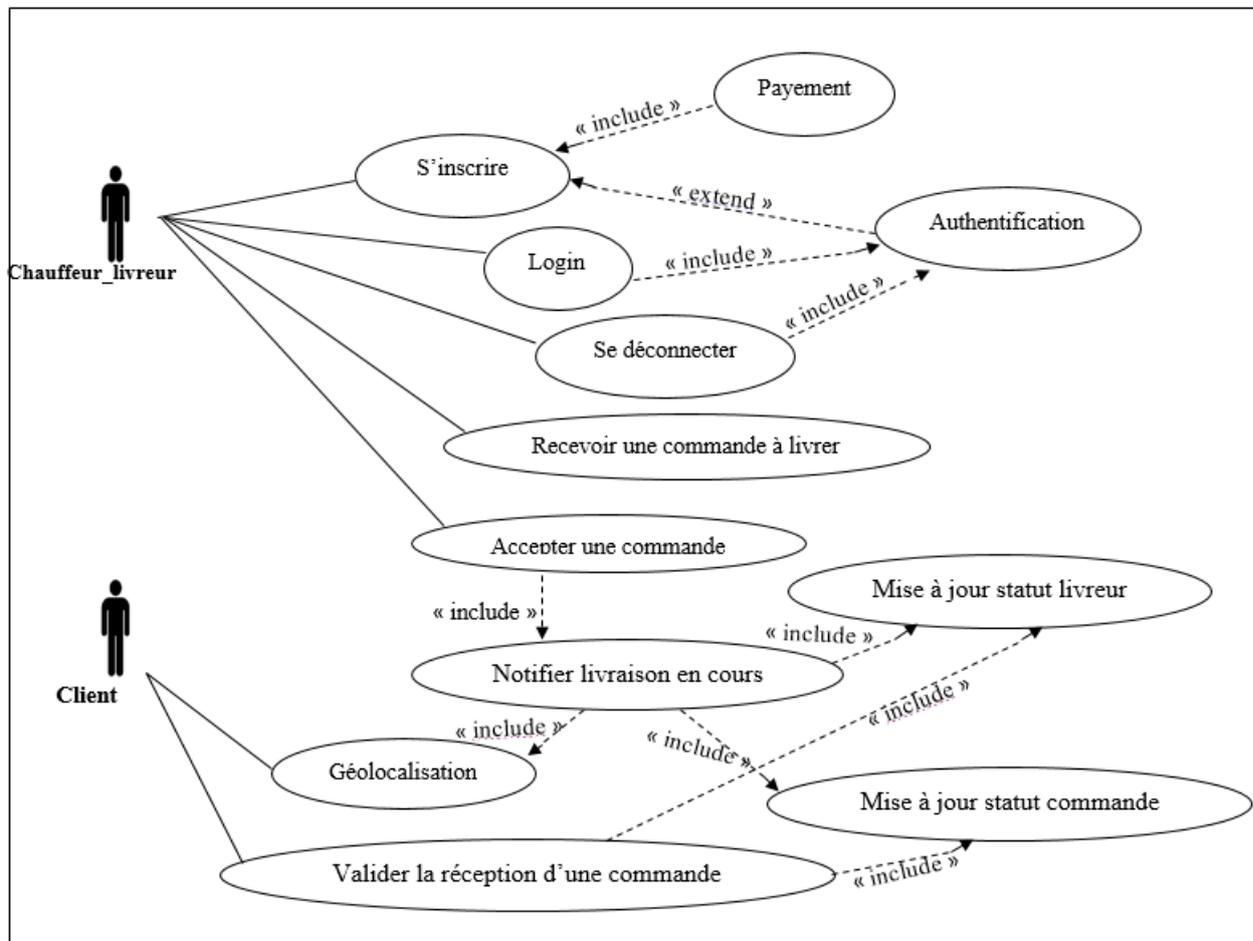
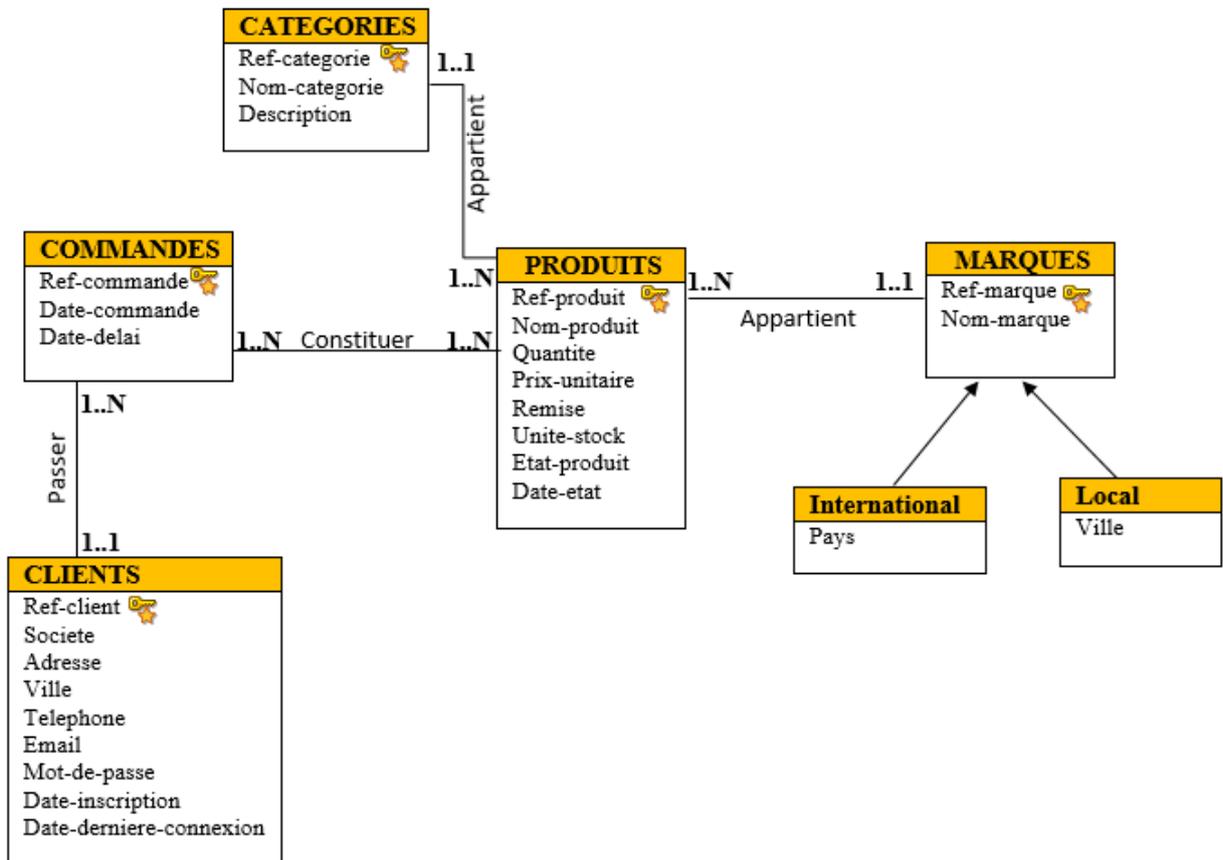


Figure 3.4: Diagramme de cas d'utilisateur « chauffeur-livreur »

3.5.3 Diagrammes de classe des bases de données source

Un diagramme de classes est un diagramme de structure (ou diagramme statique) nous permet de représenter les classes intervenant dans le système [28]. Les schémas suivants représentent les deux bases de données correspondent aux sources de données utilisés dans notre application :

Figure 3.5: schéma conceptuel de base de données « **Produits** »

Le modèle relationnel de base de données « **PRODUITS** »

PRODUITS(Ref_produit, Nom_produit, Quantité, Prix_unitaire, Remise, Unite_stock, Etat_produit, Date_etat, #Ref_categorie, #Ref_marque)

COMMANDES (Ref_commande, Date_commande, Date_delai, #Ref_client)

CLIENTS (Ref_client, Societe, Adresse, Ville, Telephone, Email, Mot_de_passe, Date_inscription, Date_deniere_connexion)

CATEGORIES (Ref_catégorie, Nom_categorie, Description)

MARQUES (Ref_marque, Nom_marque, Type_marque)

Constituer (#Ref_commande, #Ref_produit)

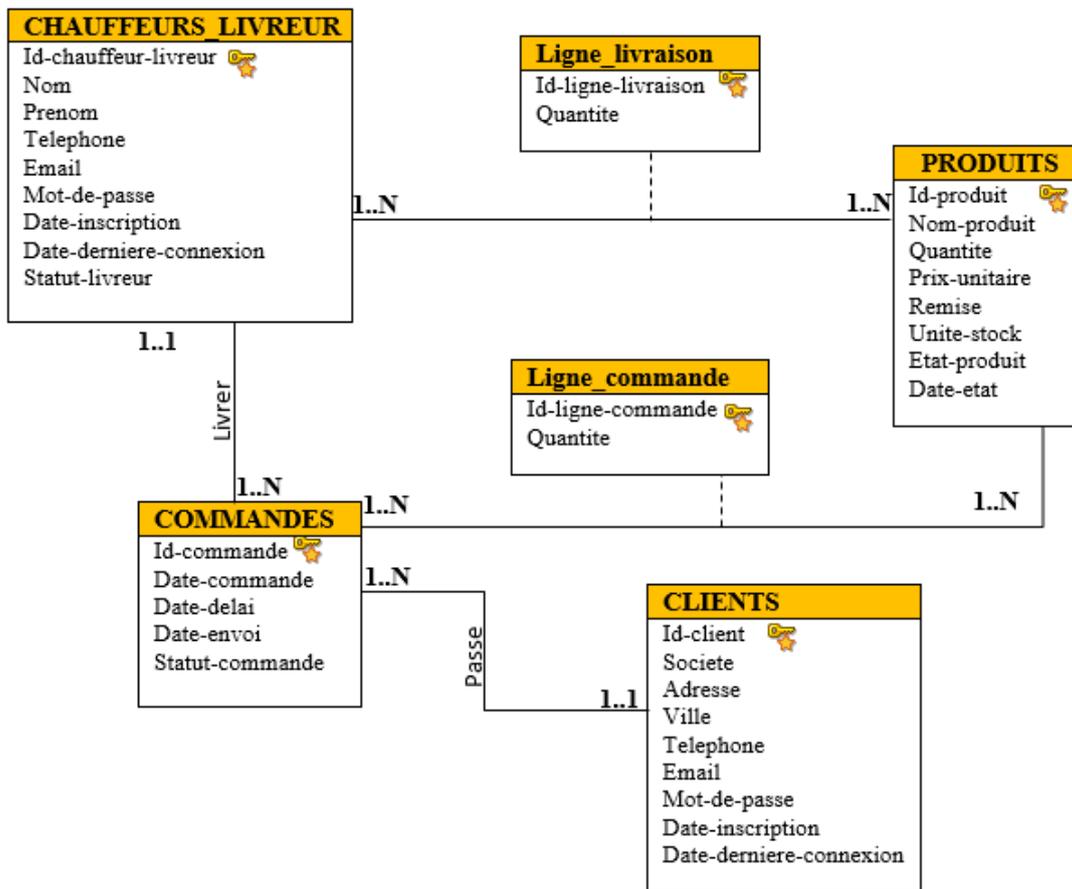


Figure 3.6: schéma conceptuel de base de données « CHAUFFEURS-LIVREUR »

Le modèle relationnel de base de données « CHAUFFEURS-LIVREUR »

CHAUFFEURS LIVREUR (Id_chauffeur_livreur, Nom, Prenom, Telephone, Email, Mot_de_passe, Date_inscription, Date_derniere_connexion, Statut_livreur)

COMMANDES (Id_commande, Date_commande, Date_delai, Date_envoi, Statut_commande, #Id_client, #Id_chauffeur_livreur)

CLIENTS (Id_client, Societe, Adresse, Ville, Telephone, Email, Mot_de_passe, Date_inscription, Date_derniere_connexion)

PRODUITS (Id_produit, Nom_produit, Quantite, Prix_unitaire, Remise, Unite_stock, Etat_produit, Date_etat, #Ref_categorie, #Ref_marque)

LIGNE_COMMANDE (Id_ligne_commande, Quantite, #Id_commande, #Id_produit)

LIGNE_LIVRAISON (Id_ligne_livraison, Quantite, #Id_chauffeur_livreur, #Id_produit)

3.6 Conception de l'entrepôt de données (QUALITE_DE_SERVICE)

Pour faciliter l'analyse, on présente la modélisation conceptuelle de l'entrepôt de données que nous proposons. Le modèle en Flocon de Neige permet de formaliser une hiérarchie au sein d'une dimension et d'agrégant les mesures selon des niveaux hiérarchiques, afin d'aider à obtenir des résultats rapides et clairs répondant à nos besoins.

Les concepts d'un entrepôt de données sont :

- ◇ Fait
- ◇ Mesure
- ◇ Dimension
- ◇ Hiérarchie
- ◇ Attribut de dimension

3.6.1 Les faits et les dimensions

Nous allons présenter la table de faits et les tables de dimensions qui sont utilisées pour créer le schéma "F_QUALITE_SERVICE". D'ailleurs, on va mentionner les données nécessaires pour définir chaque dimension et leurs hiérarchies s'ils existent.

3.6.1.1 La table des faits et les mesures

La table F_QUALITE_SERVICE contient l'enregistrement d'une combinaison d'attributs de différentes tables de dimension (les clés étrangères) et l'ensemble des mesures d'analyse.

Attribut	Type du champ	Description	Type d'attribut
Id_chauffeur_livreur	Number	L'identifiant d'un chauffeur livreur	Clés étrangères
Id_client	Number	L'identifiant d'un client	
Id_produit	Number	Produits à commander	
Id_jour	Number	Jour de service	
Id_commande	Number	La commande de client	
Taux_disponibilite_produit_cat	Number	Taux de disponibilité de produit par catégorie	Mesures
Taux_disponibilite_produit_mrq	Number	Taux de disponibilité de produit par marque	
Taux_commande_sans_livreur	Number	Taux de commandes en attends	
Taux_livreur_sans_commande	Number	Taux de chauffeurs livreurs en chômage	
Temps_moyen_de_reponse	Number	Le temps moyen de réponse	
Taux_retardement	Number	Taux de retardement de livraison	

Table 3.2: La table des faits (F_QUALITE_SERVICE).

3.6.1.2 Les dimensions et leurs hiérarchies

Concernant chaque dimension, nous allons présenter leurs hiérarchies avec tous les attributs de dimension.

Dimensions	Hiérarchies		
Temps	D_JOUR	→	D_MOIS → D_ANNEE
Articles	D_PRODUICTS	→	D_CATEGORIES
		→	D_MARQUES
Customers	D_CLIENTS		
Livreurs	D_CHAUFFEURS_LIVREUR		
Ordres	D_COMMANDES		

Table 3.3: Dimensions & hiérarchies.

◆ La dimension LIVREURS

Le contenu de cette dimension :

◇ D_CHAUFFEURS_LIVREUR

ATTRIBUT	TYPE	DESCRIPTION
ID_CHAUFFEUR_LIVREUR	Number	Clé primaire
NOM	Varchar2	Le nom de chauffeur livreur
PRENOM	Varchar2	Le prénom de chauffeur livreur
TELEPHONE	Varchar2	Nombre de téléphone du chauffeur livreur
EMAIL	Varchar2	E-mail du chauffeur livreur
MOT_DE_PASSE	Varchar2	Mot de passe pour accéder au compte de chauffeur livreur
DATE_INSCRIPTION	Date	La date d'inscription
DATE_DERNIERE_CONNEXION	Date	La date de dernière connexion du Chauffeur-livreur
STATUT_LIVREUR	Varchar2	Représente l'état du chauffeur-livreur

Table 3.4: D_CHAUFFEURS_LIVREUR.

◆ La dimension ORDRES

Le contenu de cette dimension :

◇ D_COMMANDES

ATTRIBUT	TYPE	DESCRIPTION
ID_COMMANDE	Number	Clé primaire
DATE_COMMANDE	Date	La date de la commande
DATE_DELAI	Date	La date limite de livraison de la commande
DATE_ENVOI	Date	La date d'arrivée de la commande
STATUT_COMMANDE	Varchar2	Représente l'état de la commande

Table 3.5: D_COMMANDES.

◆ La dimension ARTICLES

Les niveaux d'hérarchie qui définie cette dimension sont :

◇ D_PRODUIITS

ATTRIBUT	TYPE	DESCRIPTION
ID_PRODUIT	Number	Clé primaire
NOM_PRODUIT	Varchar2	Le nom de produit
QUANTITE	Number	La quantité de produit
PRIX_UNITAIRE	Number	Le prix unitaire de produit
REMISE	Number	Le remise de produit
UNITE_STOCK	Number	L'unité de produit qui est disponible dans le stock
ETAT_PRODUIT	Varchar2	La disponibilité de produit
CODE_MARQUE	Number	Clé étrangère
CODE_CATEGORIE	Number	Clé étrangère

Table 3.6: D_PRODUIITS.

◇ D_MARQUES

ATTRIBUT	TYPE	DESCRIPTION
CODE_MARQUE	Number	Clé primaire
NOM_MARQUE	Varchar2	Le nom de la marque
TYPE_MARQUE	Varchar2	Le type de la marque

Table 3.7: D_MARQUES.

◇ D_CATEGORIES

ATTRIBUT	TYPE	DESCRIPTION
CODE_CATEGORIE	Number	Clé primaire
NOM_CATEGORIE	Varchar2	Le nom de catégorie
DESCRIPTION	Varchar2	La description de catégorie

Table 3.8: D_CATEGORIES.

◆ La dimension TEMPS

Les niveaux d'hierarchie qui définit cette dimension sont :

◇ D_JOURS

ATTRIBUT	TYPE	DESCRIPTION
ID_JOUR	Number	Clé primaire
JOUR	Date	La date du jour
CODE_MOIS	Number	Clé étrangère

Table 3.9: D_JOURS.

◇ D_MOIS

ATTRIBUT	TYPE	DESCRIPTION
CODE_MOIS	Number	Clé primaire
MOIS	Varchar2	La désignation du mois
CODE_ANNEE	Number	Clé étrangère

Table 3.10: D_MOIS.

◇ D_ANNEE

ATTRIBUT	TYPE	DESCRIPTION
CODE_ANNEE	Number	Clé primaire
ANNEE	Number	L'année

Table 3.11: D_ANNEE.

◆ La dimension CUSTOMERS

Le contenu de cette dimension :

◇ D_CLIENTS

ATTRIBUT	TYPE	DESCRIPTION
ID_CLIENT	Number	Clé primaire
SOCIETE	Varchar2	Le nom de la société (client)
ADRESSE	Varchar2	L'adresse du client
VILLE	Varchar2	La ville du client
TELEPHONE	Varchar2	Le nombre de téléphone du client
EMAIL	Varchar2	E-mail du client
MOT_DE_PASSE	Varchar2	Mot de passe pour accéder au compte de client
DATE_INSCRIPTION	Date	La date d'inscription du client
DATE_DERNIERE_CONNEXION	Date	La date de dernière connexion du Client

Table 3.12: D_CLIENTS.

3.6.2 Schéma en flocon de neige

Le schéma suivant représente le modèle dimensionnel décrivant les données dans l'entrepôt de données (schéma en flocon de neige):

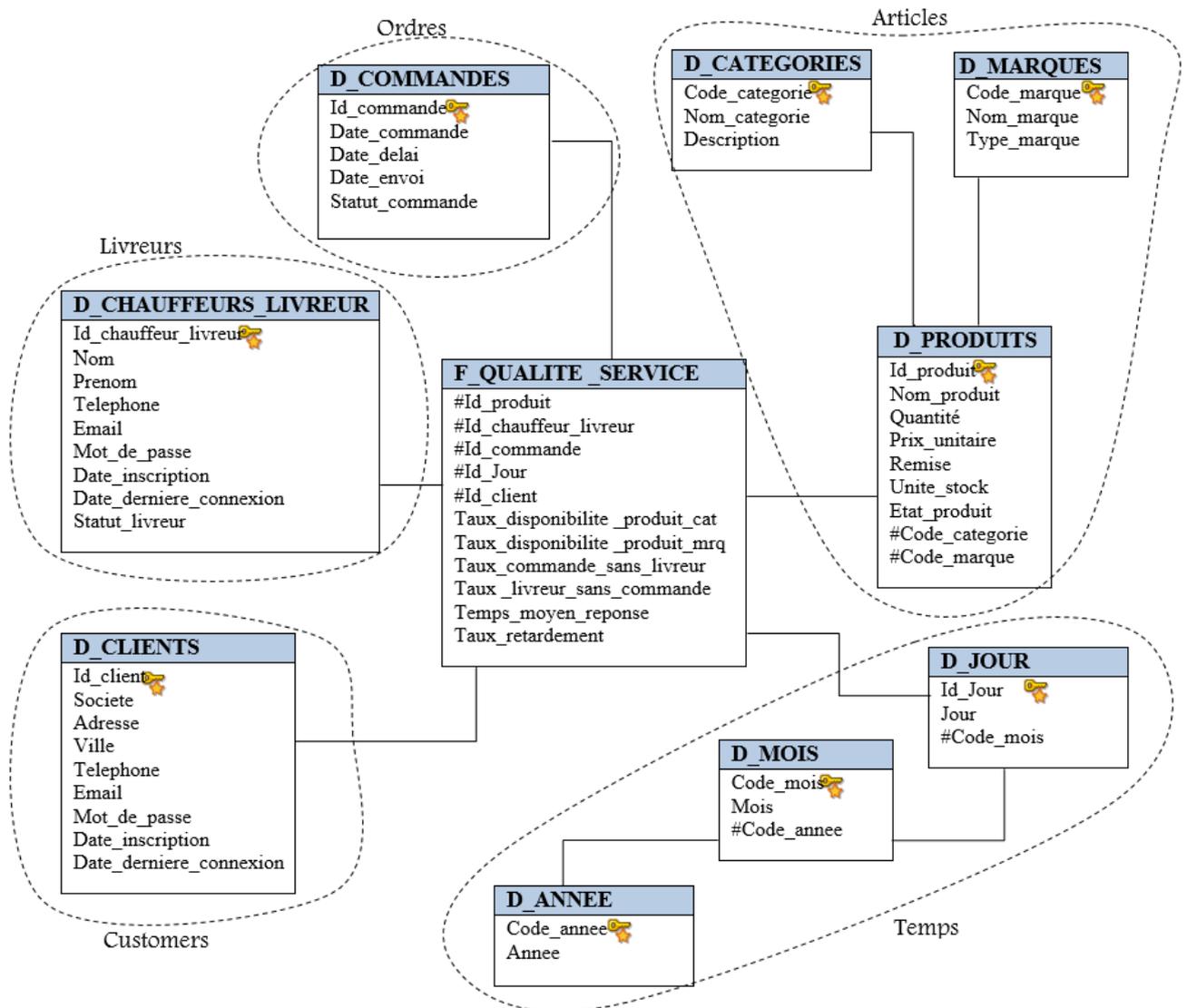


Figure 3.7: Modèle dimensionnel décrivant les données dans l'entrepôt de données.

3.7 La conception des règles d'analyse de notre travail

3.7.1 Syntaxe pour spécifier les règles d'analyse

Pour spécifier les règles d'analyse, la syntaxe générale suivante est inspirée des travaux de (Thalhammer et al.) dans [18] :

```

LA RÈGLE D'ANALYSE : <nom de la règle>
ÉVÈNEMENT : BEFORE/AFTER {<nom d'événement>, <type d'événement>}
CONDITION :
USING CUBE : <définition de cube> {<les dimensions attachées>}
IF (<condition principale>) <OPERATION OLAP> FROM <table de fait>
ACTION : <exécuter le trigger>
INSERT/DELETE/UPDATE {nom de la table}

```

Figure 3.8: La syntaxe pour spécifier une règle d'analyse.

3.7.2 Les règles d'analyse

◆ La règle d'analyse 1 :

```

LA RÈGLE D'ANALYSE : BloqueChauffeurLivreur
ÉVÈNEMENT : AFTER {FIN_MOIS, Périodique}
CONDITION :
USING CUBE : Qualité_de_service { Temps T, Ordres O, Livreurs L }
IF (taux_de_retardement_par_livreur >= 35%) ROLLUP L TO
D_CHAUFFEURS_LIVREUR, O TO D_COMMANDES, T TO D_MOIS
SLICE L.Id_chauffeur_livreur = '...' FROM F_QUALITE_SERVICE
ACTION :
UPDATE {CHAUFFEURS_LIVREUR}

```

Figure 3.9: La règle d'analyse **BloqueChauffeurLivreur**.

◆ La règle d'analyse 2 :

```

LA RÈGLE D'ANALYSE : NouveauChauffeurLivreur
ÉVÈNEMENT : BEFORE {INSERT ON CHAUFFEURS_LIVREUR, Fréquence}
CONDITION :
USING CUBE : Qualité_de_service { Livreurs L}
IF (nouveau_chauffeur_livreur ∉ L) ROLLUP L TO D_CHAUFFEURS_LIVREUR
FROM F_QUALITE_SERVICE
ACTION :
INSERT INTO {CHAUFFEURS_LIVREUR}

```

Figure 3.10: La règle d'analyse **NouveauChauffeurLivreur**.

◆ La règle d'analyse 3 :

```

LA RÈGLE D'ANALYSE : SupprimerChauffeurLivreurInactif
ÉVÈNEMENT : BEFORE {DELETE ON CHAUFFEURS_LIVREUR, Périodique}
CONDITION :
USING CUBE : Qualité_de_service { Livreurs L}
IF (Sysdate – date_dernière_connexion >= 30) ROLLUP L TO
D_CHAUFFEURS_LIVREUR
FROM F_QUALITE_SERVICE
ACTION :
BEFORE FROM {CHAUFFEURS_LIVREUR}

```

Figure 3.11: La règle d'analyse **SupprimerChauffeurLivreurInactif**.

◆ La règle d'analyse 4 :

```
LA RÈGLE D'ANALYSE : NouveauProduit  
ÉVÈNEMENT : BEFORE {INSERT ON PRODUIT, Périodique}  
CONDITION :  
USING CUBE : Qualité_de_service {Articles A}  
IF (nouveau_produit ∉ A)ROLLUP A TO D_PRODUIITS  
FROM F_QUALITE_SERVICE  
ACTION :  
INSERT INTO {PRODUITS}
```

Figure 3.12: La règle d'analyse **NouveauProduit**.

◆ La règle d'analyse 5 :

```
LA RÈGLE D'ANALYSE : ModifierProduit  
ÉVÈNEMENT : BEFORE {UPDATE ON PRODUIT, Périodique}  
CONDITION :  
USING CUBE : Qualité_de_service {Articles A}  
ROLLUP A TO D_PRODUIITS  
SLICE A.Id_PRODUIITS = '...' FROM F_QUALITE_SERVICE  
ACTION :  
UPDATE ON {PRODUITS}
```

Figure 3.13: La règle d'analyse **ModifierProduit**.

◆ La règle d'analyse 6 :

```

LA RÈGLE D'ANALYSE : SupprimerProduit
ÉVÈNEMENT : BEFORE {DELETE ON PRODUITS, Périodique}
CONDITION :
USING CUBE : Qualité_de_service {Articles A}
IF (etat_produit = 'Rupture' ) ROLLUP A TO D_PRODUITS
SLICE A.Id_PRODUITS = '...' FROM F_QUALITE_SERVICE
ACTION :
DELETE FROM {PRODUITS}

```

Figure 3.14: La règle d'analyse **SupprimerProduit**.

◆ La règle d'analyse 7 :

```

LA RÈGLE D'ANALYSE : NouveauClient
ÉVÈNEMENT : BEFORE {INSERT ON CLIENTS, Périodique}
CONDITION :
USING CUBE : Qualité_de_service {Costumers C}
IF (nouveau_client ∉ C) ROLLUP C TO D_CLIENTS
FROM F_QUALITE_SERVICE
ACTION :
INSERT INTO {CLIENTS}

```

Figure 3.15: La règle d'analyse **NouveauClient**.

◆ La règle d'analyse 8 :

```

LA RÈGLE D'ANALYSE : SupprimerClientInactif
ÉVÈNEMENT : BEFORE {DELETE ON CLIENTS, Périodique}
CONDITION :
USING CUBE : Qualité_de_service {Costumers C}
IF (Sysdate -date_derniere_connexion ≥ 30) ROLLUP C TO D_CLIENTS
SLICE C.Id_client = '...' FROM F_QUALITE_SERVICE
ACTION :
DELETE ON {CLIENTS}

```

Figure 3.16: La règle d'analyse **SupprimerClientInactif**.

◆ La règle d'analyse 9 :

```

LA RÈGLE D'ANALYSE : SupprimerMarque
ÉVÈNEMENT : BEFORE {DELETE ON MARQUES, Périodique}
CONDITION :
USING CUBE : Qualité_de_service {Articles A}
IF (Code_marque ∉ D_PRODUICTS) ROLLUP A TO D_MARQUES
SLICE A.Code_marque = '...' FROM F_QUALITE_SERVICE
ACTION :
DELETE ON {MARQUES}

```

Figure 3.17: La règle d'analyse **SupprimerMarque**.

◆ La règle d'analyse 10 :

```

LA RÈGLE D'ANALYSE : NouveauMarque
ÉVÈNEMENT : BEFORE {INSERT ON MARQUES, Périodique}
CONDITION :
USING CUBE : Qualité_de_service {Articles A}
IF (nouveau_marque ∉ A) ROLLUP A TO D_MARQUES
FROM F_QUALITE_SERVICE
ACTION :
INSERT INTO{MARQUES}

```

Figure 3.18: La règle d'analyse **NouveauMarque**.

◆ La règle d'analyse 11 :

```

LA RÈGLE D'ANALYSE : NotifierChauffeurLivreur
ÉVÈNEMENT : AFTER {FIN_MOIS, Périodique}
CONDITION :
USING CUBE : Qualité_de_service {Livreaux L, Ordres O}
IF (statut_livreur = 'Libre' && il existe statut_commande = 'en attend')
ROLLUP L TO D_CHAUFFEURS_LIVREUR, O TO D_COMMANDES
SLICE L.Id_chauffeur_livreur = '...' FROM F_QUALITE_SERVICE
ACTION :
(Envoyer la liste des commandes en-attend au Chauffeurs-livreur libre.)

```

Figure 3.19: La règle d'analyse **NotifierChauffeurLivreur**.

3.8 Conclusion

Ce chapitre contient notre présentation de domaine de gestion des livraisons à distance. En commençant par l'architecture globale de notre système, la description conceptuelle des bases de données sources, puis en définissant l'entrepôt de données passif. Par la suite, nous avons défini les règles d'analyse nécessaires pour administrer notre système. La phase qui est suivie la conception est l'implémentation ou la mise en œuvre, que nous aborderons dans le chapitre suivant.