

Le Modèle Relationnel

HAI502I

Pascal Poncelet

LIRMM

Pascal.Poncelet@lirmm.fr

<http://www.lirmm.fr/~poncelet>



Domaine

- Un **domaine D** est un **ensemble de valeurs caractérisé par un nom**. Du point de vue du modèle relationnel, chaque valeur du domaine est atomique et donc indivisible
 - Cette notion permet de définir les ensembles de départ. Un domaine peut être défini en **extension** en donnant la liste des valeurs composantes ou en **compréhension** en définissant une propriété caractéristique du domaine.

COULEUR = { jaune ; vert ; rouge ; bleu ; rose ; orange ; pourpre }

ABONNE = { Personne possédant une carte d'abonné valide pour l'année en cours }



Domaine

- Du point de vue de la réalisation informatique, le domaine se restreint à la notion de **type de données**
- Néanmoins, il est essentiel au cours de l'étape de conception de clairement définir les domaines – (Voir Cours Conception – Dictionnaire de données)
- Attention domaine sémantique vs. Domaine syntaxique



Exemples de domaines

Nom du domaine	Sémantique	Syntaxique
NOM_PILOTE	Noms des pilotes de la compagnie	STRING/VARCHAR
SALAIRE	Salaire des pilotes de la compagnie	REAL/FLOAT
ADRESSE	Adresse des pilotes de la compagnie	STRING/VARCHAR

Schéma de relation

- Un **schéma de relation R**, dénoté
$$R(A_1:D_1, A_2:D_2, \dots, A_n:D_n)$$
est un ensemble d'attributs
- Chaque attribut A_i est le nom d'un rôle joué par son domaine D_i dans le schéma de relation R
- Un schéma de relation R est utilisé pour décrire une relation



Schéma de relation

- Autre définition :
 - une relation R correspond au sous ensemble du produit cartésien de n domaines :

$$R \subseteq D_1 \times D_2 \times D_3 \times \dots \times D_n$$

- n : degré de la relation
- attribut : rôle joué par un domaine dans une relation



Exemple

PILOTE : NUM_PILOTE × NOM_PILOTE × ADRESSE × SALAIRE

Pilote (NumPil, NomPil, adr, sal)

AVION : NUM_AVION × NOM_AVION × ADRESSE × CAP

Avion (NumAv, AvNom, loc, cap)

VOL : NUM_VOL × NUM_PILOTE × NUM_AVION × VILLE × VILLE ×
HEURE × HEURE

Vol (NumVol, NumPil, NumAv, Ville_dep, Ville_arr, Heure_dep, Heure_arr)



Schéma de relation (suite)

- Définir un schéma de relation revient à **spécifier un nouveau type de données** équivalent à un type STRUCT en C par exemple
- **Attention : Le modèle relationnel n'autorise qu'un seul niveau de structure (i.e., première forme normale ... voir cours dépendances)**
 - Il n'est pas possible par exemple de définir l'attribut Adresse qui se décompose en Rue, Ville et Code Postal



Relation

- Une **relation** r dénotée $r(R)$ du schéma de relation $R(A_1:D_1, A_2:D_2, \dots, A_n:D_n)$ est un ensemble d'enregistrements
- Chaque enregistrement e_i est une liste ordonnée de n valeurs $e_i = \langle v_1, v_2, \dots, v_n \rangle$ où chaque v_i est une valeur du domaine de l'attribut A_i ou une valeur nulle spéciale (**NULL**) représentant l'absence d'information
 - Attention la présence de valeurs nulles dans une relation est souvent difficile à interpréter !!



Relation

- Extension = ensemble d'enregistrements (tuples/n-uplets) = relation
- Intention = ensemble des attributs avec leur domaine = schéma de relation
 - Les 2 notions cohabitent en permanence : *le schéma de la base est défini par le concepteur alors que chaque relation correspond à la réalisation à un instant donné de la base de données elle même (ETAT)*



Représentation d'une relation

PILOTE	NumPil	NomPil	Adresse	Sal
<div>Nom de la relation</div> <div>Attribut</div> <div>tuple</div>	1	Dupond	Nice	15000
	2	Dupré	Paris	20000
	3	Duchamp	Toulouse	9000
	4	Dujardin	Nîmes	17000
	5	Dupond	Paris	18000

Degré = nombre d'attributs = 4

Cardinalité = nombre de tuples = 5



Clé d'une relation

- Une clé de relation **est un sous-ensemble d'attributs** qui permet de caractériser tout enregistrement d'une relation

Attribut ou ensemble d'attributs qui permet d'identifier de manière unique chaque tuple de la relation

- Par définition, une relation est un ensemble d'enregistrements et il ne peut donc pas y avoir deux enregistrements strictement identiques dans la même relation



Clé d'une relation

- Il existe généralement un sous-ensemble SC d'attributs d'un schéma de relation R pour lequel deux enregistrements de toute relation $r(R)$ ne peuvent avoir la même combinaison de valeurs pour ces attributs :

Quelque soit $t_1, t_2 \in r(R)$, $t_1 [SC] \neq t_2 [SC]$
où $t_i [X]$ correspond à la valeur du tuple t_i pour la colonne X

- Tout ensemble d'attributs vérifiant cette propriété est appelé **superclé** du schéma R
- Il existe au moins une superclé qui est l'ensemble de tous les attributs



Clé d'une relation (suite)

- Une clé est invariante dans le temps.
- En général, il existe plusieurs clés pour une même relation R
- Parmi les clés possibles, on choisit une clé qui sera appelée clé primaire, les autres seront appelées candidates
- Lors de la définition d'un schéma cette clé est mise en évidence (soulignement ou gras)



Clé (suite)

- Domaine primaire : domaine de définition d'un attribut clé primaire
- Clé étrangère : attribut défini sur un domaine primaire et qui n'est pas clé primaire dans sa relation
- 2 types de relations :
 - Relations statiques (pas de CE, indépendantes des autres)
 - Relations dynamiques (avec CE)



Exercices

- Dans les relations pilotes, avions et vols déterminer les relations statiques et dynamiques et préciser les différentes clés
- Donner les différents schémas de relations pour Prof, Etudiant et Enseignements




Exercices

Pilote (**NumPil**, NomPil, adr, sal) Relation statique

Avion (**NumAv**, AvNom, loc, cap) Relation statique

Vol (**NumVol**, *NumPil*, *NumAv*, Ville_dep, Ville_arr,
Heure_dep, Heure_arr) Clés étrangères

Relation dynamique

The diagram consists of two blue arrows. The first arrow originates from the attribute 'NumPil' in the 'Vol' relation and points to the attribute 'NumPil' in the 'Pilote' relation. The second arrow originates from the attribute 'NumAv' in the 'Vol' relation and points to the attribute 'NumAv' in the 'Avion' relation. These arrows indicate that 'NumPil' and 'NumAv' are foreign keys in the 'Vol' relation that reference the primary keys of the 'Pilote' and 'Avion' relations, respectively.

Attention Numpil et Numav ne peuvent pas être clés primaires. Pourquoi ?



Exercices

Un Professeur a plusieurs Etudiants

Professeur (**NumProf**, NomProf, Ville, ???)

Un Professeur peut avoir 1 étudiant

Professeur (**NumProf**, NomProf, Ville, *NumEt*)

Un Professeur peut avoir 2 étudiants

Professeur (**NumProf**, NomProf, Ville, *NumEt1*, *NumEt2*)

Un Professeur peut avoir n étudiants

Professeur (**NumProf**, NomProf, Ville, *NumEt1*, *NumEt2*,
....., *NumEtn*)



Exercices

Un Professeur peut avoir n étudiants

Professeur (**NumProf**, NomProf, Ville, *NumEt1*, *NumEt2*,
....., *NumEtn*)

NumProf	NomProf	Ville	NumEt1	NumEt2	...	NumEtn
1	DUPOND	NICE	100	101	...	104
2	DURAND	PARIS	100	NULL	NULL	NULL
3	DUPONT	LILLE	104	105	NULL	NULL
4	DUCHEMIN	NICE	NULL	NULL	NULL	NULL

RAPPEL : Attention la présence de valeurs nulles dans une relation est souvent difficile à interpréter !!

Problème : on ne connaît pas à l'avance la valeur de n qui peut être très grand



Exercices

Un Professeur a plusieurs Etudiants

Un Etudiant a plusieurs Professeurs

- « Un professeur vit indépendamment des étudiants »
- « Un étudiant vit indépendamment des professeurs »
- « Ils se rencontrent lors d'un COURS »

Professeur (**NumProf**, NomProf, Ville) (*Relation Statique*)

Etudiant (**NumEt**, Nom, Prenom) (*Relation Statique*)

Cours (**NumProf**, **NumEt**, Note) (*Relation Dynamique*)



Exercices

<i>NumProf</i>	<i>NumEt</i>	Note
1	100	15
1	101	17
1	104	12
2	100	13
3	104	15
3	105	18

Le professeur 1 a comme étudiants 100, 101 et 104

L'étudiant 100 a comme professeur 1 et 2



Exercice

- Etudiant (Numero_Etudiant, Num_Secu, Nom, Prenom, Adresse, Telephone)
- Quelle est la clé primaire ?



Schéma de base de données et CI

- Un **schéma de base de données relationnel** **S** est un ensemble de schémas de relation

$$S = \{R_1, R_2, \dots, R_p\}$$

et un ensemble de **contraintes d'intégrité** **CI**

- Une **instance de base de données relationnelle** **BD** est un ensemble d'instances de relations

$$BD = \{r_1, r_2, \dots, r_n\}$$

où chaque r_i respecte les contraintes d'intégrité



Schéma de base de données et CI

- Une contrainte d'intégrité est une propriété du schéma, invariante dans le temps
- Il existe différents types de contraintes d'intégrité :
 - Structurelles ou statiques (liées au modèle relationnel)
 - Applicatives ou dynamiques (contraintes de cohérences liées à l'application)



Les contraintes du modèle

- CI de domaine
 - «toute valeur d'un attribut doit appartenir à son domaine de définition»
- CI de relation
 - «toute valeur de clé primaire existe et est unique»
- CI de référence
 - «Toute valeur de CE existe dans la CP associée»
 - la valeur d'attribut de la relation r_1 doit apparaître comme valeur de clé dans une autre relation r_2



- Des questions ?

