

La base de données :

Modèle relationnel

Table des matières

<u>1. Les données du modèle relationnel</u>	3
a) <u>L'importance de l'organisation</u>	3
<u>Donnée et base de données</u>	3
b) <u>L'organisation des données</u>	3
c) <u>L'organisation en modèle relationnel</u>	3
d) <u>Les données retenues</u>	4
<u>Les différents types d'informations</u>	4
<u>Les informations à modéliser</u>	4
e) <u>Le format des données</u>	5
<u>2. Le vocabulaire du modèle relationnel</u>	6
<u>3. Le formalisme du modèle relationnel</u>	8
<u>4. Les dépendances fonctionnelles du modèle relationnel</u>	9
a) <u>La dépendance fonctionnelle directe</u>	9
b) <u>La dépendance fonctionnelle entre deux relations</u>	10
c) <u>La dépendance fonctionnelle composée</u>	11
<u>5. Les règles de normalisation</u>	12
a) <u>L'intérêt des règles de normalisation</u>	12
b) <u>Les trois formes normales</u>	13
<u>La première forme normale</u>	13
<u>La deuxième forme normale</u>	13
<u>La troisième forme normale</u>	14

1. Les données du modèle relationnel

a) L'importance de l'organisation

Donnée et base de données

Une donnée (data en anglais) est une information (texte, image...) transmise à l'ordinateur pour être traitée dans une application.

Exemple : « Martin », nom d'un client, est une donnée.

Une base de données est un ensemble de données.

Exemple : la liste regroupant l'ensemble des coordonnées des clients est une base de données.

b) L'organisation des données

Les **données** (ou informations) ont un **rôle majeur** dans l'entreprise. Elles sont à la base des systèmes comptable, statistique, stratégique, décisionnel... La gestion de la PME passe par la gestion des données.

Ainsi, la **pérennité** dépend de l'organisation structurée de la base de données de la PME-PMI. Seule cette **organisation** permettra aux utilisateurs du système informatique d'accéder facilement et de manière **fiable** aux informations.

c) L'organisation en modèle relationnel

Le modèle relationnel est une méthode d'organisation des informations de la base de données selon des critères inventés par Edgar Frank Codd, informaticien britannique, dans les années 1970.

Les données sont organisées de façon logique et rationnelle afin de les rendre exploitables par un logiciel de type SGBRD (système de gestion de base de données relationnel). Cette organisation a pour objectif de faciliter le traitement des problèmes de cohérence et de redondance des données.

Dans le modèle relationnel, les données sont regroupées en ensembles homogènes appelés « relations ». Ces relations sont systématiquement reliées entre elles.

d) Les données retenues

Les différents types d'informations

En observant une facture, par exemple, on distingue :

- des informations saisies (depuis le clavier de l'ordinateur ou issues de la base de données où elles ont été préalablement saisies) ;
- des informations calculées (résultats d'un calcul) ;
- et des informations paramètres (données fixes).

SPORAGOGO
SARL SPORAGOGO
25 rue des Lilas
24210 Pennes-Les-Fontaines
Tél. : 04 90 66 25 23
RC Avignon A 562137 448
SIRET 562 137 448 0017 - APE 4207

Facture n° V547
Date 17/11/N
Client Martin Pierre
Adresse Rue Auguste Porte
Ville 84800 Cavaillon

Référence article	Désignation	Quantités	PU HT	Total HT
AIRMAF	Maillot collection Air Femme	5	10	50,00
AIRSHE	Maillot collection Air Enfant	8	8	64,00
FORMAF	Maillot collection Forme Femme	3	20	60,00
	Total HT			174,00
	Remise 5 %			8,70
	Net commercial			165,30
	Escompte 3 %			4,96
	Net financier			160,34
	TVA 19,6 %			31,48
	Total TTC en euros			191,82

Règlement comptant
 Oui Taper oui ou non

Les informations à modéliser

Toutes les informations n'ont pas leur place dans le modèle relationnel. Seules les données saisies sont à intégrer dans la base de données. Les informations calculées peuvent être retrouvées à tout moment en répétant les calculs.

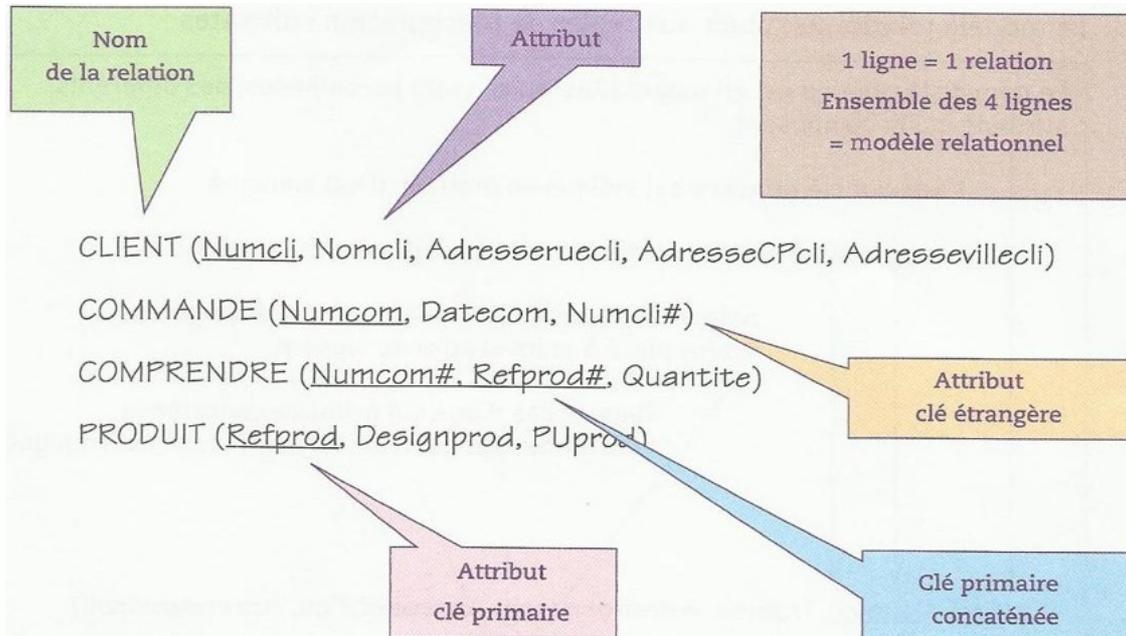
Les informations paramètres peuvent être mentionnées directement dans les documents de l'entreprise (les coordonnées de l'entreprise ou le taux de TVA, par exemple).

e) Le format des données

Les données seront caractérisées par un format afin d'être stockées et traitées correctement.

Format	Domaine de valeur	Possibilité de calcul	Exemples
Numérique	Données quantitatives	Oui	<ul style="list-style-type: none">• Prix unitaire• Quantité vendue
Texte	Données non quantitatives	Non	<ul style="list-style-type: none">• Nom du salarié• Référence produit• Code postal• Numéro de téléphone
Booléen	Données de type oui/non, vrai/faux, masculin/féminin...	Non	<ul style="list-style-type: none">• 0/N• V/F• M/F
Date	Données de type date	Oui	<ul style="list-style-type: none">• 20/11/12• 20 nov. 2012
Monétaire	Données quantitatives (le symbole monétaire est inscrit)	Oui	<ul style="list-style-type: none">• 2 000 €• 500 \$

2. Le vocabulaire du modèle relationnel



- Relation : une relation décrit des objets ou des faits.

Exemple : la relation COMMANDE décrit les bons de commande grâce à leur numéro, la date de passation de la commande et le numéro du client concerné par cette commande.

- Attribut : un attribut est un identificateur décrivant un type de données dans une base de données.

Exemple : Nomcli (nom du client) est un attribut de la relation CLIENT.

Chaque client sera décrit, entre autres, par son nom.

- Clé primaire : la clé primaire est un des attributs d'une relation. C'est l'attribut qui permet de retrouver systématiquement les autres de manière sûre et unique.

Exemple : si on recherche un bon de commande, connaître uniquement la date de commande ne suffit pas. En effet, l'entreprise peut passer plusieurs commandes le même jour. Connaître uniquement le numéro du client concerné ne suffit pas non plus : un client peut être amené à passer plusieurs commandes... En revanche, connaître le numéro de la commande suffit pour la retrouver de manière sûre car toute commande est systématiquement numérotée et il n'est pas donné deux fois le même numéro à deux commandes différentes. Ainsi, dans une clé primaire, il n'y a ni vide ni doublon.

- Clé étrangère : La clé étrangère est un attribut d'une relation qui fait référence à la clé primaire d'une autre relation.

Exemple : dans la relation COMMANDE, Numcli (numéro du client ayant passé la commande) correspond à un numéro de client dans la relation CLIENT. Toute valeur de l'attribut clé étrangère est obligatoirement présente dans les valeurs de l'attribut clé primaire.

Exemple : un bon de commande ne peut pas faire référence à un numéro de client ne figurant pas dans la relation CLIENT. La clé étrangère a un rôle déterminant : elle établit les liaisons entre les relations.

- Clé primaire concaténée : Il s'agit d'une clé primaire composée de deux attributs ou plus.

Exemple : pour trouver le nombre de produits « Alpha » commandés, il faut connaître à la fois le numéro de la commande et le code du produit concerné (car une même commande peut comprendre plusieurs produits différents).

3. Le formalisme du modèle relationnel

Le modèle relationnel obéit aux règles de présentation suivantes :

Le nom de la relation est en majuscules, suivi, entre parenthèses, des différents attributs qui la composent.

L'attribut clé primaire est indiqué en premier. Il est souligné.

Les différents attributs sont séparés par des virgules.

Si la relation comprend un attribut clé étrangère, il sera placé à la fin et suivi du signe #.

Dans le cas d'une clé primaire concaténée, l'ensemble des attributs formant la clé est souligné.

CLIENT (Numcli, Nomcli, Adresserveuecli, AdresseCPcli, Adressevillecli)

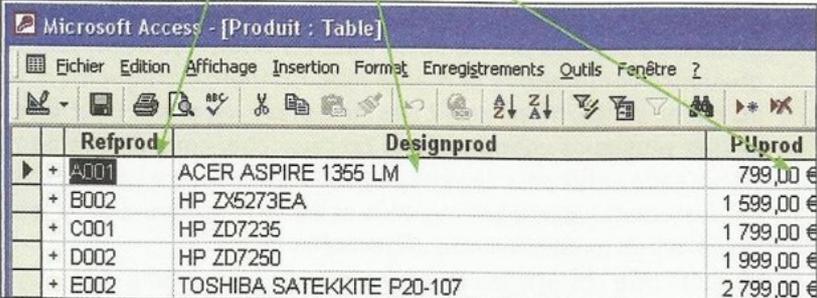
COMMANDE (Numcom, Datecom, Numcli#)

COMPRENDRE (Numcom#, Refprod#, Quantite)

PRODUIT (Refprod, Designprod, PUpod)

Dans sa traduction dans un logiciel de type SGBDR, chaque relation du modèle relationnel deviendra une table :

PRODUIT (Refprod, Designprod, PUpod)



	<u>Refprod</u>	Designprod	PUpod
▶ +	A001	ACER ASPIRE 1355 LM	799,00 €
+	B002	HP Z5273EA	1 599,00 €
+	C001	HP ZD7235	1 799,00 €
+	D002	HP ZD7250	1 999,00 €
+	E002	TOSHIBA SATEKKITE P20-107	2 799,00 €

4. Les dépendances fonctionnelles du modèle relationnel

L'étude des dépendances fonctionnelles conduit à identifier les liens entre les données et permet de normaliser le modèle relationnel.

a) La dépendance fonctionnelle directe

Lorsque la connaissance de la valeur de l'attribut d'une relation permet de connaître de manière sûre et unique la valeur d'un autre attribut de la même relation, il y a dépendance fonctionnelle directe entre les attributs de cette relation.

Exemple	Conséquence
La connaissance d'un numéro de client (Numcli) permet de trouver un nom et une adresse de client. Il y a dépendance fonctionnelle directe entre Numcli et Nomcli, entre Numcli et Adresseruecli...	L'attribut Numcli devient clé primaire de la relation CLIENT.
<u>Remarque :</u> La connaissance du nom du client (Nomcli) permettrait aussi de trouver son adresse mais à la seule condition qu'il n'y ait jamais un même nom pour deux clients différents. Auquel cas, la connaissance de ce nom renvoie vers un ou plusieurs clients. Il n'y a alors pas de dépendance fonctionnelle. C'est pourquoi il faut créer un numéro unique pour chaque client.	

b) La dépendance fonctionnelle entre deux relations

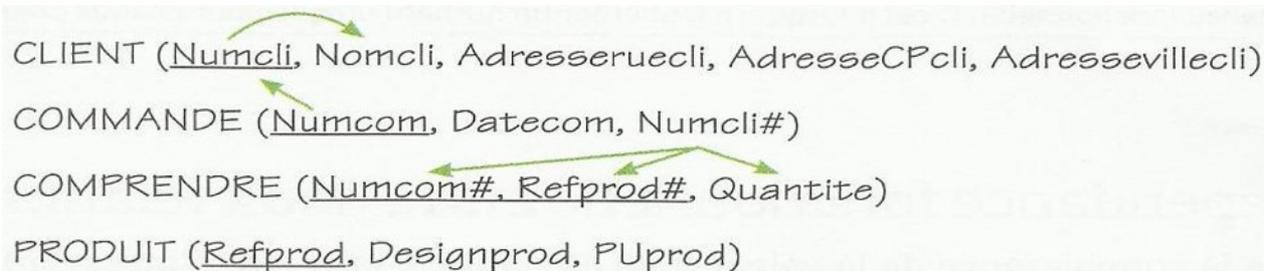
Lorsque la connaissance de la valeur d'un attribut clé primaire d'une relation permet de renvoyer vers une valeur unique de l'attribut clé primaire d'une autre relation, il y a dépendance fonctionnelle entre ces deux relations.

Exemple	Conséquence
La connaissance d'un numéro de commande (Numcom) permet de retrouver le numéro de client (Numcli) qui correspond.	L'attribut Numcli devient clé étrangère dans la relation COMMANDE. Les relations COMMANDE et CLIENT sont alors liées.
<p><u>Remarques :</u></p> <p>Numcli est clé étrangère dans la relation COMMANDE. On peut remarquer qu'il y a dépendance fonctionnelle directe entre la clé primaire de la relation COMMANDE (Numcom) et Numcli. En effet, à un numéro de commande correspond un et un seul client.</p> <p>En revanche, le numéro de commande (Numcom) n'aurait pas pu être clé étrangère dans la relation CLIENT. En effet, à un numéro de client (Numcli) peut correspondre plusieurs numéros de commande (Numcom). Il n'y a donc pas dépendance fonctionnelle directe entre le numéro de client et le numéro de commande. Ainsi, le numéro de commande ne peut pas être un attribut de la relation CLIENT.</p>	

c) **La dépendance fonctionnelle composée**

Lorsque la connaissance de la valeur d'un attribut clé primaire d'une relation et la connaissance d'une valeur d'un attribut clé primaire d'une autre relation permettent de connaître de façon unique la valeur d'un troisième attribut, il y a dépendance fonctionnelle composée (ou multiple) entre les deux premières relations.

Exemple	Conséquence
La connaissance d'un numéro de commande (Numcom) et d'une référence de produit (Refprod) permet de retrouver la quantité commandée de ce produit lors de cette commande.	Il faut créer une relation supplémentaire (dans notre exemple : COMPRENDRE), Elle sera composée d'une clé primaire concaténée (Numcom. Refprod) et de l'attribut Quantité.
<p>Remarques</p> <p>L'attribut Quantité ne pouvait être intégré à aucune relation. Hors, aucun attribut ne peut rester isolé dans un modèle relationnel. La création de la relation COMPRENDRE permet de l'intégrer.</p> <p>De plus, les deux attributs formant la clé concaténée sont aussi, individuellement, clés étrangères. Il y a donc une liaison entre les relations COMPRENDRE et COMMANDE, COMPRENDRE et PRODUIT.</p> <p>Toutes les relations de ce modèle sont donc liées. Aucune n'est restée isolée. C'est un principe incontournable du modèle relationnel.</p>	



5. Les règles de normalisation

a) L'intérêt des règles de normalisation

Ces règles ont pour but de constituer un modèle évitant les redondances (ou répétitions) inutiles de données, pour limiter les incohérences au sein des données et faciliter leur gestion et leur mise à jour.

En effet, lors de la construction d'un modèle relationnel à partir de données réelles que l'on souhaite gérer, plusieurs choix de modèles sont parfois possibles. Or, le modèle doit exprimer les liens logiques qui existent entre les différents objets. Les choix ne sont pas toujours évidents.

Exemple : l'assistant(e) de gestion ou l'assistant(e) manager souhaite compléter une base de données sur le personnel de l'entreprise. La base contient les coordonnées personnelles de chaque salarié (nom, prénom, adresse, date d'embauche, poste occupé...).

L'assistant(e) de gestion ou l'assistant(e) manager souhaite gérer également les données relatives au service dans lequel travaille chaque salarié.

Faut-il ajouter à la relation SALARIE les attributs correspondant aux services (nom du service, responsable du service, n° de téléphone du service...) ou faut-il créer une autre relation ? Ce type de choix doit être effectué en fonction des règles de normalisation établies pour le modèle relationnel.

b) Les trois formes normales

La première forme normale

Une relation est en première forme normale si :

- la relation a une clé primaire (donc les autres attributs sont en dépendance fonctionnelle avec la clé primaire) ;
- les attributs contiennent une valeur atomique (non divisible). Les attributs à valeur multiple sont interdits.

Exemple : l'assistant(e) de gestion ou l'assistant(e) manager souhaite compléter la base de données relative au personnel avec les diplômes possédés par chaque salarié.

Il/Elle ne peut pas ajouter un attribut diplômes_obtenus à la relation SALARIE car un salarié peut posséder plusieurs diplômes. La relation présenterait une erreur de première forme normale.

Il/Elle doit créer une relation supplémentaire de type DIPLOME qui contiendra une clé primaire concaténée (numéro salarié/diplôme détenu).

La deuxième forme normale

Une relation est en deuxième forme normale si :

- la relation est en première forme normale ;
- les attributs non clés dépendent pleinement de la clé et non d'une partie seulement de celle-ci.

Exemple : l'assistant(e) de gestion ou l'assistant(e) manager souhaite compléter une base de données FACTURE avec la date de facture et le nom du client. Il dispose de la relation suivante : LIGNE_FACTURE (numfact, refprod, Q) La clé primaire est une clé concaténée.

Il/elle ne peut pas ajouter les attributs date_fact et nomcli à cette relation car ils dépendent uniquement du numéro de facture et non pas de la référence du produit. La relation LIGNE_FACTURE (numfact, refprod, Q, date_fact, nomcli) présenterait une erreur de deuxième forme normale.

Il/elle doit décomposer les attributs en deux relations :

- FACTURE (numfact, date_fact, nomcli) ;
- LIGNE_FACTURE (numfact#, refprod, Q).

La troisième forme normale

Une relation est en troisième forme normale si :

- la relation est en deuxième forme normale ;
- les attributs non clés dépendent directement de la clé primaire (ils ne dépendent pas directement d'un attribut non clé).

Exemple : l'assistant(e) de gestion ou l'assistant(e) manager souhaite compléter la base de données avec les adresses des clients.

Il/elle dispose du modèle suivant :

FACTURE (numfact, date_fact, nomcli)

LIGNE_FACTURE (numfact#, refprod, Q).

Il/elle ne peut pas ajouter les attributs ruecli, CPcli, villecli dans la relation FACTURE car ils dépendent directement de l'attribut nomcli et indirectement de la clé primaire numfact. La relation FACTURE (numfact, date_fact, nomcli, ruecli, CPcli, villecli) présenterait une erreur de troisième forme normale.

Il/elle doit créer une relation supplémentaire et ajouter une clé étrangère à la relation FACTURE.

CLIENT (numcli, nomcli, ruecli, CPcli, villecli)

FACTURE (numfact, date_fact, numcli#)

LIGNE_FACTURE (numfact#, refprod, Q)