

420-C42

Langages d'exploitation des bases de données

Partie 2

Le modèle relationnel

Selon Wikipédia

- En bref, le **modèle relationnel** (MR) est :
 - une approche de gestion des données
 - utilisant une structure et un langage conformes à une logique formelle permettant des déclarations plus expressives
 - basée sur des bases mathématiques éprouvées : théorie ensembliste, algèbre relationnel, calcul relationnel, théorie des domaines et des tuples, logique du premier ordre, fonctions et correspondances, ...
 - proposé en 1969 par le scientifique informatique anglais Edgar F. Codd
 - toutes les données sont représentées en termes de tuples formés d'attributs, regroupés en relations
- Une base de données organisée selon le modèle relationnel est une base de données relationnelle.
- Voir ces articles Wikipédia en [français](#) et en [anglais](#).

Généralité

- L'intention du MR est :
 - de créer une méthode déclarative paradigme déclaratif vs impératif ou procédural
on décrit ce que l'on désire et non les mécanismes d'obtention
le quoi et non le comment
 - permettant de spécifier :
 - l'organisation des données
 - ainsi que les moyens d'y accéder.
- C'est le langage SQL qui donne naissance au MR en ce sens.

Relation (table)

- Le MR est simple, il n'est constitué que d'un seul élément principal : **la relation**
- Dans le jargon du MR on nomme *relation* une entité qu'on appelle plus communément une *table*. Pour le reste de ce cours, nous utiliserons le terme *table*.
- La *table* est définie ainsi :
 - elle porte un nom;
 - elle est composée d'attribut(s) nommé(s) ayant différentes contraintes; colonnes
 - elle est constituée de données appelées tuple respectant les contraintes. lignes
- Deux concepts fondamentaux au MR :
 - pour une *table*, chaque tuple est unique et contraint par le concept de clé primaire;
 - les liaisons existantes entre différents tuples (de la même *table* ou de *tables* différentes) se font par le concept de clé étrangère.
- Toutes les opérations du MR utilisent des *tables* en entrée **et** en sortie.

Attributs et tuples

- Les attributs :
 - définissent la structure de la *table*
 - correspondent aux colonnes de la *table*
 - possèdent des contraintes : types, unicité, existence, ...
- Les tuples :
 - définissent les données de la *table*
 - correspondent aux lignes de la *table*
 - chacun décrit un enregistrement unique (les données d'un même sujet)
 - les données respectent les contraintes des attributs

Table, attributs et tuples

Table

Employe					
NAS	Nom	Prenom	Courriel	Département	Salaire
123	Dupuis	Lancelot	ldupuis@abc.com	Ventes	28.50
987	Bordeleau	Marina	mbordeleau@abc.com	R&D	55.25
666	Pignon	François	fpignon@abc.com	Achats	19.75
456	Leblanc	Juste	jleblanc@abc.com	Administration	32.00
741	Brochant	Pierre	pbrochant@abc.com	Ventes	65.00

← Nom de la table

← Liste des attributs

← Tuple 1

← Tuple 2

← Tuple 3

← Tuple 4

← Tuple 5

ATTENTION

L'ordre des tuples est fictif et simplement pour faciliter la représentation visuelle.

L'ordre n'existe pas!

Attribut 6 : Salaire

Attribut 5 : Département

Attribut 4 : Courriel

Attribut 3 : Prenom

Attribut 2 : Nom

Attribut 1 : NAS

Contraintes

- Il existe plusieurs formes de contraintes :
 - les contraintes sur les attributs;
 - les contraintes sur les tuples;
 - les contraintes sur les valeurs;
 - les contraintes d'intégrité;
 - les contraintes sur la base de données.

Contraintes : attributs & tuples

- Les contraintes sur les attributs (colonnes) sont :
 - chaque relation possède un nombre déterminé d'attributs $n_a > 0$
 - obligatoirement, un type de données doit être assigné à chaque attribut
 - optionnellement, des contraintes peuvent être assignées à chaque attribut: valeur nulle autorisée ou non, unicité, validation personnalisée, ...
- Les contraintes sur les tuples (lignes) sont :
 - chaque relation possède un nombre fini de tuple $n_t \geq 0$
 - les données respectent les contraintes données aux colonnes
 - une relation ne peut avoir deux tuples identiques (*pas tout à fait vrai*)

Contraintes : valeurs

- Les contraintes sur les valeurs sont :
 - Les valeurs doivent respecter les contraintes données aux attributs
 - chaque valeur doit être **atomique**
 - si l'existence d'une donnée n'est pas requise, elle peut posséder une valeur nulle dont l'interprétation doit être documentée :
 - aucune valeur applicable (date de mariage d'une personne célibataire)
 - aucune valeur connue (date de naissance inconnue)

Contraintes d'intégrité

- Il existe trois contraintes d'intégrité :
 - Intégrité de domaine
 - donnée correspondant au type et aux contraintes de l'attributs
 - valeur atomique
 - Intégrité d'enregistrement (clé primaire)
 - chaque relation doit posséder une clé primaire définie par un ou plusieurs attributs
 - tous les tuples d'une même table doivent avoir une valeur unique de clé primaire
 - une clé primaire ne peut être nulle
 - Intégrité référentielle (clé étrangère)
 - chaque tuple peut posséder n clé(s) étrangère(s) (liaison(s) vers d'autres données)
 - une clé étrangère ne peut avoir que deux états possibles :
 - la valeur de la donnée liée (la donnée liée doit exister)
 - la valeur nulle s'il n'existe pas de liaison et que l'attribut autorise la valeur nulle

Clés primaires et étrangères

- Concept de clé primaire
 - Défini par n_p attributs : $1 \geq n_p \geq n_a$
 - Chaque tuple doit avoir une valeur de clé primaire différente de tous les autres tuples.
 - La clé primaire possède systématiquement les contraintes suivantes :
 - valeur nulle interdite
 - unicité
 - Attention, l'adoption de la contrainte de clé primaire est :
 - obligatoire pour la théorie du MR
 - optionnelle pour l'utilisation de SQL
- Concept de clé étrangère
 - Défini un lien entre des données différentes (de la même table ou d'une autre table).
 - Défini par n_e attributs :
 - $n_e = \text{aux données liées}$
 - *la(les) colonne(s) liée(s) doit(doivent) :*
 - avoir la contrainte d'unicité
 - être de même type
 - valeur nulle permise

Clés primaires et étrangères

Clé primaire l'attribut ou les attributs soulignés

- l'attribut *NAS* constitue la clé primaire de la table *Employe*

Employe					
<u>NAS</u>	Nom	Prenom	Courriel	Département	Salaire
123	Dupuis	Lancelot	ldupuis@abc.com	3	28.50
987	Bordeleau	Marina	mbordeleau@abc.com	7	55.25
666	Pignon	François	fpignon@abc.com	4	19.75
456	Leblanc	Juste	jleblanc@abc.com	2	32.00
741	Brochant	Pierre	pbrochant@abc.com	3	65.00

Clé primaire l'attribut ou les attributs soulignés

- l'attribut *ID* constitue la clé primaire de la table *Departement*

Departement		
<u>ID</u>	Nom	Superviseur
2	Administration	456
3	Ventes	123
4	Achats	666
7	R&D	987

Clé étrangère une flèche dirigé de *n* attributs vers *n* autres attributs

- la colonne *Département* de la table *Employe* est liée à la colonne *ID* de la table *Departement*
- indiquant que le département d'un employé est :
 - défini dans la table *Employe*
 - lié à un département existant dans la table *Departement* – ou y retrouvera toutes les informations liées au département

Clé étrangère une flèche dirigé de *n* attributs vers *n* autres attributs

- la colonne *Superviseur* de la table *Departement* est liée à la colonne *NAS* de la table *Employe*
- indiquant que le superviseur d'un département est :
 - défini dans la table *Departement*
 - lié à un employé existant dans la table *Employe* – on y retrouvera toutes les informations liées à l'employé

Contraintes : base de données

- Les contraintes sur la base de données sont :
 - Une base de données est un ensemble logique constituée de
 - *tables* ou *relations* : $n_r > 0$
 n_r = nombre de tables (de relations)
 - d'objets supplémentaires (on y reviendra)
 - Chaque relation et objet doit avoir un nom unique dans la base de données.
- Finalement, le concepteur doit éviter la redondance d'information en créant une seule entrée du même enregistrement dans tout le système!

Gestion des contraintes

- Il existe plusieurs contraintes gérées automatiquement par les SGBD :
 - contraintes implicitement définies lors de la création de la table :
 - validation de type
 - atomicité (implicite*)
 - contraintes explicitement ajoutées par le créateur de la table :
 - contrainte d'intégrité :
 - d'enregistrement : clé primaire
 - référentielle : clé étrangère
 - contrainte de valeur :
 - nulle permise ou non
 - unique
 - par défaut
 - avec validation personnalisée

Liaison entre les données

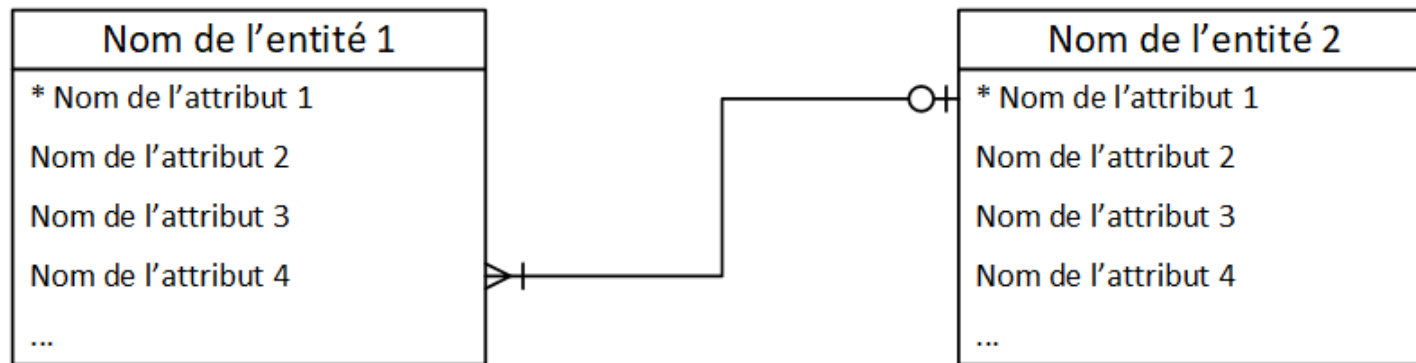
- Il existe trois types de liaison du monde réel qui doivent être représentés dans le modèle relationnel :
 - Relation 1 vers 1 :
Étudiant <---> Carte étudiant
 - Relation 1 vers n :
Étudiant <---> Programme
 - Relation m vers n :
Étudiant <---> Cours

Notation graphique

- Il existe un plusieurs notations graphiques permettant la schématisation de la structure d'un système informationnel :
 - que ce soit pour des étapes préliminaire de conception (entités-associations)
 - ou pour la base de données elle-même (schéma relationnel)

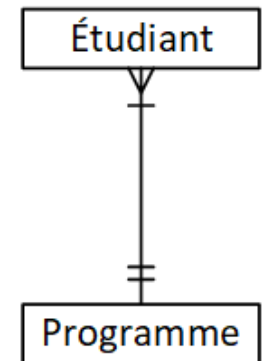
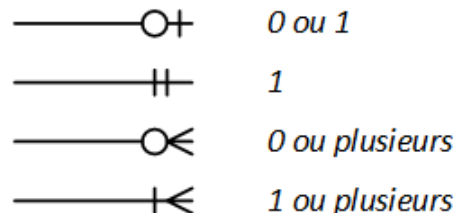
Notation graphique

- La notation à patte d'oiseau (« *crow foot notation* ») est très utilisée pour décrire un système informationnel par ses entités et ses associations (souvent lié à la conception d'un système avant la réalisation de la BD).
- Attention, cette notation n'est pas un schéma de base de données détaillé mais correspond plutôt à une étape préliminaire de la conception avec une représentation partielle. Typiquement, le travail d'un analyste.



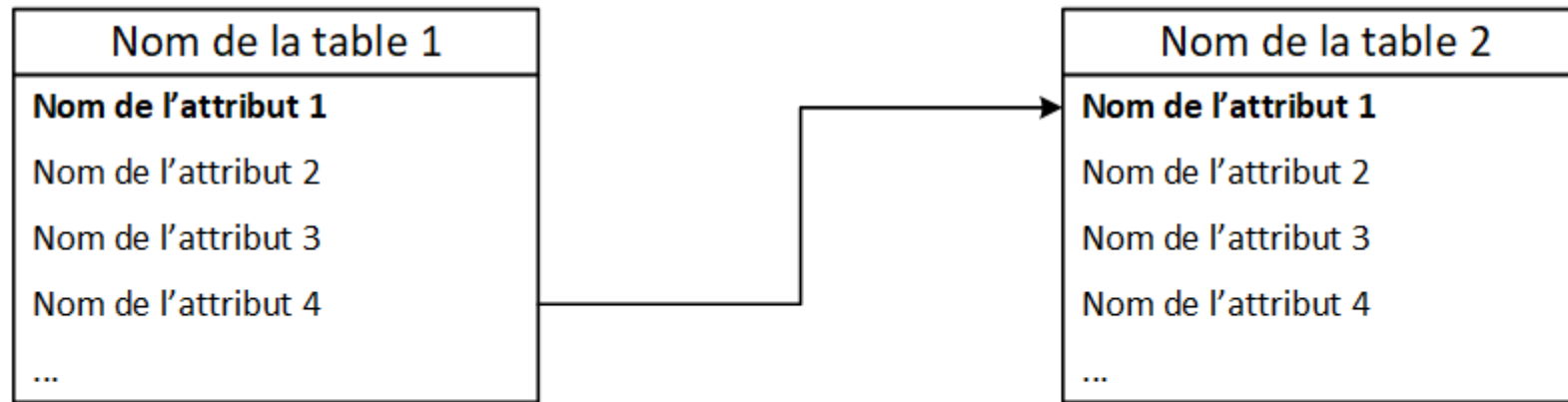
Notes :

- L'astérisque indique que l'attribut participe à la clé primaire
- La ligne indique que les deux entités sont associées l'un à l'autre (les symboles indiquent la cardinalité de l'association)



Notation graphique

- Le schéma relationnel est un schéma qui représente la structure d'une base de données (avec une représentation partielle des contraintes).



Notes :

- Le caractère gras indique que l'attribut participe à la clé primaire
- Une flèche indique une clé étrangère
(l'origine de la flèche indique la clé étrangère alors que la pointe de la flèche indique la référence)

Un exemple simple

- Un premier exemple fait ensemble :
 - *twitter* considérant :
 - Les usagers
 - Les messages
 - Les « followers »

Le modèle relationnel

- Un deuxième exemple en exercice de groupes :
 - une école considérant
 - les étudiants
 - les enseignants
 - les cours
 - les départements