

# CHAPITRE 1

---

## *Introduction aux bases de données*

# Contenu du chapitre 1

---

- *Pourquoi utiliser une bases de données?*
- *Définitions et objectifs d'un SGBD*
- *Niveaux d'abstraction des données*
- *Méthodes de modélisation d'une BD*
- *Modèles de structuration des données*
- *Structure globale d'un Système de Gestion de Base de Données.*

# ***Pourquoi utiliser une base de données?***

---

**La démarche classique consiste à réaliser pour un type d'information, un programme d'application.**

Une entreprise doit conserver un volume élevé d'information:

- noms, adresses, salaire, adresse des fournisseurs, quantités, prix des items, bilan financier, etc.

Ces informations se retrouvent dans différents systèmes de traitement de fichiers.

- Système de gestion des stocks, système de facturation, système de préparation de paie, programme de gestion de personnel, etc.

**Exemple d'un système décentralisé:** (Schéma en classe (figure 1.1))

# ***Pourquoi utiliser une base de données?***

---

## **Exemple d'un système décentralisé (suite)**

Pour obtenir une information, l'employé doit:

1. *déterminer le système à consulter*
  2. *trouver la bonne personne concernée.*
- Perte de temps

De plus, certaines informations sont souvent conservées en plusieurs endroits.

- Duplication de données
- Gaspillage au niveau du volume de fichiers.

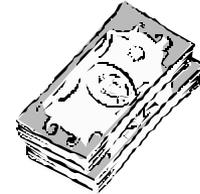
# Pourquoi utiliser une base de données?

---

Résumons le système décentralisé...

Principaux problèmes de ce système:

- *Redondance de certaines informations*
- *Ne peut répondre rapidement aux demandes d'information provenant de fichiers multiples.*
- *Coûts élevés pour les modifications (plusieurs systèmes)*



Avec le temps, il y aura...

- *Accroissement inutile de:*
  - de l'ensemble des fichiers;
  - de la taille des fichiers;
  - des temps d'accès.
- *Code développé par différents programmeurs et écrits dans différents langages.*
- *Formats de fichiers différents*
- *Inconsistance des données*
- *Lourd à supporter*



# Inconvénients des système de traitement de fichiers

---

## Redondance et inconsistance des données

- *Informations identiques répliquées dans plusieurs fichiers.*

*Ex: Institution financière*

- Adresse et téléphone d'un employé
  - ◆ *dans le fichier du système de paie*
  - ◆ *dans le fichier de gestion du personnel*
- *Accroissement inutile :*
  - ◆ *de la taille des fichiers;*
  - ◆ *des temps d'accès.*
- *Risque d'inconsistance des données si le changement d'adresse ne s'effectue pas dans les deux fichiers.*

# Inconvénients des système de traitement de fichiers

---

## Difficulté d'accès aux données

- *Il faut un programme spécifique pour toute nouvelle demande d'information.*

*Ex : Estimation pour l'augmentation de 10% sur le prix des items vendus du mois dernier.*

- *Le temps d'accès à une requête non prévue peut être très long.*

# Inconvénients des système de traitement de fichiers

---

## Isolement des données

- Les données sont stockées sous différents formats.
  - *Ex: Numéro civique, NAS*
    - ◆ type caractère dans un fichier
    - ◆ type entier dans un autre fichier.
  
- *Grande difficulté d'écrire un programme d'accès général à toute l'information.*

# Inconvénients des système de traitement de fichiers

---

## Multiplicité des remises à jour

- Les traitements concurrents peuvent générer des erreurs.

Ex: Mises à jour d'un compte en même temps  
solde de 400 \$

T1: dépôt de 300 \$

T2: retrait de 500 \$

- si T1 avant T2 : 400\$, 700\$, 200\$      OK
- si T2 avant T1 : 400\$, -100\$, 200\$      ERREUR!!!

- *Nécessite un programme superviseur pour gérer les transactions: Difficile, voir impossible.*

# Inconvénients des système de traitement de fichiers

---

## Sécurité

- *La sécurité des données et les accès non-autorisés ne sont pas garanties.*

Ex: Le personnel ne devrait pas avoir accès au programme de paie.

## Intégrité des données

- *difficulté d'imposer des contraintes*

Ex : Le solde ne doit jamais être inférieur à 0.

# ***Pourquoi utiliser une base de données?***

---

*Solution : une banque de données commune, entièrement centralisée*

➤ *Base de données*

Idéalement, il devrait y avoir...

- *Un seul exemplaire de chaque élément de données*
- *Tous les utilisateurs ont accès aux données en ne communiquant qu'avec la base (sans intermédiaire).*
- *Mesures de protection pour l'information confidentielle*
- *La complexité du stockage ne doit pas être apparente à l'utilisateur.*

***Les SGBD ont été créés pour résoudre tous ces problèmes!!!***

***Schéma en classe (FIGURE 1.3)***



# Définitions d'un SGBD

---

**Un Système de Gestion de Bases de Données est un ensemble de programmes qui :**

- gère un ensemble de fichiers (base de données).
- permet aux utilisateurs d'extraire ou de stocker des données.

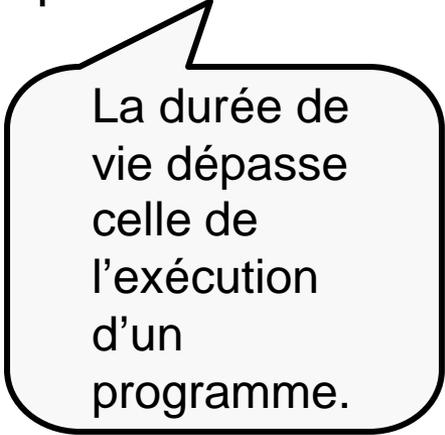
**Ainsi, il permet de:**

- Créer
- Mettre à jour
- Interroger
- Visualiser
- Administrer ... une base de données

# Définitions d'une BD et objectifs d'un SGBD

---

Une BD est une collection de données persistantes utilisées par des systèmes informatiques.



La durée de vie dépasse celle de l'exécution d'un programme.

Les **BD** sont conçues pour traiter de gros volumes d'informations. La gestion de ces données demande :

- *une structuration de ces données.*
- *une rationalisation des procédures d'accès et de traitement.*
- *une assurance de la sécurité des informations stockées dans la BD.*

# Objectifs d'un SGBD

---

- *Masquer les aspects de stockage.*
  - Indépendance logique
  - Indépendance physique
- *Gérer efficacement les données.*
- *Faciliter l'extraction et l'ajout d'information.*
- *Optimiser les traitements de l'information.*
- *Assurer la sécurité des données.*
- *Éviter les conflits lors d'exploitation partagée.*

# Exemples de BD et SGBD

---

## ***Exemples de données :***

- Ensemble des informations relatives à une entreprise:
  - ♦ *liste du personnel;*
  - ♦ *liste des clients;*
  - ♦ *liste des produits.*
- bottin téléphonique;
- contenu d'une cave à vin;
- dossiers médicaux hospitaliers.

## ***Exemples de SGBD***

- PC : Access, Foxpro, Paradox ...
- Gros Systèmes: Oracle, DB2, Sybase, SQL Server, ...
- Freewares et Sharewares : MySQL, MSQL, ...

# Historique

---

## Années 60 :

Systèmes de gestion de **fichiers** plus ou moins sophistiqués, reliés par des pointeurs.

- programmes de la conquête spatiale (APOLLO)

Systèmes de gestion de base de données **réseaux** (travaux de CODASYL) et **hiérarchiques** proches des systèmes de gestion de fichiers.

- TOTAL, IDMS, IDS.2 et IMS.2, etc.

## Années 70:

Codd, mathématicien et chercheur chez IBM, propose le modèle **relationnel** (basé sur le concept de Relation de la théorie des ensembles)

Codd définit l'**Algèbre Relationnelle** (base du SQL)

## Années 80:

**Systèmes de gestion de base de données relationnels**

- SGBD commercialisés à partir de 1980
- ORACLE, SYSDATABASE, DB2, etc.

# Historique (suite)

---

## Années 90:

Les systèmes de gestion de base de données relationnels dominant le marché.

- Extension du relationnel (Oracle 8, DB2 Universal Database, etc.)

Début des systèmes de gestion de bases de données orientés objet.

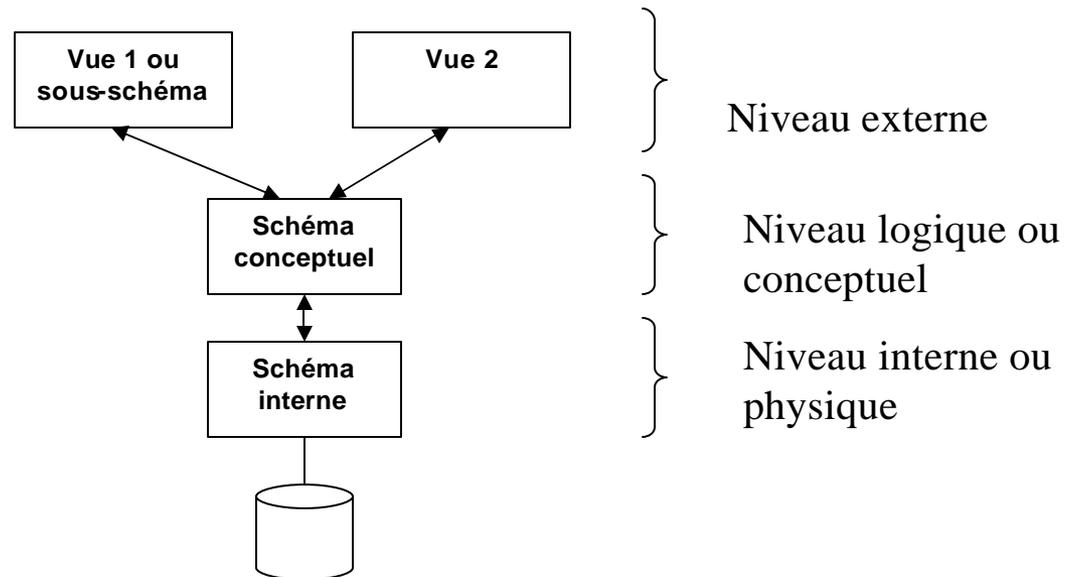
- ObjectStore, GemStone, O2, etc.

4ème génération: Internet, les informations non structurées, le multimédia, etc...

**MC**  **URS.com** (Zone Cours)  
www.mcours.com : Site N° 1 des Cours et Exercices

# Niveaux d'abstraction des données

---



## Niveau externe

- Présente les données sous plusieurs vues.
- Décrit des contenus spécifiques de la BD qui peuvent être des structures complexes.
- Raisonnement

# Niveaux d'abstraction des données

---

## Niveau logique (conceptuel)

- Données considérées de façon sémantique.
  - *i.e. d'après contenu effectif et les relations qui les relient.*
- C'est le niveau qui permet de décrire toute la BD:
  - *Les entités*
  - *Les types de données*
  - *Les relations*
  - *Les opérations des usagers*
  - *Les contraintes*
- Utilise des structures relativement simples qui sont très complexes au niveau physique

# Niveaux d'abstraction des données

---

## Niveau physique (interne) :

- Mode d'inscription sur le support physique (disque) (chapitre 7)
- Structuration décrite de façon très détaillée.
- Chemins d'accès pour la BD

# Niveaux d'abstraction des données

---

## Exemple: Analogie des niveaux avec déclaration de types

*Déclaration d'un enregistrement en C :*

```
typedef struct
{
    char nom[15];
    char rue[25];
    char ville[15];
}Client;
```

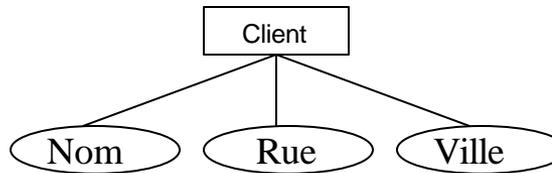
## Au niveau externe

- L'enregistrement est décrit par des vues :
  - vue #1 : tous les clients demeurant dans la ville X
  - vue #2 : tous les clients ayant le même nom de famille
  - vue #3 : tous les clients demeurant sur la rue Y

# Niveaux d'abstraction des données

## Au niveau conceptuel

*L'enregistrement est décrit par son contenu significatif et ses relations.*



## Au niveau physique



- L'enregistrement est décrit comme un bloc d'emplacement mémoires consécutifs (mots ou octets)

Fichier Client

Nom	Rue	Ville	Ptrs
Tremblay	3 <sup>ième</sup> Avenue	Montréal	
Audet	Des Érables	Loretteville	
Voyer	51 <sup>ième</sup> rue	New-York	

# Processus de modélisation d'une BD

---

À quoi sert la modélisation?

Rép: À éviter les problèmes potentiels

Exemple: BD d'un éditeur (titre, année d'édition, nom auteur)

➤ *Problèmes lors d'une insertion:*

- Un livre peut être représenté plusieurs fois de manière différentes (ex: plusieurs dates d'édition).
- Qu'est-ce qui distingue deux livres?

➤ *Problèmes lors d'une modification:*

- Changement du nom d'auteur pour un livre.
- Doit-on faire le changement pour les autres livres du même auteur?

➤ *Problèmes lors d'une destruction:*

- Supprimer un livre qui a été édité 2 fois.
- On peut oublier de détruire la deuxième édition.

# Processus de modélisation d'une BD

---

Ce que la modélisation nous apporte...la simplification!

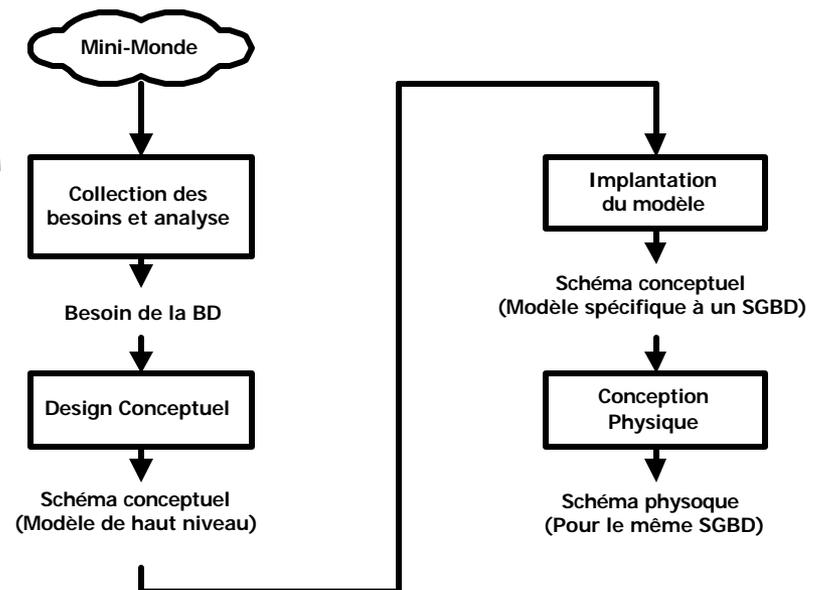
Exemple (suite):

- *Il faut représenter auteurs et livres dans deux listes différentes:*
  - Livre (titre, année d'édition)
  - Auteur(nom auteur)
  
- *Il faut conserver le lien entre les auteurs et les livres sans redondances!*

# Processus de modélisation d'une BD

## 1ère étape : Collection des besoins et analyse

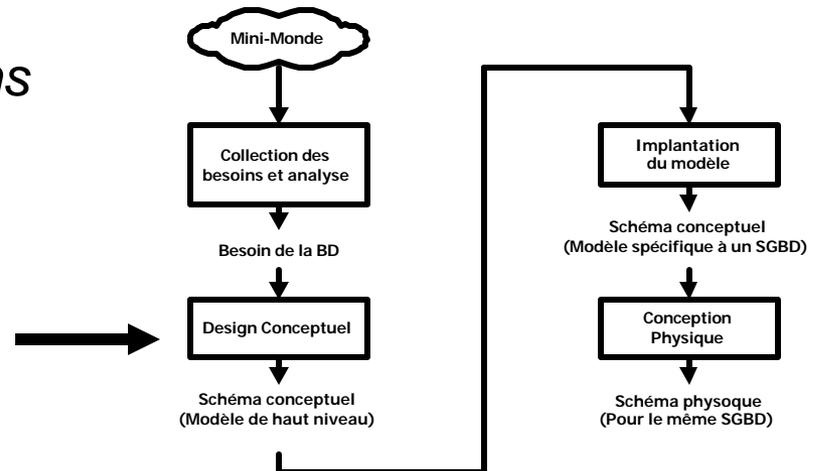
- *Collection des besoins des utilisateurs.*
  - Spécifiques
  - Détaillés
  - Complets
- *Analyse du problème:*
  - On se questionne sur quelles informations nous voulons stocker?



# Processus de modélisation d'une BD

## 2ième étape : Design Conceptuel

- *Création du **schéma conceptuel** basé sur le modèle **Entité-Relation**.*
- *Description concise des besoins des utilisateurs :*
  - des types de données,
  - des relations
  - des contraintes.
- *Facile à comprendre de l'utilisateur.*
- *Aucune implantation. Il est seulement une représentation graphique.*

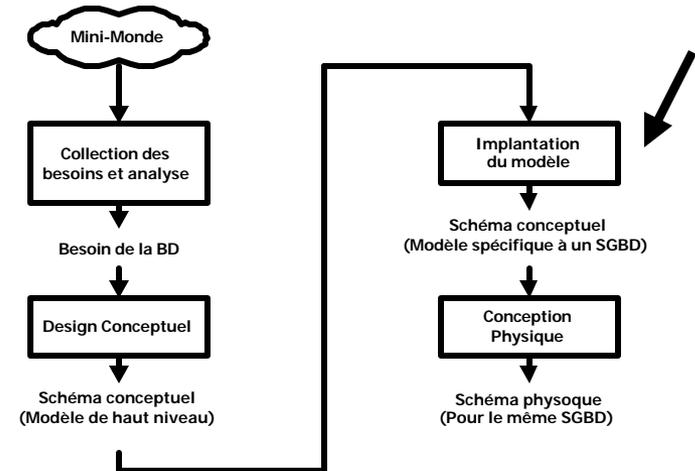


Remarque: Le fait que le concepteur n'a pas de souci d'implantation lui permet de développer et concevoir une **bonne BD conceptuelle**.

# Processus de modélisation d'une BD

## 3ième étape : Implantation du modèle

- *Déterminer la structure de la BD:*
  - relationnelle,
  - réseau
  - hiérarchique.



- *Le choix du SGBD dicte donc la structure interne (physique) de la BD.*
- *Transformation du schéma conceptuel du modèle de données haut niveau au **modèle d'implantation de données**.*

## 4ième étape : Conception physique

- *Spécification des structures de stockage interne et de l'organisation des fichiers de la BD.*

# Modèles de structuration des données

---

La modélisation des données permet de décrire correctement l'architecture d'une BD.

- les données;
- leurs relations;
- leur sémantique;
- les contraintes.

Deux grands modèles de structuration :

- Modèle logique orienté objet
- Modèle logique orienté enregistrement

# Modèle logique orienté objet

---

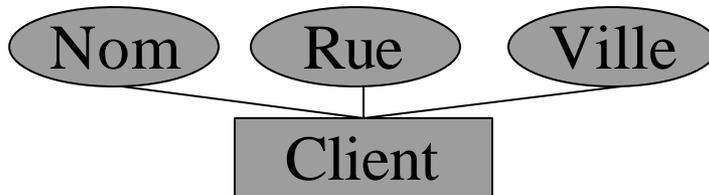
- *Niveau conceptuel et externe*
- *Grande flexibilité*
  - description des structures
  - explication des contraintes
- *Très grand nombre de modèles*
  - plus connu → modèle Entités-Relations (E-R)
    - ◆ *Entités et Attributs*
    - ◆ *Relations*
    - ◆ *Contraintes*

# Modèle logique orienté objet

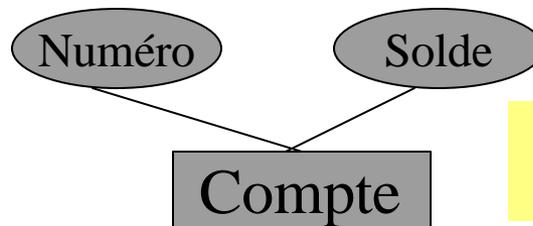
**ENTITÉ** : Objet discernable des autres objets qui est affecté d'attributs qui lui sont propres.

*Ex:*

- Client affecté de : nom, rue, ville



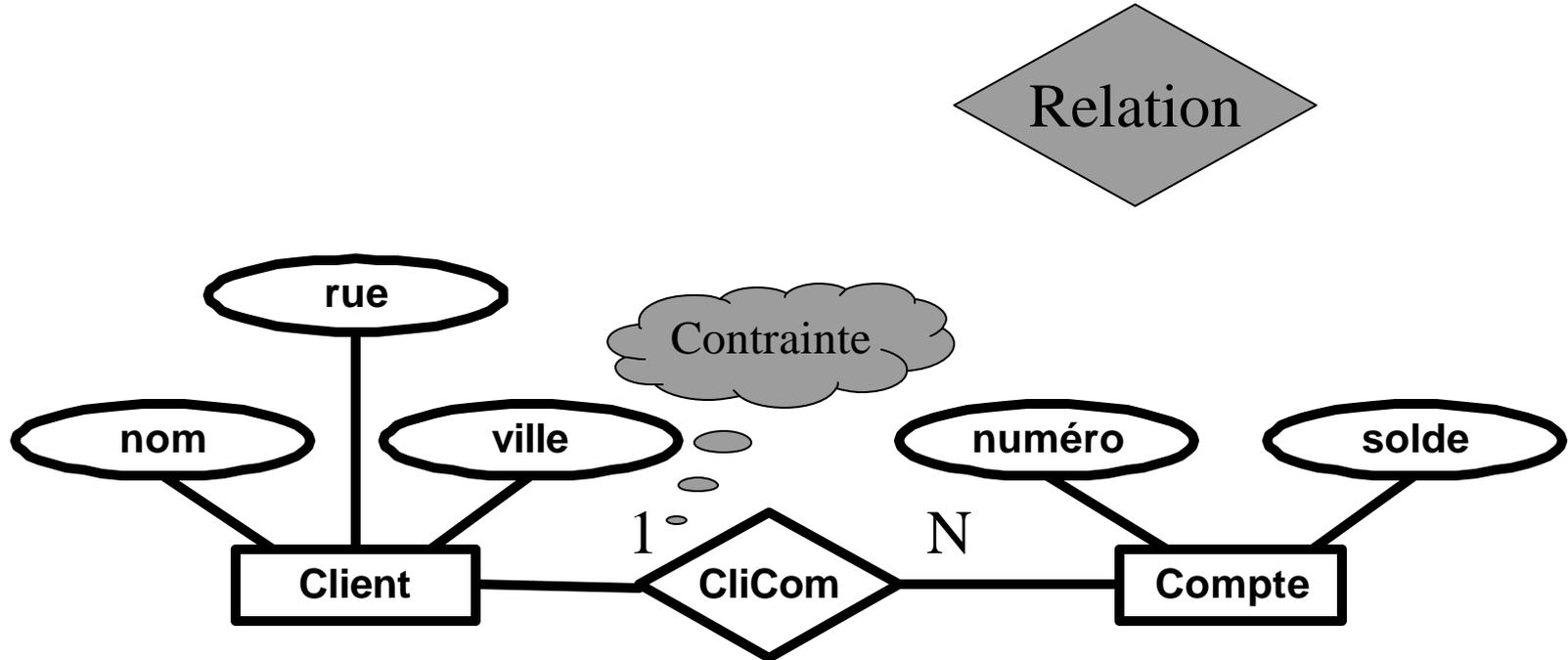
- Compte affecté de : numéro, solde



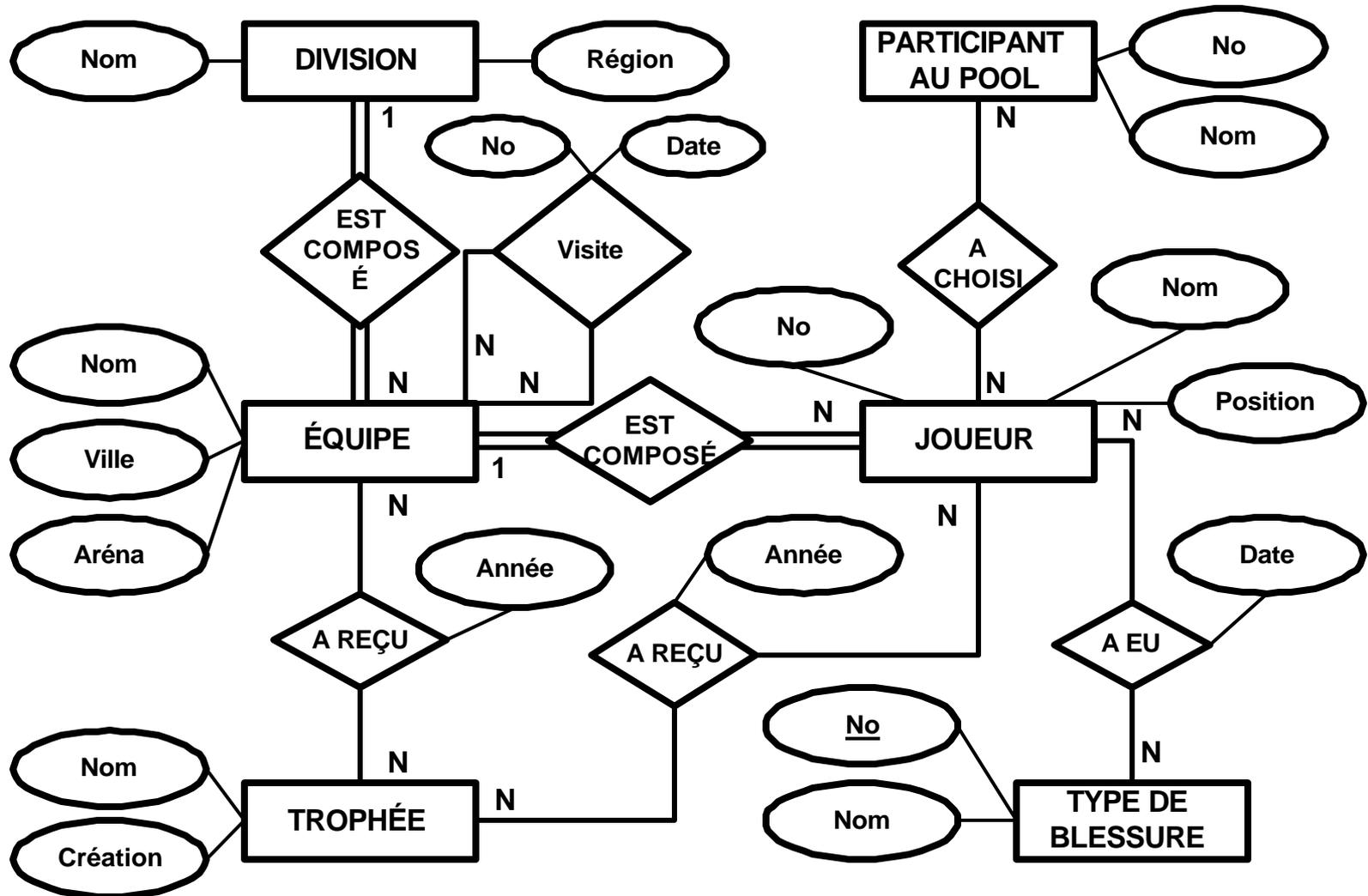
# Modèle logique orienté objet

**RELATION** : Associe plusieurs entités

*Ex : La relation CliCom associe un(e) client(e) à chacun des comptes qu'il (elle) possède.*



# Exemple de modèle E-R



# Modèles logiques orientés enregistrements

---

- *Niveau conceptuel et interne de la BD;*
- *Se rapproche de la structure physique des données;*
- *Trois modèles les plus connus :*
  - Modèle relationnel;
  - Modèle hiérarchique;
  - Modèle réseau.

# Modèles logiques orientés enregistrements

---

## Modèle relationnel (dominant sur le marché)

1. Données perçues comme des tableaux.
2. Existe un ensemble de relations entre les tableaux. (domaines communs)

**FOURNISSEURS**

F#	FNOM	ÉTAT	VILLE
F1	Smith	20	Montréal
F2	Jones	10	Laval
F3	Blake	30	Laval

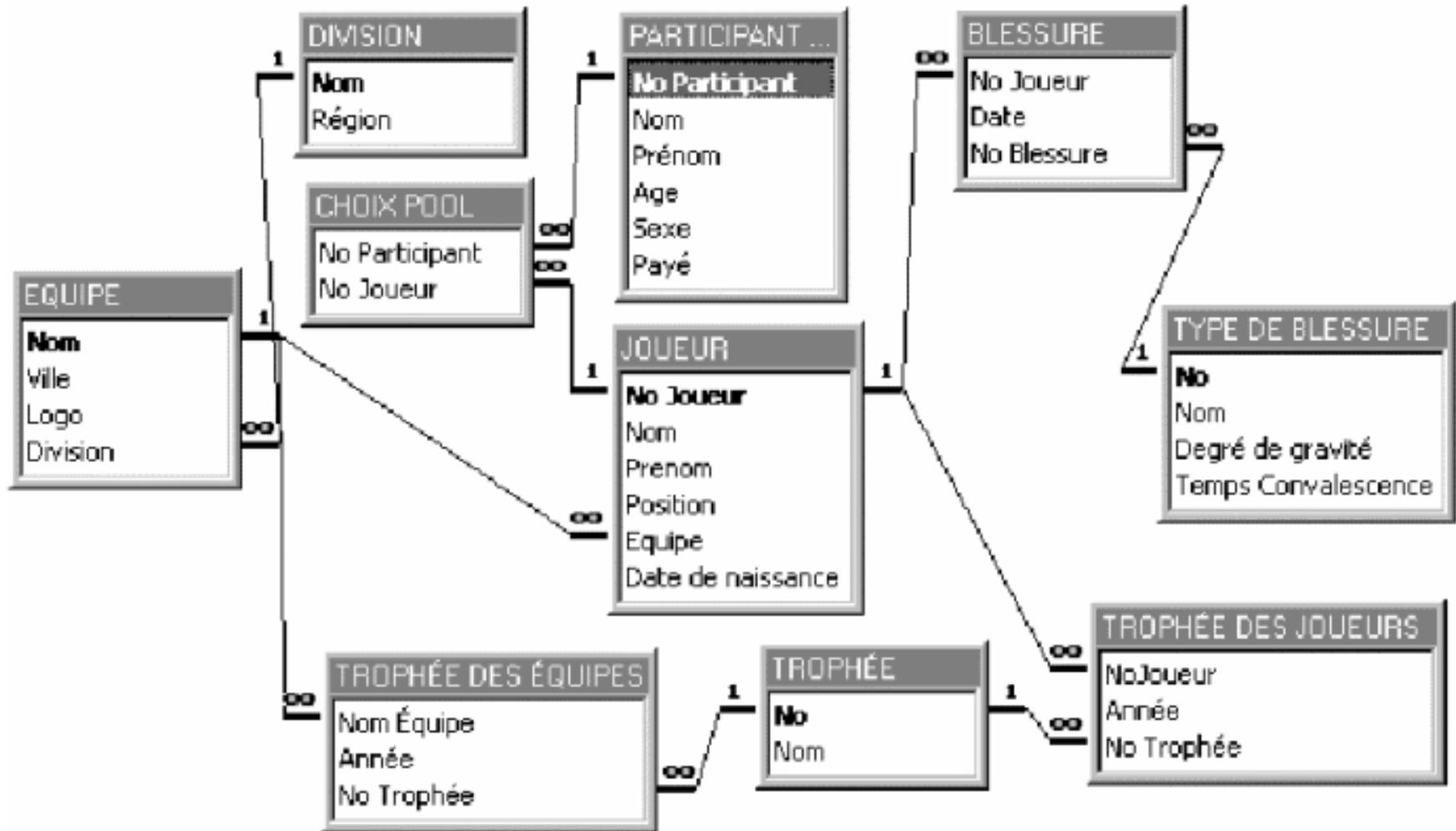
**PIÈCES**

P#	PNOM	COULEUR	POIDS	VILLE
P1	Noix	Rouge	12	Montréal
P2	Boulon	Vert	17	Laval
P3	Écrou	Bleu	17	Québec
P4	Écrou	Rouge	14	Montréal

**COMMANDE**

F#	P#	QUANTITÉ
F1	P1	300
F1	P2	200
F1	P3	400
F2	P1	300
F2	P2	400
F3	P2	200

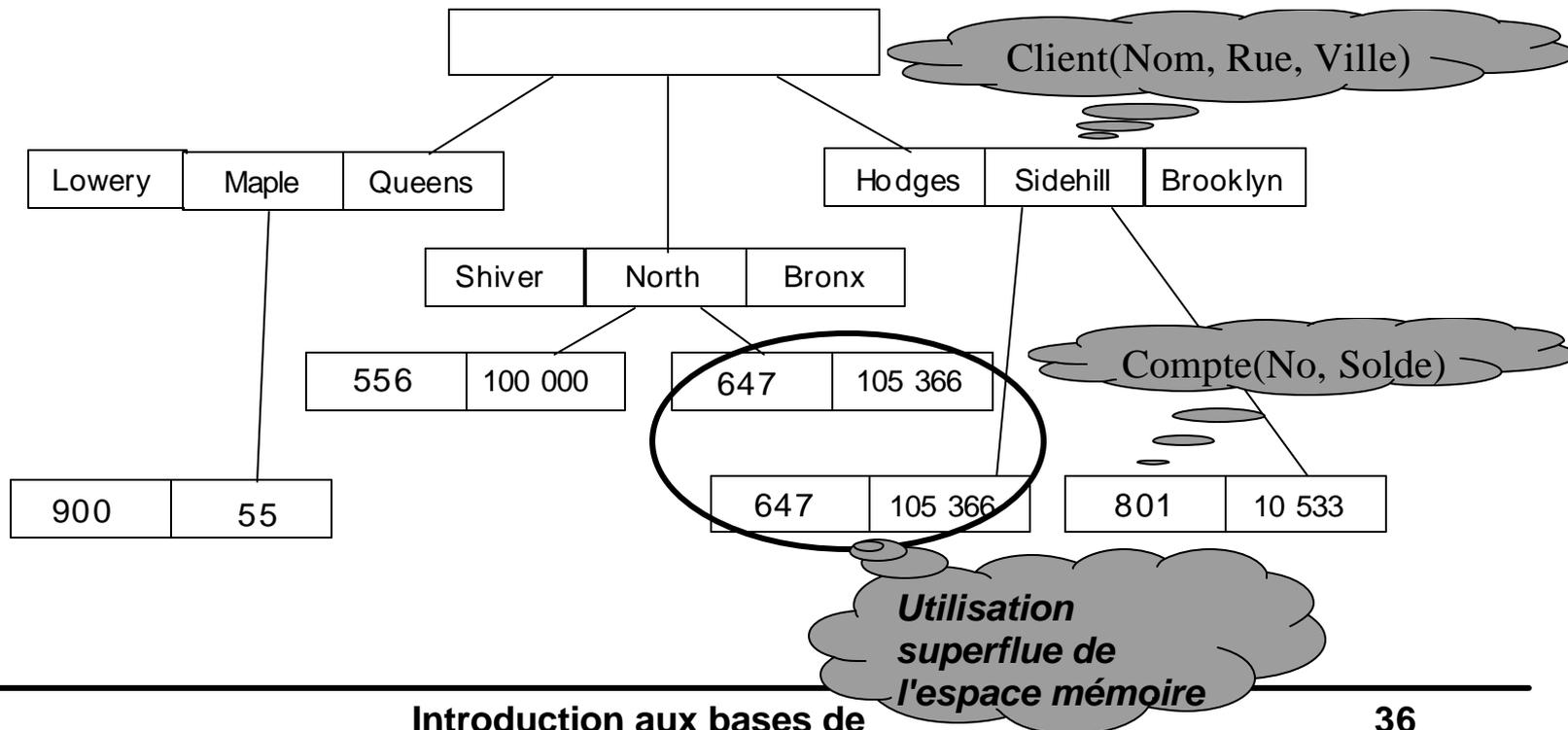
# Exemple de modèle Relationnel



# Modèles logiques orientés enregistrements

## Modèle hiérarchique

- la structure de données est représentée par des arbres.
- les informations sont décomposées en groupes répétitifs appelés enregistrements.



# Modèles logiques orientés enregistrements

---

## Modèle hiérarchique

Désavantages :

- *répétition des données*
- *modification d'une information sur toutes les occurrences lors d'une mise à jour;*
- *toutes les interrogations dépendent de la racine (traverser l'arbre en partant de la racine pour retrouver l'information).*



# Modèles logiques orientés enregistrements

---

## Modèle réseau

Avantages :

- *Évite les répétitions des données, seuls les pointeurs le sont;*
- *Diversifie l'accès à un groupe de données.*

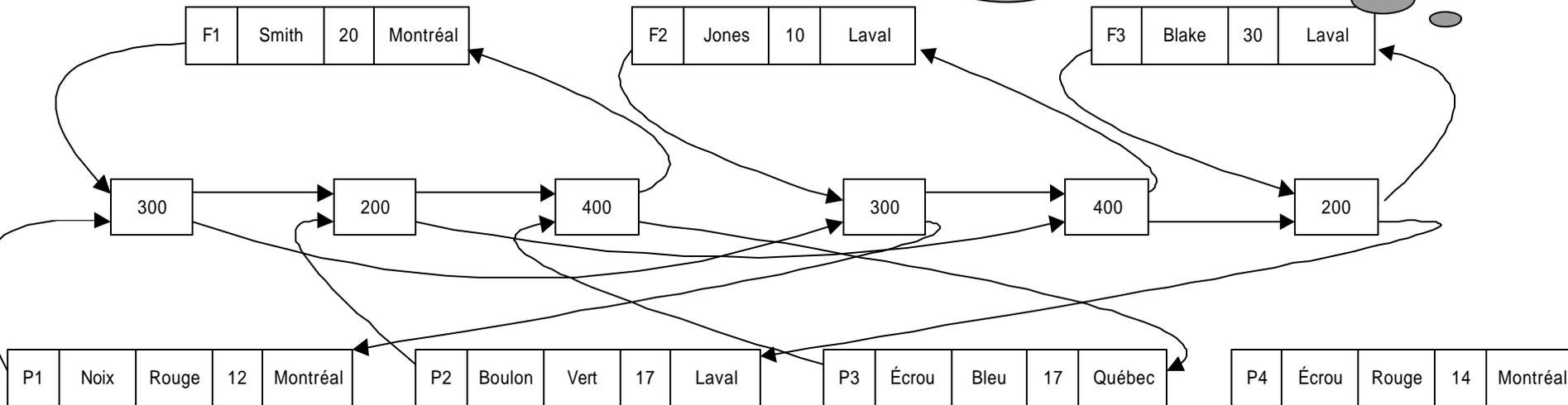
Désavantage:

- *Structure très complexe à définir.*
- *Peut causer des problèmes sérieux si une donnée a été oubliée lors de la définition du problème...*

# Exemple d'un modèle réseau

**Exemple** : La structure réseau des entités Fournisseurs et Pièces (p.19)

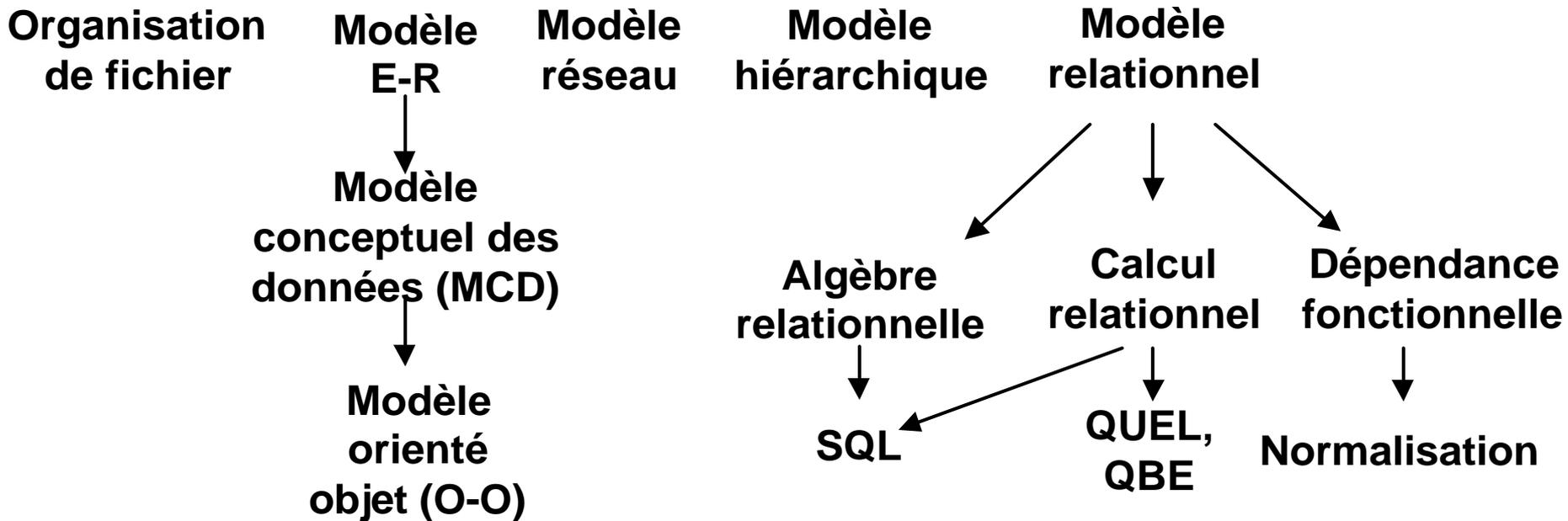
Oups... j'ai oublié de spécifier qu'il y avait 2 types d'écrous (1/4 et 1/2) !!!



# Modèles de structuration des données

---

Principaux points qui seront abordés dans la première partie du cours (Chapitres 1 à 6)



# Structure globale d'un SGBD

---

***Un SGBD se divise en :***

- Modules (tâches spécifiques)
- Structure de données

***Auxquels ont accès différents usagers :***

- Exploitants
- Administrateur
- Concepteur(s)
- Analyste de systèmes
- Programmeurs d'applications

# Structure globale d'un SGBD

---

Exploitants:

- *Accèdent à la base dans leurs fonctions quotidiennes:*
  - recherche,
  - mise-à-jour,
  - générer des rapports.

Administrateur:

- *Administre les ressources (BD, SGBD et les logiciels).*
- *Autorise les accès et gère l'utilisation.*
- *Responsable de la sécurité et des améliorations du temps de réponse du système.*

*(Dans les grandes organisations, il est assisté par du personnel.)*

# Structure globale d'un SGBD

---

Concepteur(s) de la base:

- *Le concepteur et l'administrateur peuvent être la même personne.*
- *Responsable d'identifier les données à être stockées.*
- *Choisis les structures pour représenter les données.*
- *Doit bien comprendre les requêtes des usagers.*
- *S'occupe du module de Langage de définition des données (DDL).*

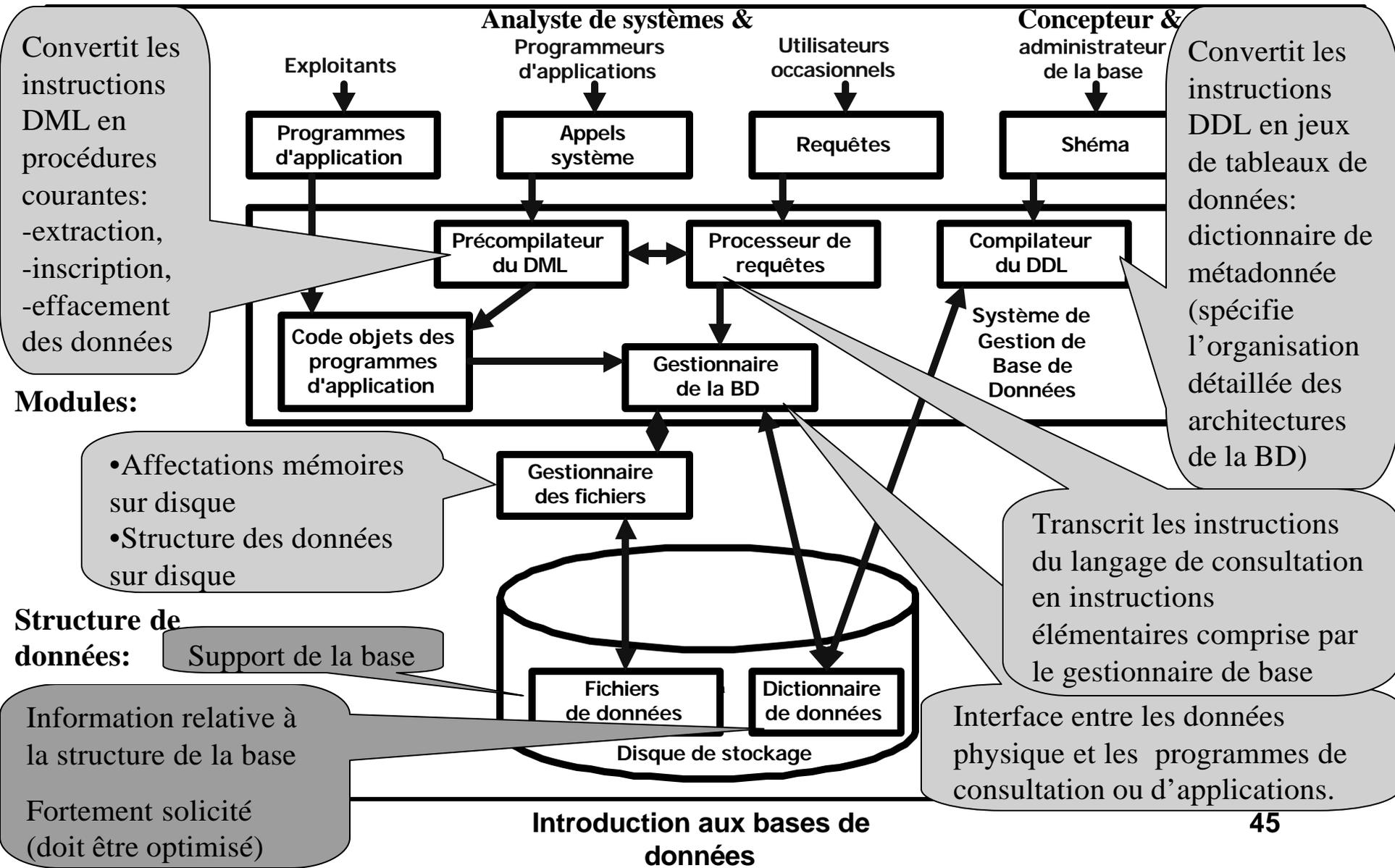
Analyste de systèmes:

- *Détermine les besoins des usagers et développe les spécifications.*
- *S'occupe du module de Langage de Manipulation des données (DML)*

Programmeurs d'applications:

- *Implante les spécifications dans un programme et effectue des tests, les documente et les entretient.*

# Structure globale d'un SGBD



# Vue d 'ensemble... SGBD

---

Le SGBD est un système logiciel qui facilite les processus:

- de définition,
- de construction
- et de manipulation des BDs.

## **Exemple:** Description fonctionnelle

Considérons la base de données d'une université pour maintenir l'information concernant les étudiants et leurs cours. Chaque étudiant suit un certain nombre de cours. Un cours peut posséder certains pré-requis. Pour chaque cours effectués, un étudiant doit avoir un résultat.

# Vue d 'ensemble... Mise en situation d'un exemple

---

## Définition des données:

- Permet de spécifier les types de données de façon détaillée.
- La définition des types de données est stockée dans le catalogue (dictionnaire) qui comprend:
  - La structure de chaque fichier
  - Le type et le format de chaque élément de données
  - Différentes contraintes sur les données

Cette information est appelée: **Méta-data**

Cette BD pourrait être organisée en 4 entités comme suit:

ÉTUDIANT:	Information sur chaque étudiant.
COURS:	Information sur chaque cours.
RÉSULTAT:	Résultat pour les cours des étudiants.
PRÉ-REQUIS:	Les pré-requis de chaque cours.

## Vue d 'ensemble ... Définition des données

---

Pour définir cette BD, il faut spécifier les données à stocker dans chaque enregistrement:

### **Enregistrement ÉTUDIANT:**

Nom  
Prénom  
Code Permanent  
Adresse

### **Enregistrement COURS:**

Nom du cours  
Numéro du cours  
Crédit  
Département

### **Enregistrement RÉSULTAT:**

Code Permanent  
Nom du cours  
Note

### **Enregistrement PRÉ-REQUIS:**

Numéro du cours  
Numéro du cours pré-requis

# Vue d 'ensemble ... Définition des données

---

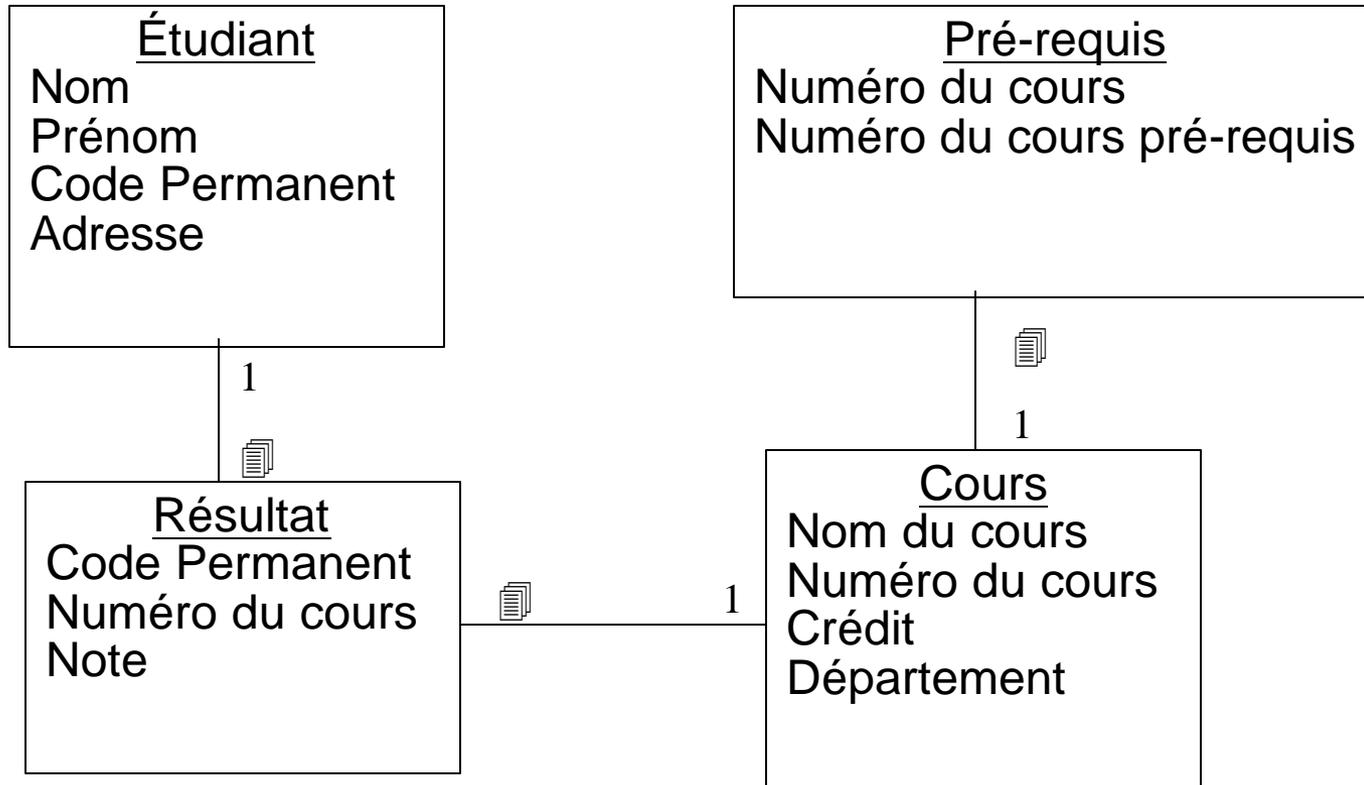
Pour définir cette BD, il faut aussi spécifier le type des données.

## **Enregistrement COURS:**

Nom du cours	Caractère(20)
Numéro du cours	Caractère(6)
Crédit	Entier
Département	Caractère(3)

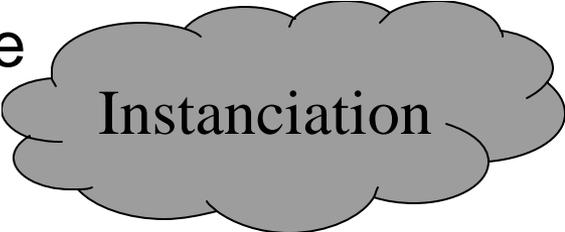
# Vue d'ensemble ... Modélisation des données

Modèle relationnel:



## Construction de la BD:

Stockage des données dans une mémoire secondaire contrôlée par le SGBD.



Instanciation

On emmagasine les données représentant chaque Étudiant, Cours, Résultat et Pré-requis comme enregistrement dans le fichier approprié.

**NOTE:** Les enregistrements dans les différents fichiers peuvent être reliés entre eux.

Ex: Les cours et les pré-requis

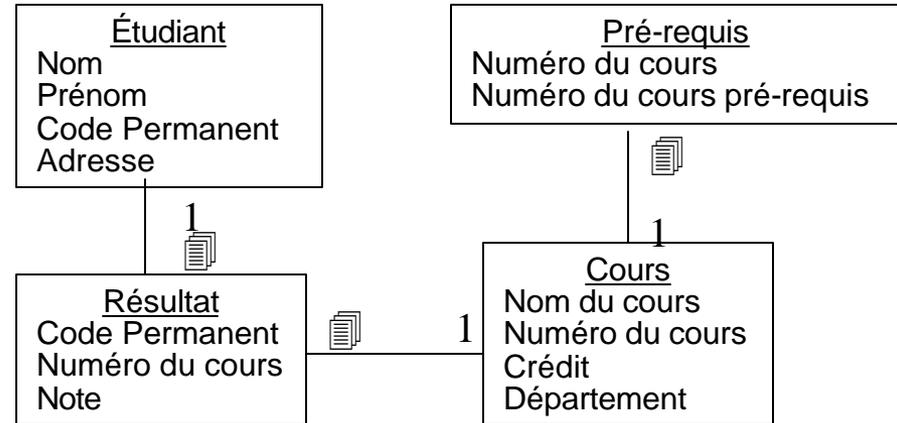
Les étudiants et les cours

Les étudiants et les résultats

# Vue d'ensemble... Manipulation des données

La manipulation de la BD implique:

- interrogations,
- mises à jour,
- et génération de rapports.



Exemple:

Retrouver la liste de tous les cours et résultats de l'étudiant Pierre Tremblay.

1. Aller dans le fichier ÉTUDIANTS.
2. Trouver le code permanent de Pierre Tremblay.
3. Aller dans le fichier RÉSULTAT.
4. Lister tous les cours et leur résultat qui ont le code permanent de Pierre Tremblay.

Comment puis-je la trouver?

Lister les cours d'un département.

1. Aller dans le fichier COURS.
2. Lister tous les cours du département correspondant.

# Vue d'ensemble... Manipulation des données

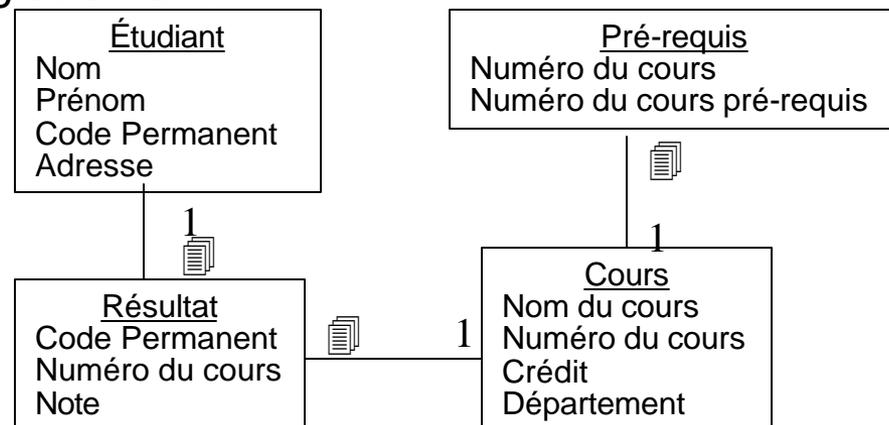
Exemple (suite):

Quels sont les cours pré-requis du cours de GPA775.

1. Aller dans le fichier PRÉ-REQUIS
2. Lister tous les numéros de pré-requis correspondant au cours GPA775.
3. S'il y a résultat, aller dans le fichier COURS
4. Chercher les noms de cours correspondant au numéro de pré-requis trouvé précédemment

Créer un nouveau pré-requis pour le cours de GPA775.

1. Aller dans le fichier PRÉ-REQUIS.
2. Ajouter l'enregistrement.



# Vue d'ensemble... Manipulation des données

Exemple (suite):

Changer l'adresse d'un étudiant.

1. Aller dans le fichier ETUDIANT
2. Chercher l'étudiant correspondant
3. Modifier le champ de l'adresse.

Effacer un cours

1. Aller dans le fichier COURS
2. Effacer l'enregistrement du cours correspondant
3. Aller dans le fichier PRÉ-REQUIS
4. Effacer tous les enregistrement correspondant au cours correspondant.

Attention!!! Le cours que vous venez d'effacer est peut-être le pré-requis d'un autre cours!!!! Qui s'assure de l'intégrité de la base de données???

