

# Bases de Données (USSIOG)

## Modèle Relationnel

Thibault Bernard

[Thibault.bernard@univ-reims.fr](mailto:Thibault.bernard@univ-reims.fr)

# Algèbre Relationnelle

- Modèle logique associé aux SGBD relationnels (Oracle, DB2, Mysql, Postgres ...)
- Objectifs:
  - Indépendance physique
  - Traitement de redondance des données
  - LMD non procéduraux
  - Standard

# Le modèle relationnel

- **Domaine:** ensemble fini ou infini de valeurs
  - Les valeurs d'un domaine sont atomiques
- **Relation:** sous-ensemble du produit cartésien d'une liste de domaines
  - Une relation est caractérisée par un nom
- R sur  $D_1, \dots, D_n$  est constitué
  - En-tête = ensemble fixé d'attributs tous distincts deux à deux
  - Corps = ensemble de t-uples
- Intention = **Schéma** = en-tête
- **Extension** = corps

# Exemple de relations

Vins (idVin: Entier, NomVin: chaine de caractères, région: Dom\_Vin, Millésime: Année).

Où :

- Dom\_Vin = {Alsace, Anjou, Beaujolais, Bordeaux, Bourgogne, Champagne, Côte dū Rhône, Savoie,...}.
- Année = [1900 ... 2018].

Ex: {(1, Haut Médoc, Bordeaux, 2005), (5, Château neuf du pape, Côte du rhône, 2009), (18, Aloxe Corton, Bourgogne, 2010), (27, Dom Ruinart, Champagne, 1999)}.

Très souvent les domaines sont implicites.

On note Vins = Vins(idVin, NomVin, Région, Millésime) (ie : nom (liste d'attributs))

# Modèle Relationnel / SQL

- Schéma d'une base = ensemble de toutes les relations de la base
- Différences Relations – tables:
  - Pas de lignes dupliquées, pas d'ordre entre lignes et colonnes, valeurs des attributs sont atomiques.

## **Modèle Relationnel**

- Relation
- Tuple
- Attribut
- Cardinalité
- Degré
- Clé primaire

## **SQL**

- Table
- Ligne
- Colonne
- Nbre de lignes
- Nbre de colonne
- Identifiant unique

# Contraintes d'intégrité

- Clé primaire: Ensemble d'attributs dont les valeurs permettent de distinguer les tuples les uns aux autres
    - Clé simple (1 seul attribut) ou composée (plusieurs attributs)
- Clé primaire => Irréductibilité + unicité
- Clé étrangère: attribut qui est clé primaire d'une autre relation
- => contraintes

# Contraintes d'intégrité

- Contrainte d'entité
  - Impose que toute relation possède une clé candidate et que tout attribut participant à cette clé candidate soit non nul.
- Contrainte référentielle
  - Contrainte portant sur une relation S, consistant à imposer que la valeur d'un groupe d'attributs apparaisse comme valeur de clé candidate dans une autre relation R.
- Contrainte de domaine
  - Contrainte imposant qu'un attribut d'une relation doit comporter des valeurs vérifiant une assertion logique.

# Transformation M E/A -> Modèle relationnel

- Transformation d'entité :
  - Une entité -> une relation !
  - Les propriétés de l'entité deviennent les attributs de la relation
  - L'identifiant devient clé primaire

## Exemple :

L'entité:

Vehicule

- N° immat
- Modèle
- Type
- Couleur

Devient :

Vehicule (N° immat, Modèle, Type, Couleur)



# Transformation M E/A $\rightarrow$ Modèle relationnel

Transformation d'association (0 ou 1 - n)/(0 ou 1 - n) :

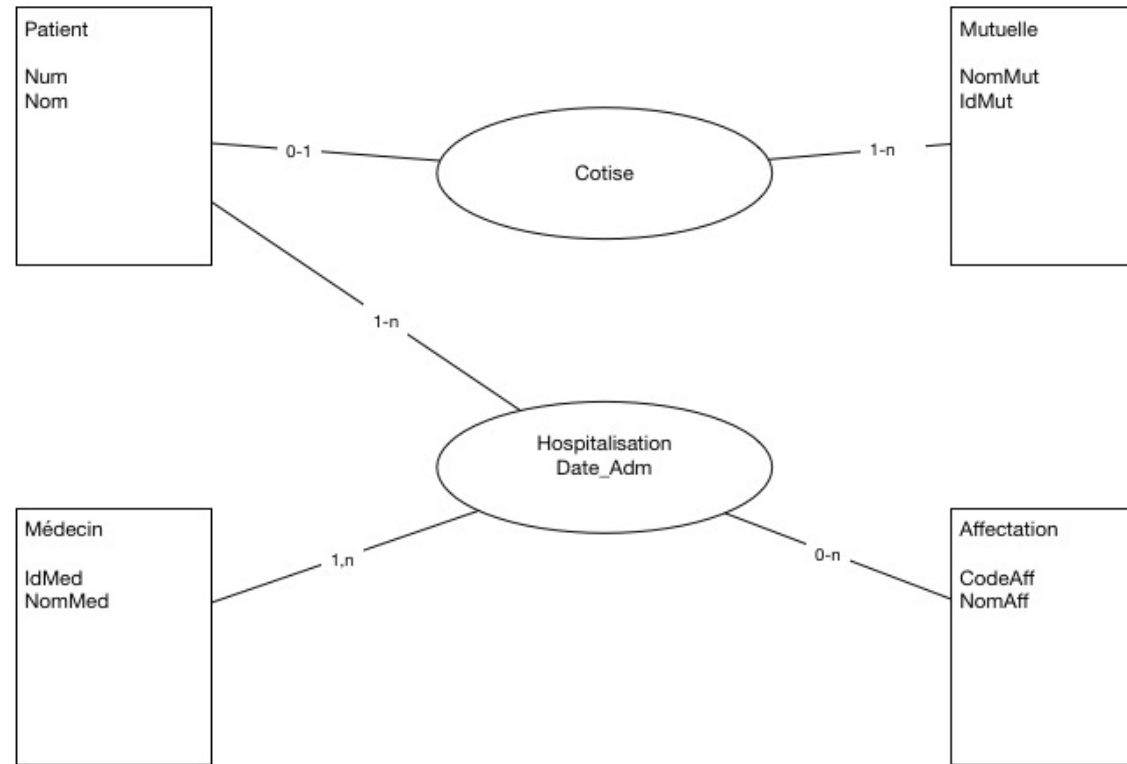
- Association entre Type d'entités E1, E2 devient une relation R
- Attributs de R = clés primaires de chacune des Type d'entités + propriétés de l'association
- Clé de R = Union des clés primaires des Types d'Entités

# Transformation M E/A -> Modèle relationnel

## Transformation des associations (1-1) / (0,1-n)

- Association entre E1 (1-1) et E2 (0,1-n)
- E1 -> R1
- E2 -> R2
- Clé de R2 devient clé étrangère de R1
- Les propriétés de l'association migrent avec la clé
- Dans le cas d'une cardinalité (0-1) la clé étrangère pourra avoir une valeur nulle.

# Exemple



# Algèbre Relationnelle

- Codd 1970
- Opérations formelles qui agissent sur des relations et produisent des relations (propriétés de fermeture)
  - Restriction
  - Projection
  - Produit
  - Union
  - Intersection
  - Différence
  - Jointure
  - Division

# Opérations ensemblistes

- Relations ayant le même schéma
  - Union de 2 relations R et S de même schéma est une relation T contenant l'ensemble des t-uples appartenant à R et S. On note  $R \cup S$ .
  - Intersection de 2 relations R et S de même schéma est une relation T de même schéma contenant l'ensemble des t-uples appartenant à la fois à R et S. On note  $R \cap S$ .
  - Différence entre 2 relations R et S de même schéma est une relation T de même schéma contenant l'ensemble des t-uples appartenant à R et n'appartenant pas à S. On note  $R \setminus S$ .

# Opérations ensemblistes

## Opérations n'ayant pas (forcément) le même schéma

- Produit cartésien de 2 relations R et S de schémas quelconques est une relation T ayant pour attributs la concaténation des attributs de R et de S et dont les t-uples sont constitués de toutes les concaténations d'un t-uple de R à 1 t-uple de S. On note  $R \times S$ .

# Opérations spécifiques

- **Affectation** sauvegarde du résultat d'une expression de recherche ou renommage d'une relation et ses attributs. On note  $R \leftarrow S$ .
- **Restriction** de la relation  $R$  par une qualification  $Q$  est une relation  $T$  de même schéma dont les t-uples sont ceux de  $R$  qui satisfont  $Q$ . On note  $\sigma_Q R$ .
- **Projection** d'une relation  $R$  de schéma  $R(A_1, \dots, A_n)$  sur les attributs  $A_{i_1}, \dots, A_{i_p}$  avec  $i_j \neq i_k$  et  $p < n$  est une relation  $T$  de schéma  $T(A_{i_1}, \dots, A_{i_p})$  dont les t-uples sont obtenus par élimination des attributs de  $R$  n'appartenant pas à  $T$  et par suppression des doublons. On note  $\pi_{A_1, \dots, A_n} R$ .

# Jointure

La jointure permet de composer 2 relations à l'aide d'un critère de jointure

- **La  $\theta$ -jointure** de R et S selon une qualification Q est l'ensemble des tuples du produit cartésien  $R \times S$  qui satisfont à la qualification Q. On note  $R \bowtie_Q S$ .
- **L'équi-jointure** entre R et S est une  $\theta$ -jointure avec pour qualification l'égalité entre 2 attributs.
- **La Jointure naturelle** de R et S est une équi-jointure sur tous les attributs de même nom dans R et dans S, suivie d'une projection qui permet de ne conserver qu'un seul de ces attributs égaux de même nom. On note  $R \bowtie S$ .



# La Division

**La division** de la relation  $R$  de schéma  $R(A_1, \dots, A_n)$  par la relation  $S$  de schéma  $S(A_{p+1}, \dots, A_n)$  est la relation  $T$  de schéma  $T(A_1, \dots, A_p)$  formés de tous les tuples qui concaténées à chaque tuples de  $S$  donnent toujours un tuple de  $R$ . On note  $R/S$ .

# Opérations de calcul

- **Compte** dénombre le nombre de lignes d'une relation  $R$  qui ont une même valeur d'attributs en commun.
- **Somme** fait la somme cumulée des valeurs d'un attribut  $A$  pour chacune des valeurs différentes des attributs de regroupement  $A_1, \dots, A_n$ .  $A$  doit être numérique.

# Notations et Remarques

- Algèbre relationnelle : représentation symbolique et de haut niveau des intentions de l'utilisateur
- A partir de l'AR il est possible de composer un Langage d'Interrogation des BD
- Question = arbre d'opérateurs relationnels
- Paraphrasage de ces expressions en anglais est à la base de SQL.
- Arbre relationnel : arbre dont les noeuds correspondent à des opérations de l'AR et les arcs à des relations de bases ou temporaires représentant des flots de données entre opérations.

# Exemples

Soit le schéma relationnel suivant :

- **Patient**(n°SS, NomP, PrenomP, TelP, AdresseP, VilleP, #CodeMut)
- **Docteur**(CodeDoc, NomDoc, PrenomDoc, TelDoc AdresseDoc, VilleDoc)
- **Mutuelle**(CodeMut, NomMut, TelMut, AdresseMut, VilleMut)
- **Hospitalise**(n°SS, CodeDoc, IdS, CodeAff, Date)
- **Service**(IdS, Dénomination, Nom\_Hopital, TelS, VilleS)
- **Affection**(CodeAff, NomAff, Dangersité)