



Contenu : - Modèle relationnel : relation, attribut, domaine, clef primaire, clef étrangère, schéma relationnel
- Base de données relationnelle

Capacités attendues : Identifier les concepts définissant le modèle relationnel.
Savoir distinguer la structure d'une base de données de son contenu.
Repérer des anomalies dans le schéma d'une base de données.

I. Introduction : bases de données - SGBD

Nous avons travaillé l'an passé sur le traitement des données, notamment en important des fichiers au *format xls ou csv*, en les travaillant en langage python puis en les exportant.

Dans la vie courante, les entreprises, l'état, les associations... sont amenés à gérer dans quantités de données qui sont difficilement exploitables sous format csv soit de par leur taille trop conséquente soit par leur structure même qui comporte des relations entre différentes données.

Il existe un outil complètement adapté à ce type de données, **les bases de données**.

Il existe plusieurs types de bases de données : Hiérarchiques, relationnelles... Ce sont ces dernières que nous allons étudier. Ce sont aussi les plus répandues. Dans les bases de données relationnelles, les données sont stockées sous forme de table.

Ce sont l'apparition des disques durs qui ont permis d'envisager le stockage des données dans les années 50. C'est lors du programme Apollo en 1960 que l'idée de la base de données a été lancée : le but était de collecter des données afin de permettre d'aller sur la lune avant la fin de la décennie.

Mais les bases de données ont pris leur envol avec l'émergence d'internet et l'apparition du big data, gigantesque collecte de données qu'il faut stocker, modifier, traiter.

Définition (base de données) :

Une base de données est ensemble d'informations structurées mémorisées sur un support persistant.

Pour traiter ces données, on utilise des Systèmes de Gestion de Bases de Données (SGBD).

Définition (SGBD) :

Un **système de gestion de base de données (SGBD)** est un système informatique qui assure la gestion de l'ensemble des informations stockées dans une base de données. Il prend en charge, notamment, les deux grandes fonctionnalités suivantes :

- **Accès aux fichiers** de la base, garantissant leur intégrité, contrôlant les opérations concurrentes, optimisant les recherches et mises à jour.
- **Interactions** avec les applications et utilisateurs, grâce à des langages d'interrogation et de manipulation à haut niveau d'abstraction.

Parmi les logiciels les plus connus il est possible de citer : MySQL, PostgreSQL, SQLite, Oracle Database, Microsoft SQL Server, Firebird ou Ingres.

Attention à bien différencier bases de données et SGBD : Les bases de données sont un ensemble de propositions vraies stockées et accessibles. On accède à ces données via un langage, le langage SQL. Le SGBD est l'outil qui permet, entre autres, d'accéder à ces données.

En fait, la base de données est au niveau logique (c'est un peu la cargaison) et le SGBD au niveau physique (c'est le moteur qui gère l'ensemble). Mais le SGBD fait bien plus qu'accéder aux données.

II. Propriétés des SGBD

- Les SGBD assurent l'écriture, la lecture et la modification des BDD
- Les SGBD assurent l'indépendance entre le traitement des données et les données directement perceptibles dans le monde réel : Une modification de la base de données, de sa structure ne doit pas impacter le rendu réel de la base de données
- Les SGBD s'assurent de la non redondance des données
- Les SGBD gèrent le partage des données : Elles assurent la possibilité à plusieurs utilisateurs l'accès simultané à la base de données. Mais elles permettent aussi la modification simultanée et la cohérence des résultats affichés pour les utilisateurs qui consultent la base. On parle d'accès concurrent.
- Les SGBD assurent la sécurisation de la Bdd et gèrent les niveaux d'accès des utilisateurs : De simples consultants à administrateur.
- Les SGBD doivent être capables d'assurer la persistance du système en cas de panne. C'est pour cela que les BDD sont stockées sur plusieurs supports, chaque modification devant être répercutée sur tous les supports.
- Les SGBD s'appuient sur des langages (LMD : Langage de manipulations des données) qui permettent l'accès rapide et efficace aux requêtes des utilisateurs.

III. Le modèle relationnel

III.1. L'entité.

Les bases de données relationnelles ne sont qu'un des types de bases de données, mais c'est le type le plus utilisé. C'est en 1970 qu'elles furent définies par un informaticien d'IBM, Edgar Codd.

Le modèle relationnel est fondé sur la notion **de relation** : on travaille avec un tableau à deux dimensions.

On appelle entité un objet pouvant être identifié distinctement. Chaque entité est donc unique et décrite par un ensemble de propriétés appelées **attribut**. Plusieurs attributs permettent d'identifier de manière unique l'entité, on parle alors d'identifiants (ou de clés).

La première ligne est une ligne d'**entête**, où chaque composante est un **attribut**.

S'il y a 6 attributs, chaque ligne contiendra donc 6 renseignements.

Chaque attribut de la relation est associé à un **domaine** : Chaîne de caractère, entier... Ce domaine peut être infini (chaîne de caractère pour caractériser un nom) ou fini (date de sortie d'un film de 1900 à nos jours ...)

Le SGBD, s'assure que chaque élément de la relation est bien unique, que le domaine de chaque attribut est bien respecté. Chaque ligne de ce tableau est appelée un **enregistrement** (il s'agit d'un n-uplets de valeurs).

Pour s'assurer de la non redondance d'un enregistrement, on utilise **une clé primaire** qui est un attribut qui permet d'identifier de façon certaine un enregistrement.

Exemple : Dans une médiathèque, les livres sont répertoriés sous forme de base de données

Voici un extrait sous forme de tableau

Livres (Auteur, Titre, Année, Pays, Genre)					
Auteur	Titre	Année	Pays	Genre	Entête
Hugo	Hernani	1830	France	Théâtre	
King	Nuit noire, étoile morte	2010	USA	S.F.	
King	Misery	1999	USA	S.F.	
De Vigan	No et Moi	2010	France	Roman	
Claudiel	Le soulier de Satin	1929	France	Littérature	Ligne – enregistrement – N-Uplet
Claudiel	L'enquête	2010	France	S.F.	
Bertholon	Twist	2010	France	Thriller	
Jackson	Vengeance	2018	USA	Thriller	
Simmons	Vengeance	2003	USA	Thriller	

Colonne – Attribut (nom et type) Valeur Tableau 1

Remarques :

Une **TABLE** = une **RELATION**

On pourra faire référence à un attribut d'une relation en utilisant la notation : *Relation.Attribut*

De plus,

1. Une base de données est constituée d'un ensemble de relations (voir §III.2.)
2. Chaque relation contient les données relatives à des entités de même nature
3. Chaque ligne d'une relation reprend les données relatives à une entité
4. Chaque colonne d'une relation décrit une propriété commune des entités
5. Les lignes d'une relation sont distinctes
6. On évite de stocker les informations qui peuvent être calculées

Exercice sur le tableau 1 de la page 2 :

1. Quels sont les attributs de ce tableau ? **5 : Auteur, Titre, Année, Pays, Genre**
2. Combien y a-t-il d'enregistrements ? **9**
3. Quel est le domaine de chaque attribut ? Préciser s'il est fini ou non.

Auteur : chaîne de caractères (non fini), Titre : chaîne de caractères(fini), Année : entier (fini), Pays : chaîne de caractères (non fini), Genre : chaîne de caractères (non fini)

4. Peut-on définir, à l'aide des attributs présents dans le tableau, une clé primaire ?

Non, car les valeurs de chaque attribut ne sont pas uniques !

Remarque : On peut définir une clé primaire à l'aide de la combinaison de deux attributs (mais on ne l'utilisera pas en terminale...).

5. Comment pourrait-on identifier de manière unique chaque enregistrement ?

En rajoutant une colonne "identifiant"

Chaque livre possède un identifiant unique appelé ISBN, noté sur la couverture du livre. Ce nombre pourrait servir de clé primaire. On préférera cependant, dans un souci de généralisation des pratiques, créer un identifiant pour chaque enregistrement. Cet identifiant est incrémenté à l'ajout d'une ligne dans la table.

id	Auteur	Titre	Année	Pays	Genre
1	Hugo	Hernani	1830	France	Théâtre
2	King	Nuit noire, étoile morte	2010	USA	S.F.
3	King	Misery	1999	USA	S.F.
4	De Vigan	No et Moi	2010	France	Roman
5	Claudiel	Le soulier de Satin	1929	France	Litterature
6	Claudiel	L'enquête	2010	France	S.F.
7	Bertholon	Twist	2010	France	Thriller
8	Jackson	Vengeance	2018	USA	Thriller
9	Simmons	Vengeance	2003	USA	Thriller

Exercice

Créer une table de votre choix avec au moins 5 attributs et 10 enregistrements. Préciser la clé primaire.

III.2. Modèle conceptuel de données : les relations

Le modèle relationnel, c'est :

- Une *structure unique*, la **relation** (ou table)
- Des *contraintes* qui définissent des formes normales, évitant les défauts de conception
- Des *langages*, concrétisés en pratique par **SQL**

Qu'est-ce qu'une relation ?

Notion mathématique : Étant donné un ensemble d'objets \mathcal{O} , une relation (binaire) sur \mathcal{O} est un sous-ensemble du produit cartésien $\mathcal{O} \times \mathcal{O}$.

Dans notre contexte, les "objets" sont des valeurs élémentaires (ou atomiques, comme les entiers I, les flottants F, les chaînes de caractères S).

L'ensemble des paires constituées par exemples des noms de Titre et de leur année est une relation sur $S \times I$.

Définition : relation

Une relation de degré n sur les domaines A_1, A_2, \dots, A_n est un sous-ensemble fini du produit cartésien $A_1 \times A_2 \times \dots \times A_n$

Schéma de relation

On peut décrire une relation par :

- Le nom de la relation
- Un nom (distinct) pour chaque dimension, dit nom d'attribut
- Le domaine de valeur de chaque dimension

C'est le *schéma de la relation*, de la forme $R(A_1 : D_1; A_2 : D_2; \dots; A_n : D_n)$

Exemple : écrire le schéma de la relation du tableau 1 :

Livres (Auteur : string, Titre : string, Année : integer, Pays : string, Genre : string)

Ou plus simplement Livres (Auteur, Titre, Année, Pays, Genre)

Notions et vocabulaire

Terme de la représentation par table	Terme du modèle
Table	Relation
Ligne	n-uplet
Nom de colonne	Nom d'attribut
Cellule	Valeur d'attribut
Type	Domaine

III.3. Réalisation d'un MCD (modèle conceptuel de données)

Dans le tableau précédent, plusieurs informations sont répétées : Il y a plusieurs fois le même genre, le même auteur, la même année ...

On va essayer de simplifier la vision de notre table en en créant plusieurs, où il n'y aura aucune *redondance*. On va créer des **LIENS** entre les différentes tables (appelées entités). Pour créer ces entités, on va s'appuyer sur les attributs présents dans notre table globale pour ne pas oublier de renseignements.

Sur notre modèle on ajoute deux informations : Le prénom et l'année de naissance de l'auteur.

id	Auteur	Prenom_auteur	Année_naissance	Titre	Année	Pays	Genre
1	Hugo	Victor	1802	Hernani	1830	France	Théâtre
2	King	Stephen	1947	Nuit noire, étoile morte	2010	USA	S.F.
3	King	Stephen	1947	Misery	1999	USA	S.F.
4	De Vigan	Delphine	1966	No et Moi	2010	France	Roman
5	Claudé	Paul	1868	Le soulier de Satin	1929	France	Littérature
6	Claudé	Philippe	1962	L'enquête	2010	France	S.F.
7	Bertholon	Delphine	1976	Twist	2010	France	Thriller
8	Jackson	Lisa	1952	Vengeance	2018	USA	Thriller
9	Simmons	Dan	1948	Vengeance	2003	USA	Thriller

1. Peut-on changer de clé primaire ?

Non, les valeurs des autres attributs ne sont pas uniques

Essayons de créer deux tables qui nous évitent les répétitions et prennent bien en compte tous les renseignements présents dans le premier tableau.

2. Pour les auteurs, de quels renseignements a-t-on besoin ?

Id_auteur, Nom_auteur, Prenom_auteur et Date_de_naissance

3. Que va-t-on choisir comme clé primaire ?

Id_auteur

4. Écrire une table AUTEURS sans les doublons :

Id_auteur	Nom_auteur	Prenom_auteur	Date_de_naissance
1	Hugo	Victor	1802
2	King	Stephen	1947
4	De Vigan	Delphine	1966
5	Claudel	Paul	1868
6	Claudel	Philippe	1962
7	Bertholon	Delphine	1976
8	Jackson	Lisa	1952
9	Simmons	Dan	1948

Voici une façon synthétique de représenter les attributs de cette table, où chaque attribut devient un champ de l'entité AUTEURS.

5. Écrire la relation AUTEURS

AUTEURS(Id_auteur, Nom_auteur, Prenom_auteur, Date_de_naissance)

On parle de l'entité **AUTEURS**. *Id_auteur* est la clé primaire et fait partie des quatre champs qui décrivent l'entité **AUTEURS**.

AUTEURS
Id_auteur
Nom_auteur
Prenom_auteur
Date_de_naissance

LIVRES
Id_livre
Titre
Année_Publication
Auteur
Genre
Pays

Pour la relation **LIVRES**, nous obtenons :

Nous sommes amenés à créer une clé primaire afin d'identifier de manière unique chaque ouvrage de la table.

Le lien entre la table **LIVRES** et **AUTEURS** se fait avec le nom de l'auteur. Il y a ici un problème. On a deux Claudel dans notre table. Pour ne pas avoir de problème, on va s'appuyer sur la clé primaire de la table **AUTEURS** qui par définition identifie de manière unique chaque élément de la table. On va donc remplacer le champ *Auteur* par *Id_auteur*, clé primaire de la table **AUTEURS**. Une telle clé sera appelée **clé étrangère** dans la table **LIVRES** et sera notée précédée d'un dièse.

LIVRES
Id_livre
Titre
Année_Publication
#Id_auteur
Genre
Pays

La relation LIVRES peut donc s'écrire :

LIVRES(Id_livre, Titre, Année_Publication, #Id_auteur, Genre, Pays)

III.4. Associations

Une association définit un lien sémantique entre des entités. Elle permet de traduire une partie des règles de gestion qui n'ont pas été satisfaites par la simple définition des entités.

L'association est à minima caractérisée par :

- Un **nom** : généralement on utilise un verbe définissant le lien entre les entités
- 2 **cardinalités** : elles sont présentes sur les deux extrémités du lien. Chaque cardinalité est un couple de valeurs (mini, maxi) qui traduisent 2 règles de gestion (une par sens)

Remarque : une association peut aussi avoir des attributs.

On peut représenter une association par le schéma suivant :



Dans l'exemple du cours, on définit une association "**écrire**" entre les entités **AUTEURS** et **LIVRES**. Ce lien possède 2 cardinalités (1,n) et (1,1) qui traduisent les deux règles de gestion suivantes :

- 1,n : Un auteur peut avoir écrit entre 1 et une infinité (noté n) de livres de l'entité **LIVRES**
- 1,1 : Un livre a été écrit par un et un seul auteur de l'entité **AUTEURS**

Remarque : Les cardinalités les plus répandues sont les suivantes : **0,n ; 1,n ; 0,1 ; 1,1**

On va utiliser le [site mocodo](https://www.mocodo.net) (<https://www.mocodo.net>). Une fois sur le site, vous avez un exemple qui vous est donné et qui permet d'appréhender assez aisément la syntaxe à utiliser pour créer votre modèle conceptuel de données.

Animaux

PEUT VIVRE DANS, 1N ESPÈCE, 1N ENCLOS: nb. max. congénères
 ENCLOS: num. enclos
 OCCUPE, 1N ANIMAL, 1N PÉRIODE, 1N ENCLOS
 PÉRIODE: date début, _date fin

ESPÈCE: code espèce, libellé
 DF, 0N ESPÈCE, _11 ANIMAL
 ANIMAL: nom, sexe, date naissance, date décès
 A MÈRE, 01 ANIMAL, 0N> [mère] ANIMAL

PEUT COHABITER AVEC, 0N ESPÈCE, 0N [commensale] ESPÈCE: nb. max. commensaux
 :
 A PÈRE, 0N ANIMAL, 0N> [père présumé] ANIMAL
 :

PEUT VIVRE DANS (code espèce, num. enclos, nb. max. congénères)
OCCUPE (code espèce, nom, date début, date fin, num. enclos)
PÉRIODE (date début, date fin)
ESPÈCE (code espèce, libellé)
ANIMAL (code espèce, nom, sexe, date naissance, date décès, code espèce mère, nom mère)
PEUT COHABITER AVEC (code espèce, code espèce commensale, nb. max. commensaux)
A PÈRE (code espèce, nom, code espèce père présumé, nom père présumé)

Exercice : Réaliser le MCD de l'exemple du cours.