

Chapitre 3

Le modèle conceptuel des données

1. Introduction

Le modèle conceptuel des données (MCD) introduit la notion d'entités, de relations et de propriétés. Nous allons commencer par voir certains aspects « théoriques » avant de plonger dans la pratique.

Le MCD décrit de façon formelle les données utilisées par le système d'information. La représentation graphique, simple et accessible, permet à un non-informaticien de participer à l'élaboration de ce modèle. Les éléments de base constituant un modèle conceptuel des données sont :

- les propriétés ;
- les entités ;
- les relations.

1.1 Les propriétés

Les propriétés sont les informations de base du système d'information.

Un client possède un numéro de client, un nom, un prénom, habite à une adresse précise, etc. Ces informations élémentaires essentielles sont des propriétés.

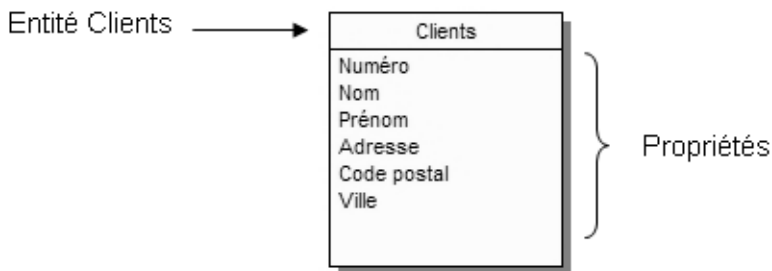
Les propriétés disposent d'un type. Elles peuvent être numériques, représenter une date, leur longueur peut être aussi définie. Par exemple : le nom est une propriété de type alphabétique et de longueur 50, c'est-à-dire que la valeur saisie ne comportera aucun chiffre et ne dépassera pas 50 caractères.

Les types ne sont pas décrits au niveau conceptuel, car ce niveau est trop proche de la définition du système physique. Nous y reviendrons plus tard.

1.2 Les entités ou objets

Comme il est aisé de le constater, les clients sont définis par certaines propriétés (numéro, nom, prénom...). Le fait de les regrouper amène naturellement à créer une entité Clients.

Le symbolisme retenu est le suivant :



L'identifiant

Une de ces propriétés a un rôle bien précis ; c'est l'**identifiant**, nommé aussi la clé.

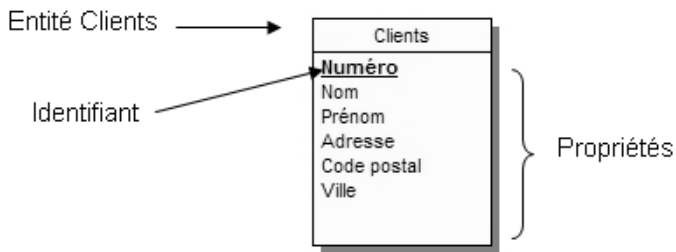
L'identifiant permet de connaître de façon sûre et unique l'ensemble des propriétés qui participent à l'entité. Par exemple, le fait de connaître la ville d'un client permet-il de connaître son nom ? La réponse est non. La connaissance du nom du client permet-elle de connaître sa ville ? La réponse est toujours non, car en cas d'homonymie la confusion entre un Durand Max et un Durand Raymond est totale.

Il faut donc trouver, ou inventer, une propriété, qui lorsque sa valeur est connue, permet la connaissance de l'ensemble des valeurs qui s'y rattachent de façon formelle.

Ainsi, lorsque le numéro du client est connu, son nom, son prénom et toutes les valeurs des autres propriétés qui se rattachent à ce client sont connues de façon sûre et unique.

Au niveau du formalisme, cette propriété se souligne.

Voici le schéma modifié de l'entité Clients :



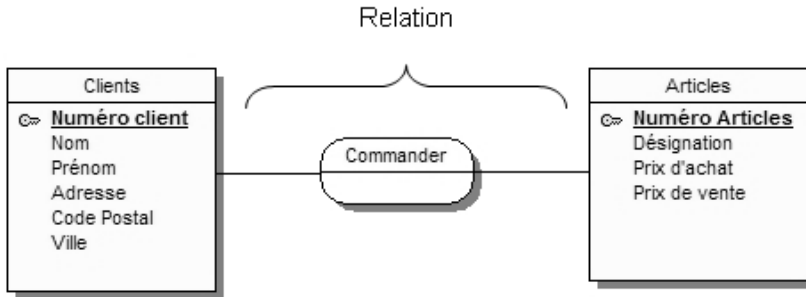
1.3 Les relations ou associations

Nous avons vu que les entités regroupent un ensemble d'informations élémentaires. Les entités sont souvent liées entre elles.

Exemple

Un client peut commander des articles.

Si nous analysons cette phrase, nous distinguons deux entités : clients et articles, et un verbe : commander, qui indique un lien entre clients et articles. Formalisons cette phrase avec Merise.



Voilà la première étape. Première, car le schéma doit être amélioré en incorporant une notion importante : les cardinalités.

1.3.1 Les cardinalités

Les cardinalités expriment le nombre de fois où l'occurrence d'une entité participe aux occurrences de la relation.

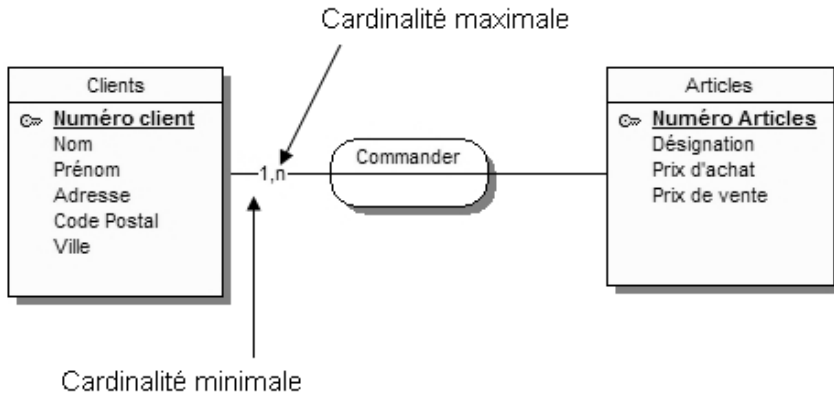
Dans notre exemple, nous pouvons nous poser les questions suivantes :

- Combien de fois au minimum un client peut-il commander un article ?
- Combien de fois au maximum un client peut-il commander un article ?

À la première question, nous pouvons répondre qu'un client, pour être client, doit commander au moins un article.

À la deuxième question, nous pouvons répondre qu'un client peut commander plusieurs articles.

Voici comment symboliser cet état de fait :



Le n représente la notion de « plusieurs » ; ici, nous avons représenté le fait qu'un client peut commander un ou plusieurs articles.

Il faut que nous nous posions les mêmes questions pour l'article :

- Combien de fois au minimum un article peut-il être commandé par un client ?
- Combien de fois au maximum un article peut-il être commandé par un client ?

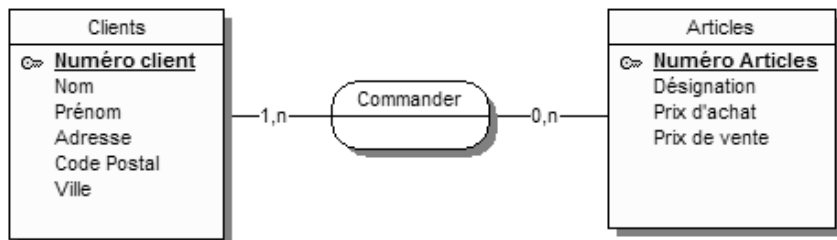
Pour le minimum, nous pouvons reformuler la question de la façon suivante : a-t-on des articles qui peuvent ne jamais être commandés ?

Si nous répondons oui, dans ce cas la cardinalité minimale est 0.

Et pour le maximum : a-t-on des articles qui peuvent être commandés plusieurs fois ?

Nous pouvons espérer que oui. Dans ce cas, la cardinalité maximale est n.

Voici le schéma finalisé :



Définitions

La cardinalité minimale (0 ou 1) exprime le nombre minimum de fois qu'une occurrence d'une entité participe aux occurrences d'une relation.

La cardinalité maximale (1 ou n) exprime le nombre maximum de fois qu'une occurrence d'une entité participe aux occurrences de la relation.

Remarque

Si le maximum est connu, il faut inscrire sa valeur. Par exemple, si dans les règles de gestion le client n'a le droit de commander qu'un maximum de 3 articles en tout et pour tout, dans ce cas-là les cardinalités s'expriment de cette façon : 1,3.

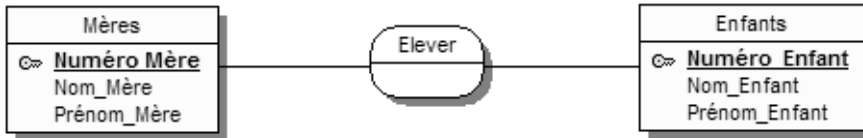
Autre exemple

Modélisons le fait qu'une mère élève des enfants.

Nous avons deux entités : Mères et Enfants.



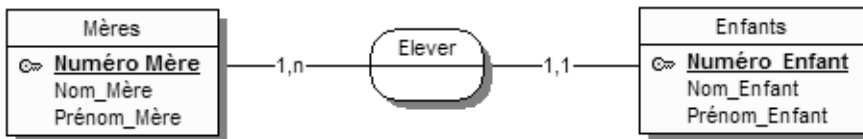
Une relation : Elever.



Des cardinalités :

Une mère peut élever un ou plusieurs enfants.

Un enfant peut être élevé par une et une seule mère.



Remarque

Bien sûr, tout est question d'interprétation. Au sein d'une équipe de développement, il peut y avoir des divergences de points de vue. Pour les cardinalités, il faut être le plus logique possible, se référer aux règles de gestion édictées par le commanditaire de l'application et se rappeler la maxime suivante : qui peut le plus peut le moins.

1.3.2 Les relations porteuses

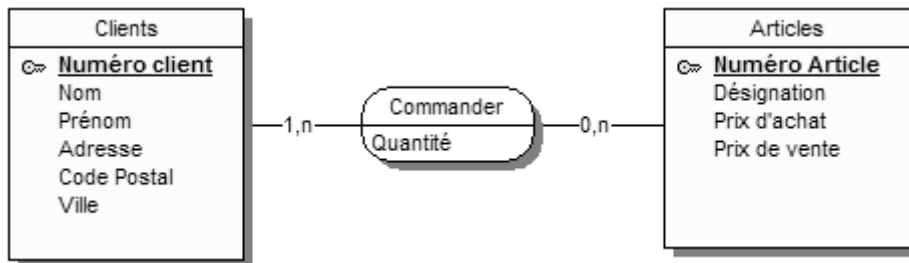
Définition

Une relation est dite porteuse lorsqu'elle contient des propriétés.

Exemple

Nous voulons connaître la quantité d'articles commandés par client. Nous nous rendons compte qu'il faut utiliser une nouvelle propriété : Quantité. Cette propriété dépend de Clients, d'Articles ou des deux ? La bonne réponse est que Quantité dépend des deux entités.

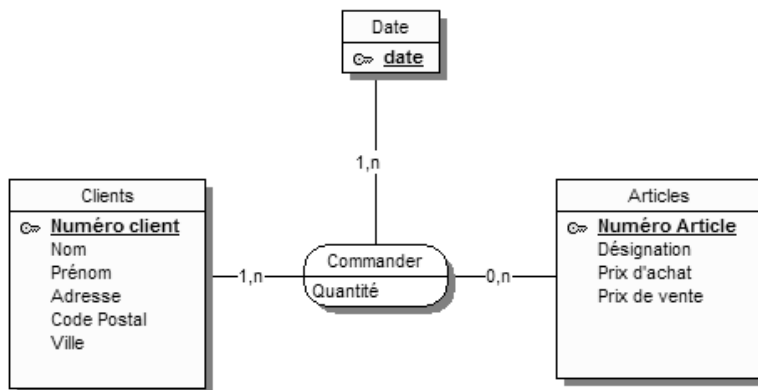
Voici le modèle conceptuel correspondant :



Nous pouvons interpréter ce schéma de la façon suivante : le client X a commandé la quantité Y d'articles Z. Si nous désirons connaître la date d'achat, il nous suffit de créer une entité Date à la relation Commander.

Remarque

Une relation faisant intervenir deux entités est dite **binnaire**.



Remarque

Une relation faisant intervenir trois entités est dite **ternaire**. Dans certains ouvrages, elle est dite **tri-pattes**.

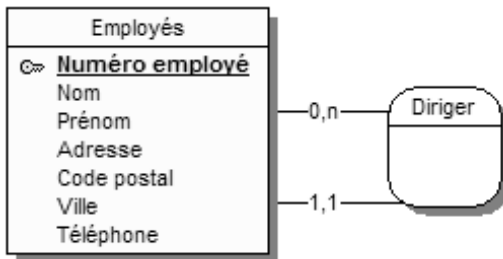
1.3.3 Les relations réflexives

Définition

Une relation réflexive est une relation d'une entité sur elle-même.

Exemple

Nous désirons modéliser le fait qu'un employé peut diriger d'autres employés.



Ce schéma signifie qu'un employé peut diriger zéro ou plusieurs personnes et qu'un employé est dirigé par un et un seul autre employé.

1.4 Les règles d'usage

- Toute entité doit comporter un identifiant.
- Toutes les propriétés de l'entité dépendent fonctionnellement de l'identifiant. C'est-à-dire que connaissant la valeur de l'identifiant, nous connaissons de façon sûre et unique la valeur des propriétés associées. Si nous recherchons le client numéro 5, nous devons récupérer le nom et le prénom du client numéro 5, et pas ceux d'une autre personne.
- Le nom d'une propriété ne doit apparaître qu'une seule fois dans le modèle conceptuel des données. Si nous établissons une entité Clients et une entité Prospects, nous ne devons pas retrouver la propriété Nom dans les deux entités. Il faut préférer les dénominations suivantes Nom_client et Nom_prospect.
- Les propriétés résultantes d'un calcul ne doivent pas apparaître dans le modèle conceptuel des données.