

# Conception de systèmes d'information

## Cours 3 – Modélisation logique de données

František Kardoš

fkardos@labri.fr

# Modélisation logique de données

Transformation du MCD en un MLD exprimé dans un formalisme logique adapté au SGBD (SGF)

- ▶ Modèle navigationnel
- ▶ Modèle relationnel

# Modèle relationnel

Notions de base :

- ▶ domaine
- ▶ table
- ▶ attribut
- ▶ clé
- ▶ tuple

# Modèle relationnel : les domaines

Un domaine est un ensemble de valeurs ayant une signification pour l'utilisateur

- ▶ le domaine des noms, le domaine des âges
- ▶ le domaine des entiers, le domaine des réels

# Modèle relationnel : les tables

Une table relationnelle (une relation) est un sous-ensemble du produit cartésien d'une liste de domaines (non nécessairement distincts).

- ▶ PERSONNE (N° INSEE, nom, prénom, adresse)
- ▶ VOITURE (Numéro, marque, type)

# Modèle relationnel : les tables

La table (ou la relation) est définie comme un tableau de données :

- ▶ Les colonnes sont les attributs
- ▶ Les lignes sont des tuples

Le(s) attribut(s) permettant d'identifier de façon unique chaque tuple de la table : la clé primaire.

La clé primaire est dite simple si elle est constituée d'un seul attribut et composée dans le cas contraire.

# Modèle relationnel

Transformation du modèle Entité/Association en un modèle relationnel :

entité	↔	table
entité type	↔	nom de table
association	↔	table
propriété	↔	attribut
propriété type	↔	en-tête de colonne
occurrence	↔	tuple
identifiant	↔	clé

# Transformations particulières

Association binaire avec une cardinalité de (0,1) ou (1,1) :

- ▶ association binaire porteuse de dépendance fonctionnelle
- ▶ elle peut porter des propriétés  
(personne – voiture)

Généralement, elle peut ne pas donner naissance à une nouvelle relation.

# Transformations particulières

Association binaire avec des cardinalité de (0,n) et/ou (1,n) :

- ▶ association binaire non porteuse de dépendance fonctionnelle
- ▶ elle peut porter des propriétés (commande – porter – article)

Elle donne forcément naissance à une nouvelle relation, dont la clé est composée des identifiants des deux entités.

# Transformations particulières

## Associations ternaires ou supérieures :

- ▶ pas de dépendance fonctionnelle :
  - ▶ nouvelle relation avec une clé composée des identifiants des entités reliées
- ▶ dépendance fonctionnelle :
  - ▶ chercher à décomposer l'association d'abord
  - ▶ la clé est composée des identifiant des entités émettrices
  - ▶ l'identifiant de l'entité cible devient attribut

# Transformations particulières

## Spécialisation / sous-types

Quatre solutions principalement possibles :

- ▶ chaque sous-type exprimé par une table spécifique, en migrant l'identifiant du surtype dans les sous-types
- ▶ chaque sous-type exprimé par une table spécifique, en dupliquant toutes les propriétés du surtype
- ▶ on duplique la totalité de contenu du surtype dans les sous-types et on supprime le surtype
- ▶ on transfère la totalité des propriétés des sous-types dans la table du surtype (vues relationnelles d'une table globale).

# Transformations particulières

## Identifiant relatif et entité faible

- ▶ l'association porteuse de l'identification relative (R) ne se transforme jamais en une table
- ▶ la clé de l'entité faible est composée de l'identifiant relatif et de la clé de l'entité associée

# Transformations particulières

## Historisation

- ▶ une propriété
  - ▶ une entité
  - ▶ une association
  - ▶ une patte d'une association
- une nouvelle table

# Transformations particulières

## Contraintes ensemblistes

- ▶ sont exprimé au niveau technique (implémentés) par des tests / triggers effectués lors des insertions / modifications de tuples.

# Normalisation

- ▶ Une mauvaise conception des entités et associations représentant le monde réel modélisé conduit à des relations problématiques
- ▶ Une redondance des données conduit à des risques d'incohérences lors des mise à jour
- ▶ Il s'agit d'éliminer toute anomalie afin de faciliter la manipulation des relations

# Première forme normale

- ▶ Une relation ne contient pas d'attribut relation.
  - ▶ Sortir le groupe répétitif de la table et le transformer en une nouvelle table ; rajouter dans la clé de cette nouvelle table la clé primaire de la table initiale.

## Deuxième forme normale

concerne les tables à clé primaire composée.

- ▶ Tout attribut non-clé dépendent de la totalité de la clé primaire.
  - ▶ Regrouper dans une table les attributs dépendant de la totalité de la clé et conserver cette clé pour cette table ;
  - ▶ Regrouper dans une autre table les attributs dépendant d'une partie de la clé et faire de cette partie de clé la clé primaire de la nouvelle table.

## Deuxième forme normale

concerne les tables à clé primaire composée.

- ▶ Tout attribut non-clé dépendent de la totalité de la clé primaire.
  - ▶ Regrouper dans une table les attributs dépendant de la totalité de la clé et conserver cette clé pour cette table ;
  - ▶ Regrouper dans une autre table les attributs dépendant d'une partie de la clé et faire de cette partie de clé la clé primaire de la nouvelle table.

ARTICLE\_COMMANDE (n° commande, n° article, désignation, quantité commandée)

# Troisième forme normale

- ▶ Éliminer les dépendances transitives au sein d'une table.
  - ▶ Ne conserver dans la table que les attributs dépendant directement de la clé ;
  - ▶ Regrouper dans une autre table les attributs dépendant transitivement.

## Troisième forme normale

- ▶ Éliminer les dépendances transitives au sein d'une table.
  - ▶ Ne conserver dans la table que les attributs dépendant directement de la clé ;
  - ▶ Regrouper dans une autre table les attributs dépendant transitivement.

COMMANDE (n° commande, date, n° client, nom client )

# Normalisation

Si le modèle entité/association est bien conçu, le modèle relationnelle correspondant est déjà dans la troisième forme normale !