



Faculté des Sciences Semlalia

## Modélisation Relationnelle

### Chapitre III

Enseigné par:  
Pr. J. ZAHIR  
j.zahir@uca.ac.ma

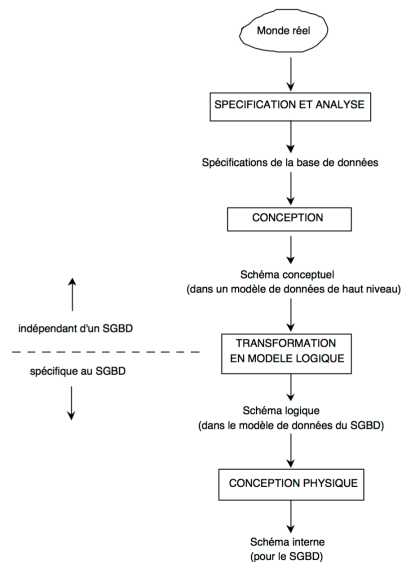
11 octobre 2018

## Objectifs d'apprentissage de la séance

- Connaître les modèles logiques classiques
- Assimiler les concepts du modèle relationnel
- Assimiler les règles de transformation du MCD en Modèle relationnel

## Plan

- 1 Modélisation logique
- 2 Modèles navigationnels
- 3 Modèle relationnel
- 4 Transformation du MCD en un Modèle relationnel



## Introduction

La modélisation logique des données :

- Est une représentation des données, appelée Modèle Logique de Données (MLD)
- Exprimée dans un formalisme logique adapté au SGBD envisagé à l'optimisation générale
- Plusieurs modèles existent pour représenter le MLD, parmi lesquels :
  - Modèles navigationnels
    - Modèle hiérarchique
    - Modèle en réseau (ou CODASYL )
  - Modèle relationnels

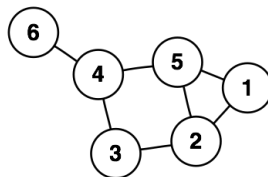
## Plan

- 1 Modélisation logique
- 2 Modèles navigationnels
  - Modèle hiérarchique
  - Modèle en réseau (CODASYL)
- 3 Modèle relationnel
- 4 Transformation du MCD en un Modèle relationnel

Anciens modèles logique de données très utilisés entre 1965-1970

### Caractéristique principale

Les modèles navigationnels utilisent des **pointeurs** et des **chemins** pour naviguer dans les **enregistrements** des données modélisés sous forme de noeuds.



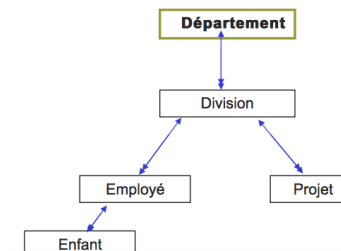
Chemin vers Noeud 1 :

- Noeud6.Noeud4.Noeud5.Noeud1, ou encore
- Noeud6/Noeud4/Noeud5/Noeud1

On distingue deux modèles navigationnels : Modèles **hiérarchiques** et Modèles en **réseau**

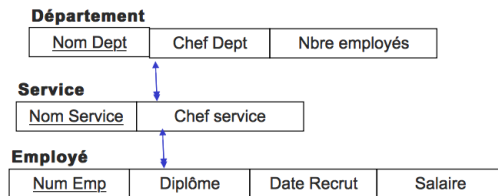
## Modèle hiérarchique : Un aperçu

- BD construite selon un modèle en arborescence, avec une racine et plusieurs niveaux de sous-arbres
- Chaque élément comporte juste un lien vers le niveau inférieur
- Les accès aux données commencent par la racine et descendent l'arborescence jusqu'aux détails recherchés



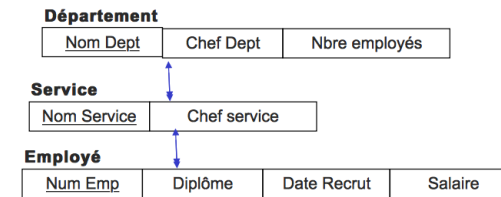
## Modèle hiérarchique : Limitations (1/2)

- **Redondance** : le modèle est incapable de représenter les liens N :M ( un et un seul parent )
- **Anomalie de destruction** : la destruction d'un nœud cause l'élimination de ses successeurs ( suppression d'un service  $\Rightarrow$  suppression de ses employés)



## Modèle hiérarchique : Limitations (2/2)

- **Anomalie d'insertion** : impossible d'insérer des données sans parent (un employé non affecté à un service ne peut pas être inséré)
- **Anomalie de modification** : une modification doit éventuellement être effectuée à plusieurs endroits (coût élevé)
- **Absence d'un langage non procédurale**



## Modèle réseau : Un aperçu

- Proposé initialement par le groupe DBTG du comité CODASYL
- Objectif  $\Rightarrow$  répondre aux problèmes que pose le modèle hiérarchique et aboutir à une représentation de données sans pertes
- Nombreux liens entre les différents éléments de données ( suppression de la contrainte de parent unique)
- Accès aux données réalisés par des cheminements divers

## Modèle réseau : Propriétés

### Avantages

- Pas d'anomalies de stockage, insertion, suppression, modification
- Permet de modéliser les liens N :M

### Inconvénients

- Pas de séparation entre schéma logique et schéma interne (les chemins d'accès doivent être pris en compte)
- Pas de langage non procédural pour les non spécialistes

## Modèles Navigationnels : Synthèse

Le modèle relationnel vient pour proposer des solutions aux limitations des modèles navigationnels. Toutefois :

- Ces modèles restent encore utilisés dans l'industrie :
  - IMS de IBM
  - IDS II de BULL

## Plan

- 1 Modélisation logique
- 2 Modèles navigationnels
- 3 **Modèle relationnel**
  - Domaine
  - Table ou Relation
  - Attributs
  - Clés primaire et étrangère
  - Schéma de relation et Schéma de base de données
  - Méta-base
- 4 Transformation du MCD en un Modèle relationnel

## Introduction (1/2)

- Le modèle relationnel a été défini par E.F. Codd en 1970 à IBM
- Les concepts du modèle relationnel découlent de la théorie des ensembles
- Permet un haut degré d'indépendance entre les programmes d'application et les représentations internes des données (stockage, type d'indexage, ...)
- Fournit une base solide pour traiter les problèmes de cohérence et de redondance des données

## Introduction (2/2)

Notions associées au modèle :

- Domaine
- Table relationnelle
- Attribut
- Tuple (ou n-uplet)
- Clés primaire et étrangère

Un domaine est un ensemble de valeurs atomiques ayant une signification pour l'utilisateur. Il peut être défini, en :

- **Intension** : En définissant une propriété caractéristique des valeurs du domaine, ou
- **Extension** : En donnant la liste de toutes les valeurs le composant

#### Exemple de domaines

- Entier
- Réel
- Chaines de caractères
- {4000...10000}
- {Rouge, Vert, Bleu}

## Produits cartésiens de domaines

Le produit cartésien  $d_1 \times d_2 \times \dots \times d_n$  est l'ensemble des tuples (n-uplets) :  $\langle V_1, V_2, \dots, V_n \rangle$  tel que  $V_i \in d_i$

#### Exemple

Soient les domaines :  $D_1 = \text{Rouge, Vert, Bleu}$  et  $D_2 = \text{Vrai, faux}$

Rouge	Vrai
Rouge	Faux
Vert	Vrai
Vert	faux
Bleu	Vrai
Bleu	Faux

#### Table ou Relation

- Sous-ensemble du produit cartésien d'une liste de domaines, elle est généralement caractérisée par un nom.
- La table (ou relation) est définie comme un tableau de données
- Colonnes de la table  $\Rightarrow$  **attributs**
- Lignes de la table  $\Rightarrow$  **tuples** ou **enregistrements** : L'ordre des lignes n'est pas significatifs
- Degré d'une relation = **Nombre de ses attributs** : Les attributs sont ordonnés.

- La notion d'attribut en relationnel correspond à la notion de propriété pour une entité
- Un attribut est défini par un nom et un domaine
- Le domaine d'un attribut décrit les valeurs autorisées pour cet attribut
- Les valeurs d'un attribut sont atomiques : non décomposables

- Un ou plusieurs attributs permettent d'identifier de façon unique chaque tuple de la table : **clé primaire**
- La clé primaire est dite simple si elle est constituée d'un seul attribut et composée dans le cas contraire
- Quand la clé primaire  $C_p$  d'une table A est dupliquée dans une autre table B,  $C_p$  devient une **clé étrangère** dans la table B

Un schéma de relation contient :

- Le nom de la relation
- Les attributs de la relation ( avec éventuellement leurs domaines)
- La clé primaire soulignée
- La clé étrangère précédée par un #
- Exemple :  $R [\underline{att_1}, att_2, \#att_3]$

Un schéma de base de données est :

- Un ensemble fini non vide S de noms de relations
- $S = [R_1 [\underline{att_1}, att_2, att_3], R_2 [\underline{att_{01}}, att_{02}, \#att_1]]$

## Représentation graphique d'une table

### EMPLOYES

Matricule	Nom	DateNaiss	Salaire	#NomService

Nom de la table: employés

Schéma: 1 ère ligne

Tuple: chacune des autres lignes

Attribut: une colonne

Clé de la relation: Matricule

Clé étrangère: #NomService

- Le dictionnaire de données contient en général la description des relations, attributs, leurs noms et leurs domaines,
- Si le dictionnaire de données est organisé sous forme d'une base de données, celle-ci est appelée **Méta-base**,
- Une version simplifiée du dictionnaire des données peut être utile lors de la conception.

Attribut	Signification	Domaine
REFPROD	Référence du produit	Chaîne(12)
DESIGN	Désignation du produit	Chaîne(30)
PRIXHT	Prix unitaire HT	réel
NUMFACT	Numéro de la facture.	entier
DATEFACT	Date de la facture.	Date/heure
QTE	Quantité facturée	entier
...	...	...

- En réalité, le dictionnaire est un ensemble plus complexe de tables et de vues.
- Contient toutes les informations concernant la structure de stockage et tous les objets de la base.
- Automatiquement mis à jour par Oracle lorsque la base de données est modifiée.

## Plan

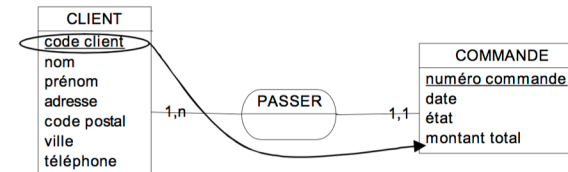
- 1 Modélisation logique
- 2 Modèles navigationnels
- 3 Modèle relationnel
- 4 Transformation du MCD en un Modèle relationnel
  - Attribut, clé primaire et table
  - Association binaire
  - Association unaire

- Une **propriété** (MCD)  $\Rightarrow$  **Attribut** (MLD)
- Un **identifiant** (MCD)  $\Rightarrow$  **Clé primaire** (MLD)
- Une **Concaténation d'identifiants** (MCD)  $\Rightarrow$  **Clé composée** (MLD)
- Une **entité** (MCD)  $\Rightarrow$  **Table** (MLD)
  - Ses propriétés deviennent les attributs de la table.
  - L'identifiant devient clé primaire de la table

## Association (0/1, n) – (0/1, 1)

### Association hiérarchique :

La clé primaire de la table issue de l'entité côté cardinalités (0/1,n) est dupliquée dans la table issue de l'entité côté cardinalités (0/1,1) où elle devient clé étrangère.

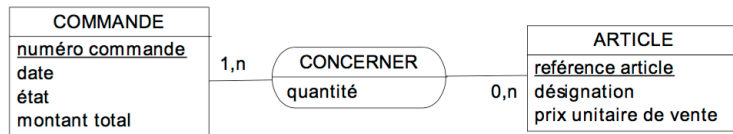


COMMANDE(numéro\_commande, date, état, montant\_total, # code\_client)

## Association (0ou1, n) – (0ou1, n)

### Association non-hiérarchique :

Une table ayant comme clé une clé composée des identifiants des 2 entités est créée. Les éventuelles propriétés de l'association deviennent les attributs de la table.



CONCERNER(#numéro commande, #référence article, quantité)

## Association (0, 1) – (1, 1) et (0, 1) – (0, 1)

### Association (0, 1) – (1, 1)

La clé primaire de la table issue de l'entité côté cardinalités (0,1) est dupliquée dans la table issue de l'entité côté cardinalités (1,1) où elle devient clé étrangère

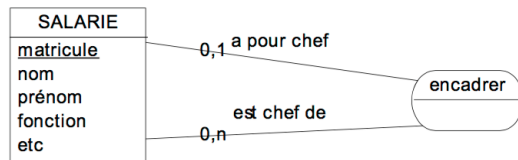
### Association (0, 1) – (0, 1)

La clé primaire de la table issue de l'une des entités est dupliquée dans la table issue de l'autre entité où elle devient clé externe étrangère

## Association unaire (0, n) – (0, 1)

### Association unaire hiérarchique

- En général, la clé primaire de la table issue de l'entité est dupliquée dans cette table où elle devient une clé étrangère.
- Il est utile de changer le nom de la clé étrangère en fonction du sens de l'association

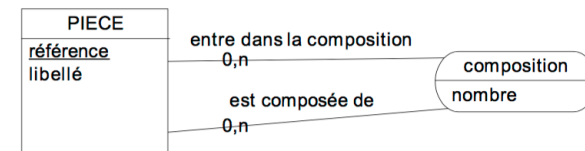


SALARIE(matricule, nom, prénom, fonction, ... , #matriculechef)

## Association unaire (0ou1, n) – (0ou1, n)

### Association unaire non hiérarchique

- Une table ayant comme clé une clé composée de 2 fois l'identifiant de l'entité est créée
- Les éventuelles propriétés de l'association deviennent des attributs de la table.

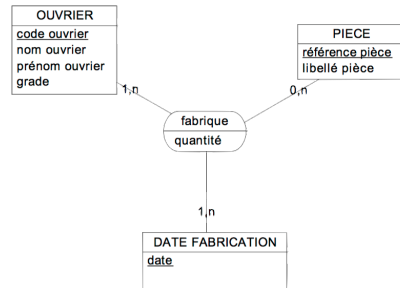


- PIECE(référence, libellé)
- COMPOSITION(#référenceComposé, #référenceComposant, nombre)



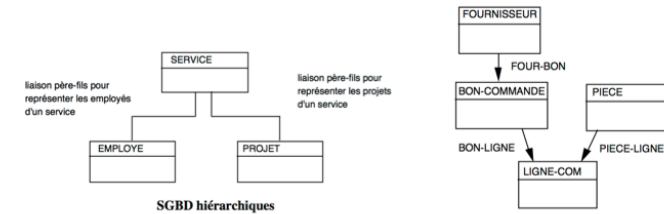
## Entité : Cas particulier

Les entités n'ayant que leur identifiant comme attribut ne deviennent pas des relations, mais des attributs dans les autres relations liées.



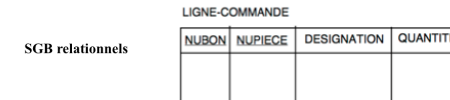
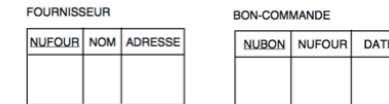
~~DATE FABRICATION(date)~~ fait partie de la clé primaire mais n'est pas clé étrangère  
FABRIQUE(#code\_ouvrier, #référence\_piece, date, quantité).

## Synthèse : Hiérarchique vs Réseau vs Relationnel



SGBD hiérarchiques

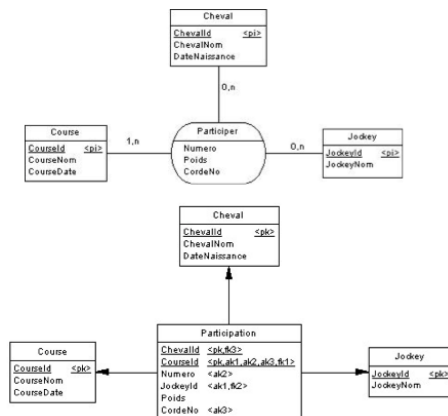
SGBD réseaux



SGB relationnels

Source: <http://eric.univ-lyon2.fr/~jdarmont/docs/old/1-introduction.pdf>

## Modèle relationnel : Représentation graphique



Source: Developpez.net