# Modèle relationnel

# Appui technique

Normalisation dans le modèle relationnel

## **Les concepts**

* Attribut : Un attribut est un identificateur (un nom) décrivant une information stockée dans une base.

Par exemple le**nir** (numéro de sécurité sociale) et le **nom** d'une personne sont des attributs.

* Domaine : Le domaine d'un attribut est l'ensemble, fini ou infini, de ses valeurs possibles.

Par exemple, l'attribut **nir** a pour domaine l'ensemble des combinaisons de quinze chiffres (ou lettre pour la septième position) et **nom** a pour domaine l'ensemble des combinaisons de lettres.

* Relation : Une relation (table) est représentée sous la forme d'un tableau à deux dimensions, dans lequel les n attributs figurent les titres des n colonnes.

Par exemple :

Cette relation sera notée : **voiture (marque : chaîne, couleur : chaîne, plaque : chaîne)**

* Degré : Le degré d'une relation est son nombre d'attributs.
* Occurrence : Une occurrence est un élément de l'ensemble figuré par une relation. Autrement dit, une occurrence est une ligne du tableau (un tupple).
* Cardinalité : La cardinalité d'une relation est son nombre d'occurrences.

## **Normalisation**

**Problèmes soulevés par une mauvaise modélisation :**

On peut alors se rendre compte que des redondances sont présentes, et l'on sait qu'elles conduiront à des problèmes :

* De contrôle de la cohérence de l'information,
* De mise à jour (changement de nom à reporter dans de multiples tuples),
* De perte d'information lors de la suppression de données (disparition des informations concernant un type de véhicule)
* De difficulté à représenter certaines informations (un type de véhicule sans propriétaire).

Dans le contexte des Bases de Données Relationnelles, la normalisation est le processus visant à optimiser la modélisation du Modèle Relationnel faite par le concepteur.

Le processus de modélisation proposé passant par le modèle Entité-Relation (ou entité-Association) et ensuite par le Modèle Relationnel pourrait produire certaines anomalies indésirables.

En suivant rigoureusement le processus de normalisation, l'optimisation obtenue tendra vers une solution quasi optimale d'un point de vue théorique.

## **Objectif de la normalisation**

**L'objectif de cette manipulation consiste à modifier le modèle relationnel initialement créé par le concepteur afin :**

* D'éliminer toutes les redondances ;
* De découper adéquatement les structures dépendantes.

**Ainsi, la robustesse de la nouvelle solution permettra :**

* Une meilleure compréhension des interrelations entre chacune des données ;
* De minimiser la tâche de modification de la structure lorsqu'on désirera la mettre à jour avec de nouveaux éléments ;
* D‘éviter la redondance (intégrité ainsi que contre-performance liée) ;
* D'éviter les anomalies transactionnelles : insertion, suppression et de mise à jour ;
* De faciliter l'écriture des requêtes non prévues au moment de la conception.

Le processus de normalisation consiste principalement à diviser les relations initiales en plus petites relations, tout en définissant les liens qui les unissent.

Ainsi, les opérations d'insertion, de suppression et de consultation se feront d'abord dans la table concernée et ensuite, se propageront dans les tables liées s'il y a lieu.

Finalement, on dira qu'une **BDR** est normalisée si elle respecte les formes normales.

## **Formes normales**

Les formes normales 1 et 2 sont essentielles à un Modèle Relationnel formel.

Très souvent, le processus de normalisation respecte uniquement les trois premières formes normales.

**Les quatre dernières formes sont moins utilisées :**

* Elles sont plus abstraites et souvent difficiles à identifier ;
* Lorsqu'elles sont appliquées, elles apportent un découpage sévère du Modèle Relationnel pour les parties concernées ;
* Souvent, les règles identifiées sont abrogées lors du processus de dé-normalisation.

## **Normalisation – 1NF**

**Une relation respecte la 1ère forme normale**

Lorsque :

* Tous les attributs sont atomiques (valeur scalaire, indivisible).

Impacts :

* Les attributs composites sont interdits ;
* Les attributs multivalués sont interdits ;
* Impose implicitement qu'un attribut membre de la clé primaire soit à valeur unique ;
* Permet d'augmenter grandement la performance du SGBD en évitant de parcourir des attributs sous forme de liste et en créant des index performants.

**Exemple de relation qui n'est pas en 1NF :**

**1ère Solution :**

* Inconvénients : Stockage de valeurs nulles, impossible de stocker un nouveau numéro de téléphone.
* Avantages : Tout est dans la même relation.
* **2ème solution :**
* Inconvénients : Deux relations au lieu d'une seule, opérations de jointure pour accéder aux informations.
* Avantages : Pas de valeur nulle, possibilité d'extension.

## **Normalisation – 2NF**

**Une relation est en 2NF :**

Lorsque :

* Elle est en 1NF.
* Et si tout attribut qui n'est pas dans une clé ne dépend pas d'une partie seulement d'une clé. C'est à dire encore que toutes les DF issues d'une clé sont élémentaires.

**Exemple de relation qui n'est pas en 2NF :**

Commande(numFou, refArticle, nomFou, adresseFou, quantité, prix)

**Solution :**

* Fournisseur(numFou, raisonSociale, adresse)
* Commande(numFou, refArticle, quantité, prix)

## **Normalisation – 3NF**

**Une relation est en 3FN :**

Lorsque :

* Elle est en 2FN,
* chaque attribut de la relation ne dépend que de la clé et pas d'un autre attribut de la relation.

**Exemple de relation qui n'est pas en 3NF :**

VéhiculeLoué(numVéhicule, numClient, codeType, catégorie, kilométrage)

**Solution :**

VéhiculeLoué(numVéhicule, numClient, codeType, kilométrage)

TypeCatégorie(codeType, catégorie)

## **Normalisation - BCNF**

**Une relation est en FNBC (Forme Normale de Boyce-Codd) :**

Lorsque :

* Elle est en 3NF,
* Les seules DF existant dans les relations sont celles de la clé vers les attributs non clés.

**Exemple de relation qui n'est pas en BCNF :**

Agence(agence, ville, département, CA)

**Solution :**

Agence(agence, ville, CA)

Ville(Ville, département)

# Étude de cas : Le modèle Entité /

# Étude de cas : Le modèle Entité / AssociationAppui technique

Le modèle Entité / Association :

Cas pratique sur une entreprise de distribution, qui fait de la vente sur stock.

Le système d'information est connu à travers ce qu'il est convenu d'appeler **l'univers du discours** appelé aussi **base de connaissance**.

En fait, il s'agit des informations recueillies sur le système dans le cahier des charges ( par exemple ) suite à :

* l'interview des acteurs du système, repris et rédigé précisément.
* Une collection de documents actuels ou futurs (entrant et sortant des postes de travail).

## **Expression des besoins du client**

On dispose du cahier des charges d'une entreprise de distribution qui fait de la vente sur stock, à partir des commandes effectuées par les clients.

On va considérer le domaine d'étude "suivi des commandes clients" qui est un "sous univers du discours" (ou partie de la base de connaissance) de cette entreprise.

## **Le cahier des charges**

**Le rédactionnel de ce domaine nous fait savoir que :**

« L'entreprise reçoit les commandes par téléphone ou par courrier.

Dans tous les cas, ces commandes doivent préciser quel est le client payeur ( en général le siège de la société) et le client, destinataire de la livraison (lorsque la société dispose de plusieurs dépôts)

Avec, pour chacun : son numéro, son nom, son adresse.

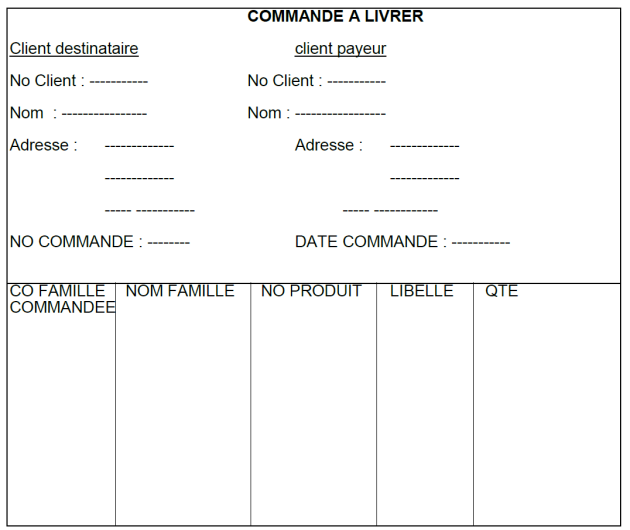
Pour chaque produit commandé, on doit connaître le code et le nom de la famille à laquelle il appartient, ainsi que son numéro, son libellé et la quantité demandée.

Le réceptionnaire de la commande notifie la date d'arrivée de la commande et lui affecte un numéro d'ordre (incrémenté de 1 à chaque fois) ».

## **La maquette des commandes**

Par ailleurs, on dispose d'une maquette des commandes à livrer, dont le format est donné dans la page suivante.

Cette maquette fait partie de la collection des documents recueillis (dans l'existant) ou élaborés (dans l'expression des besoins), lors du cahier des charges.



## **La méthode**

De donner une **représentation des données** de cette base de connaissance (partielle dans le cas de notre exemple) qui soit indépendante des traitements mis en jeu pour l'obtention de ces commandes.

Pour ce faire :

## **Étape 1 : Dictionnaire des données**

On établira**un dictionnaire des données** utiles à la gestion de ces commandes.

Si on ne dispose que du rédactionnel, on soulignera chaque mot du vocabulaire utilisé dans la gestion des commandes et on en dressera la liste.

Si on ne dispose que d'une maquette ou d'un document, des rubriques qui donnent le même résultat.

Si on dispose des deux, c'est encore mieux. Du fait que le rédactionnel complète la maquette en terme de meilleure compréhension du système et permet de découvrir des incohérences, s'il y en a.

D'autre part, on peut obtenir dans ce cas un dictionnaire plus riche et donc plus proche de la réalité. Par exemple, on pourrait imaginer que la maquette fait figurer la date de livraison souhaitée, alors que le rédactionnel n'en parle pas. Ce qui permet de contrôler des omissions.

On remarque d'ailleurs, que la notion de client destinataire et payeur a été explicitée dans le rédactionnel.

**Ici la constitution du dictionnaire nous a amené à la liste suivante :**

1. numéro client destinataire,
2. nom client destinataire,
3. adresse client destinataire,
4. numéro client payeur,
5. nom client payeur,
6. adresse client payeur,
7. numéro de la commande,
8. date de la commande,
9. code famille,
10. nom famille,
11. numéro produit,
12. libellé produit,
13. quantité commandée.

## **Étape 2 : Regroupement par entité**

On établit un regroupement des mots qui ont traits, de façon invariante, à une même **entité**.

Ces entités seront appelées des **objets**.

**On constate ici que les données :**

- **1, 2, 3** sont associées à un objet **CLIENT DESTINATAIRE,**

- **4, 5, 6** à un objet **CLIENT PAYEUR**,

- **7, 8** à un objet **COMMANDE**,

- **9, 10** à un objet **FAMILLE**,

- **11, 12** à un objet **PRODUIT**.

Ces données sont des **propriétés** des objets auxquels elles se rapportent.

Mais qu'en est-il de la donnée 13 : quantité commandée ? elle n'est pas liée invariablement à l'objet produit. Nous traiterons son cas à l'étape 4.

## **Étape 3 : Les relations entre objets**

On va décrire maintenant les rapports qui existent entre certains objets. On les appellera des **relations entre objets**. On les déduit de la **sémantique** suivante :

**A** client destinataire **dépend** de **client payeur**,

**B** client destinataire **passe** des **commandes**,

**C** produit ***appartient*** à **famille**.

la relation :

**A** sera nommée **DEPENDRE**

**B**sera nommée **PASSER**

**C** sera nommée **APPARTENIR**

## **Étape 4 : Les contraintes liéEs aux objets**

Revenons à l'exemple traité dans les étapes 1, 2 et 3 et aux problèmes que posent certaines données qui ne peuvent être affectées à des objets.

C'est ainsi que la quantité commandée (rubrique 13) dépend à la fois de la commande et du produit.

En effet, s'il n'y a pas de commande, pour toute quantité commandée, on ne saura pas à quelle commande elle se réfère.

Et s'il n'y a pas de produit, on ne saura pas à quoi s'adresse cette quantité commandée.

Autrement dit, cette donnée n'a de sens que par l'existence de ce rapport, qu'on a nommé : relation entre commande et produit.

On nommera cette relation " commander " en référence à la sémantique suivante :

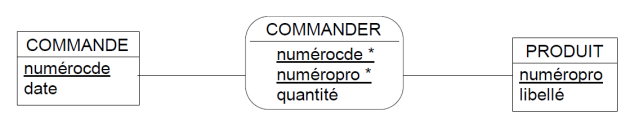
une **commande** sert à **commander des produits**.

La donnée (quantité commandée) est une **propriété** de la relation "commander».

On dit que c'est une **relation porteuse de données**, par opposition aux relations précédentes, qui étaient**non porteuses de données**, dites aussi **relations vides.**

Par ailleurs, on formalisera cette sémantique selon le schéma suivant, où l'on a inclus les propriétés des objets aussi bien que des relations, ainsi que 2 propriétés

supplémentaires (numéro commande et numéro produit) pour la relation COMMANDER :



On remarque que chaque objet ou relation, dispose d'un identifiant (ici numérocde identifiant de COMMANDE, numéropro identifiant de PRODUIT, numérocde+numéropro identifiant de COMMANDER).

Cela veut dire que l'on est capable de différencier, par cet identifiant, toutes réalisations de l'objet ou de la relation (ce que l'on appellera par la suite occurrence).

**Par exemple :**

Si une occurrence de commande était celle qui porte le numéro 158 (à la date du 12/04/89) et si une occurrence de produit était celle qui a le numéro 355 (de libellé "écrou de 14") alors une **occurrence** de commander serait celle qui a le code "158"+"355" (où la quantité commandée est de 658 unités) si, évidemment, le produit 355 a été commandé par la commande 158.

En général, l'identifiant d'une relation est la **concaténation** des identifiants des objets qu'elle relie (ici "158"+"355").

\* Le modèle conceptuel, normalement, ne fait pas apparaître les identifiants concaténés de la relation COMMANDER. Ils figurent ici à titre purement explicatifs.

## **Étape 5 : Les cardinalités**

On se pose la **question** de savoir si on peut représenter des**règles de gestion** qui expriment le fait que ces **relations** peuvent :

**Exister** (une ou plusieurs fois) ou **ne pas exister** (0 fois).

C'est ainsi que l'on mettra en place les **cardinalités**.

**Par exemple :**

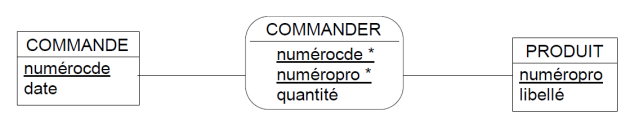
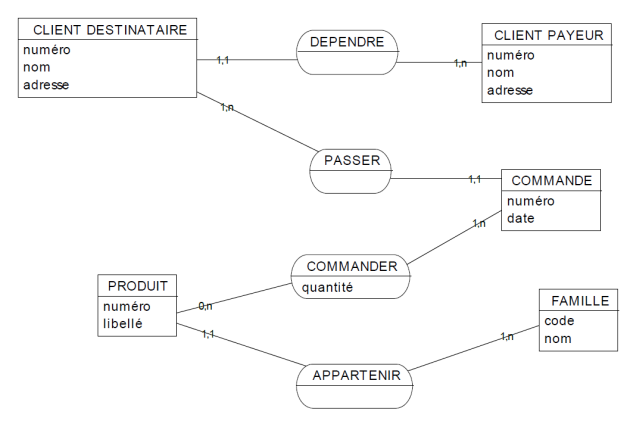
Un client payeur voit 1 client destinataire au minimum (il a au moins un dépôt chez lui ou à l'extérieur) et n° clients destinataires au maximum (s'il a n° dépôt ), ***ce sera une cardinalité 1, n(minimum 1, maximum n).***

Un client destinataire ne dépend que d'un client payeur et d'un seul**. Ce** **sera une cardinalité 1, 1 (minimum 1, maximum 1).**

Un produit peut être commandé par 0 commande au minimum (cas où il existe en stock mais non commandé) et n commandes au plus (cas où il existe en stock et on le voit apparaître dans plusieurs bons de commande), ***d'où la cardinalité 0, n*.**

De même, une commande sert à commander au moins un produit (1), sinon plusieurs(n), **ce sera une cardinalité (1, n)** etc...

Ces cardinalités seront notifiées sur le modèle comme suit :



## **Étape 6 : Création du MCD**

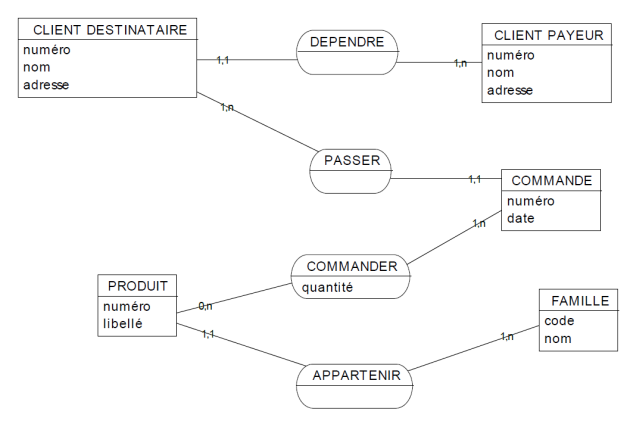
Enfin, on établira le modèle conceptuel des données définitif, où l'on reporte, pour chaque objet et chaque relation, ses propriétés. C'est à dire les données qui lui sont propres et les cardinalités de chaque objet qui voit une relation.

(Les identifiants des relations seront analysés et reportés dans le chapitre suivant) :

**MODÈLE CONCEPTUEL DE DONNÉES**

**MCD**

**Domaine "suivi des commandes clients" :**



## **Exercice de synthèse**

Dans une entreprise, on s'est limité au domaine "suivi des commandes de réapprovisionnement aux fournisseurs".

On a obtenu la liste de données suivantes, classées alphabétiquement :

1 adresse du fournisseur,

2 conditions particulières de la commande,

3 correspondant de la commande,

4 date de la commande,

5 date de livraison prévue de la commande,

6 libellé du produit,

7 montant global de la commande,

8 nom du fournisseur,

9 numéro de la commande,

10 numéro de téléphone du fournisseur,

11 prix unitaire du produit fixé par le fournisseur,

12 quantité produit commandée,

13 quantité produit en stock,

14 prix unitaire de vente (HT) du produit.

On sait que chaque produit est proposé par différents fournisseurs à un prix qu'ils ont établi et communiqué.

Déterminer les objets, les relations, les propriétés attachées à ces entités, et établir le MCD avec ses cardinalités en suivant les étapes 1-2-3-4-5.

## **Proposition de solution**

Voici une proposition de solution, en appliquant la même démarche,, énoncée précédemment.

## **Étape 1 : Dictionnaire de données**

1. adresse du fournisseur,
2. conditions particulières de la commande,
3. correspondant de la commande,
4. date de la commande,
5. date de livraison prévue de la commande,
6. libellé du produit,
7. montant global de la commande,
8. nom du fournisseur,
9. numéro de la commande,
10. numéro de téléphone du fournisseur,
11. prix unitaire du produit fixé par le fournisseur,
12. quantité produit commandée,
13. quantité produit en stock,
14. prix unitaire de vente (HT) du produit.

## **Étape 2 : Détermination des objets**

- **1,8,10** associé à **FOURNISSEUR**

- **2,3,4,5,7,9** associé à **COMMANDE**

- **6,13,14** associé à **PRODUIT**

## **Étape 3 : Détermination des relations**

**R1** Un **FOURNISSEUR** propose des **PRODUIT**

**R2** Une **COMMANDE** concerne des **PRODUIT**

**R3** Une **COMMANDE** est envoyée à un **FOURNISSEUR**

## **Étape 4 : Affectation de propriété**

**11** appartient à **R1** et est identifié par numéro fournisseur, libellé produit

**12** appartient à **R2** et est identifié par numéro de commande, libellé produit

## **Étape 5 : Affectation des cardinalités**

Une **COMMANDE** est composée de 1 à plusieurs **PRODUIT**.

Un **PRODUIT** n'est pas systématiquement commandé, mais n'est présent qu'une fois sur une **COMMANDE**.

Un **PRODUIT** est proposé par au moins un **FOURNISSEUR**.

Un **FOURNISSEUR** ne propose pas l'ensemble des **PRODUIT**.

Une **COMMANDE** concerne un seul **FOURNISSEUR**.

Chaque **FOURNISSEUR** peut avoir plusieurs **COMMANDE**.

## **Étape 6 : Modèle Conceptuel**

\* Le modèle conceptuel, normalement, ne fait pas apparaître les identifiants concaténés. Ils figurent ici à titre purement explicatifs.

