

# Conception d'une BD

## [Partie I]

## SGBD

- Déf**
- **BD**: collection de données modélisant les objets d'une partie du monde réel et servant de support à une application.
  - **SGBD**: Système de gestion de bd
    - système logiciel
    - mono ou multi-ordi
    - engénéral, peut gérer plusieurs BD
    - peut accéder aux BD d'autres SGBD
  - **Banque de données**: BD + SGBD + services

## Objectif d'une SGBD

- indépendance des programmes aux données
- simplicité de manipulation
- efficacité des accès aux données
- partage et sécurité des données
- redondance contrôlée des données
- conception facilité des applications
- facilité de l'administration SGBD

## Langage des SGBD

- **Langage de définition de données (LDD)**
  - définition de schémas conceptuel, externe et interne
  - définition stockée dans le répertoire système
- **Langage de manipulation de données (LMD)**
  - langage de requête déclaratif, peut être autonome (SQL) ou intégré (C, java)
- **Utilisateurs d'un SGBD**
  - utilisateur final (ad-hoc)
  - programmeur d'appli
  - administration de la BD

## Architecture d'un SGBD

- **Storage manager**: file manager / buffer manager
- **transaction**: séquence considérée atomique d'actions (r/w) sur la BD  
chaque transaction doit laisser la BD dans un "consistent state" et la BD dans un état cohérent avant la transaction.
- **propriété ACID des transactions**:
  - atomicité
  - cohérence
  - isolation
  - durabilité

# [Partie II]

## Modèle relationnel

### → schéma relationnel

BD relationnel : ensemble de relations

schéma d'une BD rela : ensemble de schémas de relations

schéma rela : spécifie le nom de la relation ainsi que le nom et le type de chaque attribut.

### → Définition

Relation  $\leftrightarrow$  table

Attribut  $\leftrightarrow$  colonne

Tuple  $\leftrightarrow$  ligne

Domaine  $\leftrightarrow$  valeur de la colonne

Schéma de la rela (intension)  $\leftrightarrow$  définit<sup>o</sup> de la table

Extension  $\leftrightarrow$  exemples de la table

## Contrainte du modèle relationnel

- Attribut : simple et monovalués  
structure régulière
- valeurs nulles : Un attribut peut ne pas être renseigné  
Lo l'attribut n'est pas défini pour le tuple en question  
Lo valeur de l'attribut n'est pas encore connue
- Règle d'identification :  $\forall$  relations possède un identifiant : Clé  
Soit un ensemble C:
  - 1) Deux tuples distincts ne peuvent pas avoir la même valeur pour les attributs de la clé.
  - 2) Si 1) n'est pas vraie pour un sous-ensemble d'attribut  $C'$  de C  
Si 2) n'est pas vérifié, l'ensemble C est une super clé (non minimale)
- clé primaire : clé de la relation    • clé candidate : clé non primaire  
L'identifiant n'admet pas de valeurs nulles.

## Caractéristiques

Basés sur la théorie des ensembles

- pas d'ordre entre les attributs  $\Rightarrow$  SGBD : ordre dans les attributs
- pas d'ordre entre les tuples  $\Rightarrow$  SGBD : résultats des requêtes peuvent être triés
- pas de tuples dupliqués  $\Rightarrow$  SGBD : autorise les doublons

Valeurs d'attributs sont atomiques

Degré / arité : nb d'attributs

Cardinalité : nb de tuples

## Algèbre relationnel

- Langages & BDD
  - ↳ langage de manipulation ( $\omega$ )
  - ↳ langage d'intégration ( $\pi + \text{calcul}$ )
- ↳ langage algébrique (algèbre relationnel, opération relationnel)
- ↳ langage prédictif (logique de prédicat, langage déclaratif)

## Opérations

Sélection :  $\sigma$

renommage :  $\rho$

Jointure :  $\bowtie$

Projection :  $\pi$

Union :  $\cup$

Intersection :  $\cap$

Produit Scalaire :  $\times$

différence :  $-$

division :  $/$

## Partie IV

## Modélisation E/A

entité : chose concrète ou abstraite, objet / existence autonome

Type d'entité : classe de toutes les entités de la réalité perçue qui sont de la même nature et qui jouent le même rôle.

- ↳ nom
- ↳ définition précise
- ↳ liste d'attributs

Une entité possède des attributs

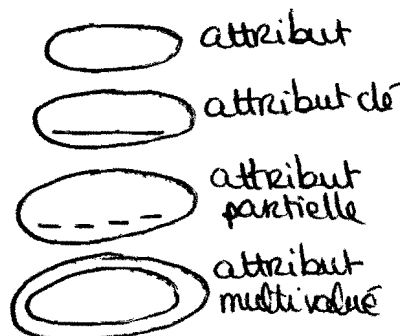
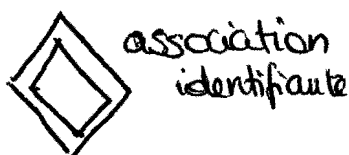
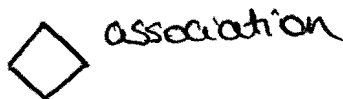
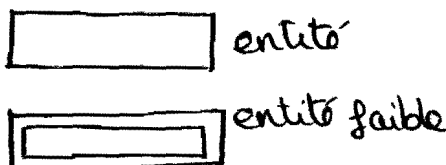
- ↳ caractéristiques ou qualité d'une entité ou d'une association
- ↳ valeur est un symbole utilisé pour représenter un fait élémentaire

Association : correspondance entre 2 ou plusieurs occurrences d'entités à propos de laquelle on veut conserver des informations.

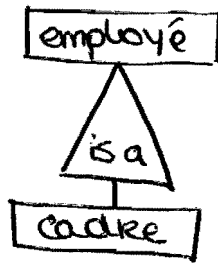
Cardinalité : min 0 : participation partielle (peut exister)

min 1 : participation totale

Entité faible : ne peut être identifiée que par rapport à une autre entité dite dominante.



## Spécialisation



## Aggrégation

- Quand on veut faire participer une asso dans une autre
- Utilisation d'une association comme entité.

## Contrainte d'intégration (CI)

Propriété non représentée par les concepts de base du modèle E/A que doivent satisfaire les données appartenant à la BD.

- > CI statique : identification, cardinalité, attributs obligatoires et facultatifs
- > CI dynamique : conséquence possible des changements d'état de la BD

Exemple modèle E/A au modèle relationnel :

nom\_entité (attr1, attr2, ...)

## Validation

- > règle de complétude du schéma
- > validation : la cohérence du schéma
- > mise en forme canonique
  - unicité des noms, absence d'homonymes
  - absence de synonymes
  - clé minimale
  - mise en évidence des attributs dérivables
  - élimination des structures redondantes
  - désagrégation d'une entité
  - Type asso ou type entité
  - validation des attributs d'un type d'asso.

## Partie V.

Danger de la redondance

### • Définition Dépendance fonctionnelle

On dit qu'il existe une dépendance fonctionnelle de  $X$  vers  $Y$  noté  $X \rightarrow Y$  ( $X$  détermine  $Y$  ou  $Y$  dépend fonctionnellement de  $X$ ) ssi il existe une fonction qui à partir de toute valeur de  $X$  détermine une valeur unique de  $Y$

Décompos<sup>o</sup> d'une DF : On ne peut pas décomposer la partie gauche de la DF

Sémantique d'une DF : L'ensemble des attributs  $K$  est une super-clé d'une relation  $R$  ssi  $K$  détermine tous les attributs de  $R$ .

$K$  est une clé d'une relation ssi :  
-  $K$  est une super clé de  $R$   
- il n'existe pas un sous ensemble propre de  $K$  formant une super clé de  $R$ .

Une DF représente une contrainte d'intégrité.

Evaluation d'une DF  $X \rightarrow Y$  peut être imposée au schéma

1.  $X$  et  $Y$  sont inclus dans le même schéma de relation  $R$  :  
la vérification de contrainte est simple
2.  $X$  et  $Y$  sont inclus dans des schémas de relation  $\neq$   
la vérification est complexe  $\rightarrow$  jointure

### Axiome d'Armstrong (base)

Reflexivité : si  $X \supseteq Y$  alors  $X \rightarrow Y$

Augmentation : si  $X \rightarrow Y$  alors  $XZ \rightarrow YZ$

Transitivité : si  $X \rightarrow Y$  et  $Y \rightarrow Z$  alors  $X \rightarrow Z$

Union : si  $X \rightarrow Y$  et  $X \rightarrow Z$  alors  $X \rightarrow YZ$

Décomposition :  $X \rightarrow YZ$  alors  $X \rightarrow Y$

Pseudo-transitivité :  $X \rightarrow Y$  et  $WY \rightarrow Z$  alors  $XW \rightarrow Z$

## Forme normale

1ère forme normale

elle est en 1<sup>re</sup> forme si tout attribut contient une valeur atomique (unique)

2ème FN

elle est en 2<sup>e</sup> forme, tout attribut non clé ne dépend pas d'une partie de la clé

3ème FN

aussi 2<sup>e</sup> forme, tout attribut n'appartenant pas à une clé ne dépend pas d'un autre attribut de la clé

Forme Normale de Boyce Codd (BCNF)

l'ensemble des dépendances fonctionnelles non triviales  $X \rightarrow Y$  implique que  $X$  est une super clé de  $R$ .