

# Modèle relationnel 1/2 : Concevoir une base de données

## 1. Introduction

### Informations et données

- Les données sont omniprésentes autour de nous.
  - Dans la vie de tous les jours : Identité, relevés bancaires, informations météo, horaires de trains, réseaux sociaux, webmail ...
  - Dans l'entreprise : Clients, produits, commandes, factures...
  - En médecine : Patients, médicaments, diagnostics, statistiques,...
  - ...
- Différence entre donnée et information : une donnée est une information formatée, codée, prête à être utilisée par un ordinateur (un identifiant, un nom, une année, une quantité, un prix, ...)
- Les données :
  - Enjeu stratégique, économique et commercial important.
  - Aide à la prise de décision.



### Les bases de données

- Convertissent l'information en données.
- Structurent et organisent les données.
- Maintiennent les données à jour.
- Sécurisent l'accès aux données.
- Partagent les données.
- Exploitent de gros volumes de données.



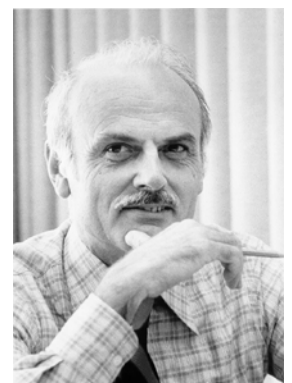
### Accès

- Modèle client/serveur.
- Logiciels d'entreprises.
- Applications web et mobiles.
- Des langages de requête : SQL (Structured Query Language), le plus connu.

```
UPDATE clause { UPDATE country
SET clause   { SET population = population + 1
WHERE clause { WHERE name = 'USA';
               Expression
               Predicate
               Statement
```

### Modèles

- Concevoir un modèle adapté à une situation bien précise est une tâche primordiale.
- Un modèle mal pensé est source d'erreurs difficilement rattrapables.
- Il existe plusieurs modèles de représentation des bases de données.
- Celui que nous allons découvrir est l'un des plus utilisés, appelé modèle relationnel.
- Il a été inventé dans les années 70 par un informaticien travaillant chez IBM : Edgar Frank Codd
- Ce modèle permet de s'adapter à de nombreuses situations.



### **Exercice 1 : Première approche**

Une société d'assurance voudrait prospecter de potentiels clients dans le Loiret. Pour cela elle a besoin de connaître leur adresse et éventuellement des informations sur les voitures qu'ils conduisent :

- Mme Fort habitant à Montargis conduit une Renault électrique Zoé blanche de 2016, immatriculée AB-123-CD .
- Mr et Mme Duprack habitent à Orléans et ont acheté en 2015 une Peugeot 208 neuve essence noire immatriculée AD-234-EF.
- Mme Louise habite à Montargis et vient d'avoir son permis de conduire.
- Mr Dubois vient d'emménager à Cepoy, il possède un vieux combi Volkswagen T3 de 1985 essence de couleur orange, immatriculé AC-456-DC

Il faut structurer ces informations :

1. Organiser ces informations dans un tableau en indiquant le nom des colonnes .
2. Ajouter une nouvelle voiture à Mme Louise, c'est la même que celle de Mme Fort.
3. Ajouter une nouvelle ligne au tableau.
4. Ajouter deux nouvelles voitures à Mme Fort.
5. Mme Fort déménage à Gien, modifier la table en conséquence.
6. Mr Duprack quitte le département, modifier la table.
7. Une autre cliente appelée aussi Mme Fort vient d'emménager à Montargis, elle possède une Fiat 500 diesel verte de 2010, immatriculée DA-620-BF.Modifier la table.

### **Remarques :**

- Il est difficile de structurer une grande quantité d'information en une seule table, même pour un ordinateur:
  - Informations redondantes : Plusieurs lignes avec le même client...
  - Ajout d'information incomplète : nouvelle adresse, nouvelle voiture,...
  - Suppression d'information: déménagement, vente de la voiture...
  - Problèmes de cohérence en cas de mise à jour : clients avec le même nom, plusieurs lignes à mettre à jour pour une modification.

## **Exercice 2 : Deuxième approche**

Nous allons utiliser deux tableaux :

- Une pour les clients
- Une pour les voitures

1. Construire le tableau VOITURE. Quelle information permet d'identifier de façon unique une voiture ?

1. Construire le tableau CLIENT en ajoutant une information qui permettra d'identifier un client de façon unique.

### **Remarques :**

- Ce découpage permet une manipulation plus aisée en cas de modification (par exemple un changement d'adresse)
- Il évite certaines redondances (plusieurs fois la même information)
- Chaque table doit comporter une information qui identifie chaque élément de manière unique. On appelle cela une clé.
- On perçoit néanmoins ici que certaines colonnes d'une même table dépendent d'une autre colonne de la même table. On appelle cela des dépendances fonctionnelles et lorsque l'on conçoit une base de données, on doit les identifier.
  - Par exemple, la connaissance d'un modèle de voiture implique la connaissance de la marque.

### Exercice 3 : Association

Nous allons créer un troisième tableau POSSEDER . Il va associer les clients et les voitures.

1. Construire ce tableau en gardant le minimum d'informations.

1. Par combien de clients peut être conduite une voiture ? entre 1 et N.

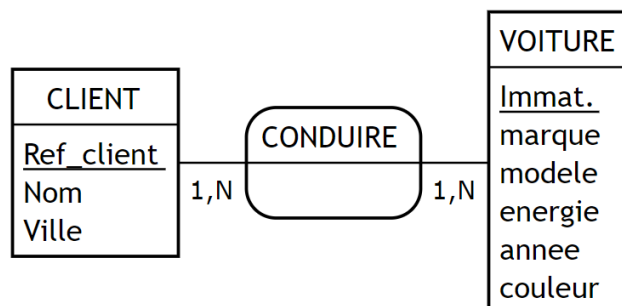
1. Combien de voitures un client conduit-il ? Entre 1 et N.

#### Remarques:

- Seules les clés des tables précédentes sont nécessaires pour établir cette association.
- Les nombres déterminés aux questions 2 et 3 sont appelées cardinalités.

## 2. Le modèle conceptuel des données (MCD)

Voici une représentation du modèle conceptuel de données de la situation précédente:



## Vocabulaire

**Entité** : Un objet du monde réel à modéliser (client, produit, article, ...)

- CLIENT et VOITURE sont des entités

**Attribut(ou propriété)** : Donnée élémentaire d'une entité.

- Nom est un attribut de l'entité CLIENT
- energie est un attribut de l'entité VOITURE

**Occurrence** : Un n-uplet qui définit un élément d'une entité avec ses attributs.

- Le triplet (Dubois, Cepoy, 5) est une occurrence de l'entité CLIENT

**Association** : Un lien reconnu ou possible entre les entités. Une association peut avoir des attributs.

- l'association CONDUIRE est le lien entre les entités VOITURE et CLIENT

**Identifiant** : Un identifiant est un attribut qui sert à identifier de manière unique les occurrences d'une entité.

- Immat. est l'identifiant de l'entité VOITURE
- Réf\_client est l'identifiant de l'entité VOITURE

**Cardinalité** : La cardinalité d'une association est constituée d'un minimum et d'un maximum

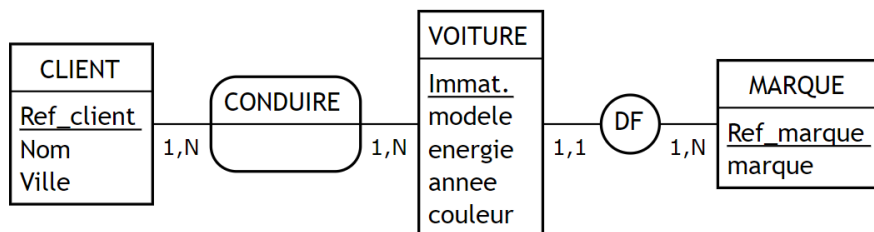
- Minimum : nombre minimum de fois qu'une occurrence de l'entité participe aux occurrences de l'association, 0 (optionnel) ou 1 (obligatoire)
- Maximale : nombre maximum de fois qu'une occurrence de l'entité participe aux occurrences de l'association, 1 (unique) ou n (multiple)
  - La cardinalité 1,  $N$  de l'entité CLIENT pour l'association CONDUIRE signifie qu'un même client conduit une ou plusieurs voitures.

### Exercice 4 : Questions de vocabulaire

1. Citer un attribut de l'entité
1. Citer une occurrence de l'entité
1. Citer tous les attributs de l'entité
1. Quelle est la cardinalité de l'entité
1. Que signifie cette cardinalité ?

### Exercice 5 : Dépendance fonctionnelle.

Voici un modèle amélioré de notre MCD.



1. Quelle nouvelle entité a été créée ?
1. Quels sont ses attributs ?
1. Quel est son identifiant ?
1. Qu'a-t-on modifié pour l'entité VOITURE :
1. Que signifie la cardinalité 1, 1 sur le schéma ?

#### Remarques :

- Nous venons d'identifier une dépendance fonctionnelle (DF). C'est une association avec une cardinalité 1, 1.
- On évite ainsi des redondances (des répétitions inutiles)
- L'attribut `marque` aurait pu servir d'identifiant, mais en général, on utilise plutôt une suite de chiffres.

## Méthodologie de conception

1. Recueil des données : identifier toutes les données élémentaires à stocker (le dictionnaire des données)
2. Identifier les dépendances fonctionnelles.
3. Regrouper les données en entités cohérentes.
4. Donner les associations entre les entités.
5. Etablir les cardinalités.

#### Conventions :

- Les noms des entités et associations s'écrivent en majuscule.
  - Une entité est désignée par un nom commun
  - Une association est désignée par un verbe
- Chaque attribut porte un nom différent.
- Le premier attribut d'une entité est son identifiant.

### Exercice 6 : La bibliothèque

Dans une bibliothèque municipale, il y a :

- Des usagers, identifiés par leur nom, leur prénom, leur commune d'habitation.
- Des livres, identifiés par leur titre et leur(s) auteur(s) eux-mêmes identifiés par leur nom et leur prénom.
- Des emprunts avec une date de retour. On veut construire un MCD adapté à cette situation.

1. Recenser un dictionnaire des données et donner un nom aux propriétés.
2. Construire un MCD en identifiant les entités, les associations et les cardinalités.

Réponse :

- 1.
- 2.

### **Exercice 7 : Les commandes**

Un magasin souhaite mettre en place une base de données pour les commandes de produits de ces clients. Voici les informations qu'elle souhaite modéliser :

- Un client est identifié par son nom, son prénom et sa ville d'habitation.
- Une commande a une date et un montant et comporte des produits qui ont un nom et un prix unitaire.

Construire un MCD en faisant apparaître les entités, les associations et les cardinalités.

Réponse:

### 3. Le modèle logique des données (MLD)

Voici maintenant une représentation logique de notre situation après l'exercice 5. Il s'agit des tables qui seront ensuite créées pour la gestion de la base de données.

**CLIENT** ( Ref\_client, Nom, Ville )  
**CONDUIRE** ( Ref\_client, Immat. )  
**VOITURE** ( Immat., modele, energie, annee, couleur, *Ref\_marque* )  
**MARQUE** ( Ref\_marque, marque )

- On franchit une étape vers la machine en précisant maintenant le formatage des données.
- Les entités ou associations sont désormais appelées des relations(ou des tables)
  - Cette situation se modélise avec 4 tables : CLIENT , VOITURE , CONDUIRE , MARQUE
- Les identifiants sont désormais appelés des clés. On peut les retrouver dans plusieurs tables sous des statut différents
  - Les clés sont ici : ref\_client , immat.



## Les contraintes d'intégrité

Pour garantir la cohérence des données de la base, on doit établir des contraintes sur celles-ci, c'est à dire des propriétés logiques qui restent vrai à tout instant. Ce sont les contraintes d'intégrité.

### Contraintes de domaine :

- Elles déterminent le type de valeur de chaque attribut et l'ensemble de définition de celui-ci. On retrouve ainsi des types de données que l'on rencontre en programmation mais aussi des types plus spécifiques. Pour la suite, on leur donnera les noms génériques suivants, même s'ils sont variés selon le système technique de base de données utilisé.
  - `Int` : Entier signé
  - `Float` : Nombre flottant
  - `String` : Chaîne de caractères
  - `Bool` : Valeurs `True` et `False`
  - `Date` : Date au format jour/mois/année.
  - `Time` : Instant au format heure:minute:seconde.
- Le choix des domaines est très important, car il doit répondre à deux impératifs:
  - Représenter exactement et sans perte d'informations toutes les valeurs possibles pour un attribut.
  - Limiter le plus possible la saisie de valeurs incohérentes.

#### Exemple:

- L'attribut `annee` de la relation `VOITURE` est de type entier. Mais cela ne suffit pas à éviter les valeurs incohérentes du type `-1` ou une année supérieure à l'année en cours.

### Contraintes d'entité :

- Elle permet de s'assurer que chaque élément d'une relation est unique. Il s'agit donc d'un identifiant.
- On l'appelle la clé primaire de la relation.
- C'est l'attribut qui est souligné dans le modèle logique de données.

#### Exemple :

- `ref_client` est la clé primaire de la relation `CLIENT`.

### Contraintes de référence :

- Une clé primaire permet aussi de servir de référence dans une autre relation.
- Lorsque c'est le cas, on dit que c'est une clé étrangère de la relation dans laquelle elle sert de référence.
- Des clés étrangères peuvent apparaître plusieurs fois dans une table, ce qui garantit la cohérence des cardinalités.
- Cette contrainte permet d'éviter le rajout de valeurs incohérentes dans cette même relation, la suppression non voulue de certains éléments.

#### Exemples :

- La relation `CONDUIRE` est constituée de deux clés étrangères : `ref_client` et `immat.`. Dans cette relation peuvent apparaître plusieurs fois le même conducteur (puisque'il peut conduire plusieurs voitures) ou une même voiture (puisque'une voiture peut être conduite par plusieurs conducteurs).
- La relation `VOITURE` contient une clé étrangère `immat.`. Dans cette relation, plusieurs voitures peuvent être de la même marque.

### Contraintes utilisateurs :

- Ces contraintes sont spécifiques à chaque situation.
- On les appelle aussi des contraintes métiers.
- Ce sont celles que l'on ne peut pas exprimer sous forme d'autres contraintes.

#### Exemples :

- Pour l'attribut `annee` de la relation `VOITURE`, on peut préciser qu'il doit s'agir d'un entier strictement positif et inférieur ou égal à l'année en cours pour éviter les valeurs absurdes comme `-1` ou `2142`.
- Pour l'attribut `immat.`, on peut spécifier qu'il s'agit d'une chaîne de caractères qui doit obligatoirement comporter deux lettres, un tiret, trois chiffres, un tiret et deux lettres.

### **Exercice 8 : La bibliothèque**

1. Ecrire le MLD de l'exercice 6.

1. Citer au moins une contrainte dans chaque cas:

Contraintes de domaine:

- 

Contraintes d'entité :

- 

Contraintes de référence :

- 

Contraintes utilisateurs :

-

### **Exercice 9 : Les commandes**

1. Ecrire le MLD de l'exercice 7.

1. Citer au moins une contrainte dans chaque cas:

Contraintes de domaine:

Contraintes d'entité :

Contraintes de référence :

Contraintes utilisateurs :