

CHAPITRE 4

Le Modèle Relationnel

Le modèle relationnel a été inventé en 1960 et a fait l'objet de très nombreuses recherches qui ont débouché sur la réalisation et commercialisation de SGBDs relationnels. C'est le modèle le plus utilisé par les SGBDs actuellement disponibles sur le marché.

C'est un modèle de données plus simple que celui de l'entité association, ce qui explique son succès tant sur le plan théorique (théorie de la normalisation, définition de langages de manipulation de données), que sur celui des réalisations. Mais cette simplicité est aussi sa faiblesse principale: c'est un outil trop pauvre sémantiquement pour pouvoir bien représenter la complexité du monde réel. C'est pourquoi d'autres modèles de type entité association ou orientés objets ont été ensuite proposés.

1. Définitions

Les objets et associations du monde réel sont représentés par un concept unique: la relation. Les relations sont des tableaux à deux dimensions, souvent appelés tables.

Exemples

Etudiant	N° Etud	Nom	Prénom	Age
	136	Rochat	Jean	19
	253	Aubry	Annie	20
	101	Duval	André	20
	147	Rochat	Marc	21

Cet exemple représente une relation (ou table) décrivant les étudiants. Etudiant est le nom de la relation. Les entêtes des colonnes, N° Etud, Nom, Prénom et Age, sont les **attributs** de la relation. Chaque ligne de la table correspond à une occurrence. Par exemple:

<253, Aubry, Annie, 20>

constitue un n-uplet ou **tuple**, qui décrit une occurrence (l'étudiante Annie Aubry).

Il est usuel de souligner l'attribut, ou les attributs, qui constituent l'identifiant de la relation.

Cours	<u>Nom C</u>	Horaire	Prof
	Algo	Lundi 10-12	Duval
	Système	Mardi 16-17	Malin

Suit	<u>N° Etud</u>	<u>NomC</u>
	253	Algo
	136	Système
	253	Système

On remarque que l'identifiant de la relation "Suit" (qui traduit un type d'association) est composé des identifiants des deux relations précédentes. Cette relation Suit exprime le lien entre un étudiant, désigné par son numéro, et un cours, désigné par son nom. Si on avait utilisé le modèle entité association, les relations Etudiant et Cours auraient été modélisées par des types d'entité, Etudiant et Cours, alors que la relation Suit aurait été modélisée par un type d'association reliant ces types d'entité.

Notion de domaine

Un domaine est un ensemble de valeurs que peut prendre un attribut; c'est le domaine définition d'un ou plusieurs attributs.

Exemple de domaines:

Dnom : chaînes de caractères de longueur maximale 30

Dnum : entiers compris entre 0 et 99999

Dcouleur : {"bleu", "vert", "jaune"}

Dâge : entiers compris entre 16 et 65

Relations

Une relation est définie par :

- son nom
- liste de couples (nom d'attribut : domaine)
- son (ses) identifiant(s)
- sa définition (phrase en français)

Les trois premières informations: nom de la relation, liste des couples (attribut : domaine) et identifiant(s) constituent le **schéma** de la relation.

Exemple : schéma de la relation Etudiant :

Etudiant (N° Etud : Dnum, Nom : Dnom, Prénom : Dnom, Age : Dâge)

La **population** d'une relation est constituée de l'ensemble des tuples de la relation. C'est un ensemble; il n'y a donc ni doubles, ni ordre (les nouveaux tuples sont rajoutés à la fin de la relation).

On appelle **schéma d'une base de données relationnelle** l'ensemble des schémas de ses relations. On appelle base de données relationnelle, la population de toutes ses relations.

2. Règles de modélisation

Les attributs sont tous simples et monovalués: toute valeur prise par un attribut pour un tuple est atomique (non décomposable par le SGBD) et unique. Les notions d'attribut multivalué, complexe ou facultatif, n'existent pas dans le modèle relationnel.

Cas d'un attribut facultatif du modèle entité association

Certains SGBDs relationnels gèrent une valeur spéciale, appelée valeur nulle et notée "?". Cette valeur peut être prise par tout attribut (sauf si dans le schéma on a spécifié le contraire); elle signifie alors que le tuple n'a pas de valeur pour cet attribut. Par exemple, un étudiant, Marc Dumont, de numéro 123, dont on ne connaîtrait pas l'âge, serait représenté par le tuple:

< 123, Dumont, Marc, ? > .

Une telle solution implique un traitement particulier de cette valeur nulle dans les langages de manipulation des données.

A l'opposé, les attributs obligatoires sont traduits lors de la création de la table en SQL par la clause NOT NULL.

Cas d'un attribut complexe du modèle entité association

Soit par exemple, un attribut complexe en entité association, Adresse, composé des attributs: n° rue, nom rue, ville, code-postal. Si on veut ajouter cet attribut dans la relation Etudiant, il y a plusieurs solutions.

Solution 1: On ajoute l'attribut Adresse qui a pour domaine les chaînes de caractères. Dans ce cas, l'utilisateur ne peut pas poser de questions sur la ville, il devra lire l'adresse et chercher dans la chaîne de caractères le nom de la ville lui-même, ou par programme.

Solution 2: On ajoute les attributs suivants : n°rue, nom-rue, ville, code-postal à la relation. Le SGBD connaît le détail de l'adresse, mais ne connaît pas la notion globale d'adresse.

La solution est choisie en fonction du type d'utilisation de l'attribut.

Cas d'un attribut multivalué du modèle entité association

Par exemple, on veut mémoriser les différents prénoms des étudiants (et non pas uniquement leur premier prénom).

Solution 1: Avoir dans la relation Etudiant plusieurs attributs : Prénom1, Prénom2,...

Problèmes : combien mettre d'attributs Prénom ? Comment poser des questions sur cet attribut (on est obligé de poser autant de questions que d'attributs déclarés !).

C'est une mauvaise solution à ne pas utiliser.

Solution 2: On ne garde que N°Etud, Nom, Age pour la relation Etudiant, et on crée une relation supplémentaire, EtudPrénoms :

EtudPrénoms	<u>N°Etud</u>	<u>Prénom</u>
	136	Jean
	136	Marie
	136	André
	253	Annie
	253	Claudette
	101	André
	147	Marc
	147	Antoine

Cette solution n'a pas les inconvénients de la solution précédente.

Si l'on veut, plutôt que l'ensemble des prénoms, la liste ordonnée des prénoms on décrira la relation: EtudPrénoms2 (N°Etud, N°Prénom, Prénom).

3. Identifiant d'une relation

Définition : L'identifiant (aussi appelé clé) d'une relation est un ensemble minimum d'attributs de la relation, tel qu'il n'existe pas deux tuples ayant même valeur pour cet identifiant.

C'est un ensemble minimum d'attributs tel que tous les autres attributs en dépendent fonctionnellement. ♦

Un identifiant peut-être composé d'un ou plusieurs attributs. Une relation peut avoir un ou plusieurs identifiants. Une relation étant un ensemble, elle a toujours un identifiant qui, dans le cas le plus défavorable, est composé de tous les attributs de la relation. Par convention, l'attribut (ou les attributs) constituant l' (ou un des) identifiant(s) est souligné.

Exemples

Etudiant (N°Etud, nom, prénom, age)

Il n'y a pas deux étudiants qui ont le même numéro.

La relation précédente, EtudPrénoms2, possède deux identifiants qui sont N°Etud, N°Prénom et N°Etud, Prénom.

On ne peut pas avoir de valeur inconnue (notée "?") pour un identifiant (on entre obligatoirement un numéro d'étudiant). Si on entrerait deux tuples sans N°Etud, alors il existerait deux étudiants avec la même valeur pour l'identifiant (valeur inconnue), ce qui est impossible d'après la définition de l'identifiant. On a donc la règle suivante:

Règle: tous les attributs de tout identifiant doivent toujours avoir une valeur connue. ♦

Exemple

Insérer dans Etudiant: < ?, Rochat, Jean, 19> est une instruction que le SGBD doit refuser.

4. Identifiant externe

Certains attributs référencent des tuples d'une autre relation (ou parfois de la même); c'est à dire que leur valeur est toujours égale à celle de l'identifiant d'un tuple existant dans l'autre relation. On les appelle identifiants externes (ou clés externes ou clés étrangères). Par exemple, la relation Suit (NomC, N°Etud) possède un identifiant (NomC + N°Etud), et deux identifiants externes : NomC et N°Etud. En effet, NomC "référence" un Cours, c'est à dire que si une valeur NomC existe dans Suit, alors il doit nécessairement exister un cours de ce nom là dans la relation Cours. De même, N°Etud "référence" un Etudiant.

Le schéma d'une relation comprend donc, en plus de la définition du nom de la relation et de ses attributs et domaines associés, la définition de son (ses) identifiant, et celle de ses identifiants externes, s'il en existe.

Les identifiants externes sont déclarés de la façon suivante:

Suit (N°Etud : Dnum, NomC : Dnom)

Identifiants externes:

N°Etud **référence un** Etudiant

NomC **référence un** Cours

Dans le cas où la relation référencée possède plusieurs identifiants il faut préciser lequel. La clause est alors:

N°Etud **référence un** Etudiant•N°Etud

Définition :

Soient deux relations R1(X, Y) et R2 (V, W), où X, Y, V, W, désignent des attributs ou des ensembles d'attributs, et où X est un identifiant de R1, on dit que W est un identifiant externe sur R1 (ou que W référence R1) si pour tout tuple de R2, la valeur prise par W est nécessairement la valeur de X pour un tuple existant de R1. Autrement dit, à tout instant, l'ensemble des valeurs prises par W est compris dans l'ensemble des valeurs prises par X. ♦

Vérification automatique assurée par le SGBD :

Une fois déclaré l'identifiant externe W de R2 sur R1, le SGBD vérifie toutes les insertions dans R2 (il vérifie que la valeur de W existe dans R1); il vérifie de la même façon les modifications de W. Il vérifie toutes les suppressions de tuples de R1 (il vérifie qu'il n'existe pas de tuple dans R2 référençant ce tuple à supprimer). Le SGBD assure ainsi, l'**intégrité de référence** (ou intégrité référentielle) de la base de données.

5. Conclusion

Un schéma relationnel se compose:

pour chaque relation de:

- nom de la relation
- définition
- attributs + domaines
- identifiant(s)
- éventuellement identifiant(s) externe(s)
- + contraintes d'intégrité associées à cette relation.

et des autres contraintes d'intégrité qui portent sur plusieurs relations.

Exemple

Base de données relationnelle "FormaPerm"

Un exemple de schéma de base de données relationnelles et du contenu de la base pour une entreprise de formation permanente est proposé ci-dessous :

1. Schéma

Domaines:

Dnom : chaînes de caractères de longueur inférieure à 30
Dch100 : chaînes de caractères de longueur inférieure à 100
Dannée : [1970 : 1990]
Dnote : [0.0 : 10.0]
Ddate : [1 : 31] / [1 : 12] / [1920 : 2100]

Relation : *Personne*

Attributs: *n°P* : entier sans nul
nom : *Dnom* sans nul
adr : *Dch100* sans nul

Identifiant: (*n°P*)

Définition: tout étudiant et tout enseignant de l'institut.

Relation : *PersonnePrénoms*

Attributs: *n°P* : entier sans nul
prénom : *Dnom* sans nul
n°prénom : entier sans nul

Identifiants: (*n°P* + *prénom*) , (*n°P* + *n°prénom*)

Identifiant externe : *n°P* référence une *Personne*

Définition: prénoms des personnes

Relation : *Etudiant*

Attributs: *n°P* : entier sans nul
n°E : entier sans nul
dateN : *Ddate* sans nul

Identifiants: (*n°E*)
(*n°P*)

Identifiant externe : *n°P* référence une *Personne*

Définition: tout individu qui est actuellement inscrit à l'institut, ou qui a déjà passé avec succès un des cours de l'institut

Relation : *EtudiantEtudes*

Attributs: *n°E* : entier sans nul
année : *Dannée* sans nul
diplôme : *Dnom* sans nul

Identifiant: (*n°E* + *diplôme*)

Identifiant externe : *n°E* référence un *Etudiant*.*n°E*

Définition: études antérieures des étudiants

Relation : Enseignant

Attributs: $n^{\circ}P$: entier sans nul
 tel : entier sans nul
 $statut$: $Dnom$ sans nul
 $banque$: $Dnom$ sans nul
 $agence$: $Dnom$ sans nul
 $compte$: entier sans nul

Identifiant: ($n^{\circ}P$)

Identifiant externe : $n^{\circ}P$ référence une *Personne*

Définition: tout individu assurant actuellement un ou plusieurs cours à l'institut

Relation : Cours

Attributs: $nomC$: $Dnom$ sans nul
 $cycle$: entier sans nul
 $n^{\circ}Ens$: entier sans nul

Identifiant: ($nomC$)

Identifiant externe : $n^{\circ}Ens$ référence un *Enseignant*

Définition: tout cours offert par l'institut

Relation : Obtenu

Attributs: $n^{\circ}E$: entier sans nul
 $nomC$: $Dnom$ sans nul
 $note$: $Dnote$ sans nul
 $année$: $Dannée$ sans nul

Identifiant: ($n^{\circ}E + nomC$)

Identifiants externes : $n^{\circ}E$ référence un *Etudiant*. $n^{\circ}E$
 $nomC$ référence un *Cours*

Définition: l'étudiant $n^{\circ}E$ a réussi le cours $nomC$ telle année et a obtenu telle note

Relation : Inscrit

Attributs: $n^{\circ}E$: entier sans nul
 $nomC$: $Dnom$ sans nul

Identifiant: ($n^{\circ}E + nomC$)

Identifiants externes : $n^{\circ}E$ référence un *Etudiant*. $n^{\circ}E$
 $nomC$ référence un *Cours*

Définition: actuellement, l'étudiant $n^{\circ}E$ est inscrit au cours $nomC$

Relation : Prérequis

Attributs: $nomC$: $Dnom$ sans nul
 $nomCpré requis$: $Dnom$ sans nul

Identifiant: ($nomC + nomCpré requis$)

Identifiants externes : $nomC$ référence un *Cours*
 $nomCpré requis$ référence un *Cours*

Définition: le cours $nomCpré requis$ est un prérequis pour le cours $nomC$

Contrainte d'intégrité: dans tout tuple, $nomCpré requis$ doit être différent de $nomC$

Contraintes d'intégrité:

Pour tout tuple de *Prérequis* $\langle nomC, nomCpré requis \rangle$, le cycle de $nomCpré requis$ dans *Cours* doit être inférieur ou égal à celui de $nomC$.

Pour tout tuple de *Etudiant*: $année_en_cours - dateN \geq 18$

Pour tout tuple de *EtudiantEtudes* $\langle n^{\circ}E, année, diplôme \rangle$, soit $dateN$ la date de naissance de l'étudiant dans la relation *Etudiant*, alors: $dateN < année$

Pour tout tuple de *Obtenu* $\langle n^{\circ}E, nomC, note, année \rangle$, soit *dateN* la date de naissance de l'étudiant dans la relation *Etudiant*, alors: *dateN* $\langle année$

Pour tout tuple de *Inscrit*, $\langle n^{\circ}E, nomC \rangle$, le $n^{\circ}E$ doit exister dans *Obtenu* associé à tous les cours qui existent dans *Prérequis* associés à *nomC*.

NB : Les mots en italique sont définis par l'utilisateur, les autres sont des clauses du langage de définition de schéma.

2. Base de données relationnelle partielle FormaPerm

Personne

<u>n°P</u>	nom	adr
1111	Rochat	Lausanne
6666	Walter	Lausanne
5555	Bernard	Yverdon
2222	Rochat	Renens
3333	Muller	Morges

Etudiant

<u>n°P</u>	n°E	dateN
6666	22	3/3/78
1111	36	1/1/79
5555	44	2/2/78

Enseignant

<u>n°P</u>	tel	statut
2222	80111111	prof
3333	80222222	assistant

Cours

<u>nomC</u>	cycle	n°Ens
système	2	2222
BD	2	2222
SI	2	2222
algo	1	3333
C	1	3333

Obtenu

<u>n°E</u>	<u>nomC</u>	note	année
22	algo	10	1998
22	C	9	1998
44	algo	9	1999

Inscrit

<u>n°E</u>	<u>nomC</u>
36	algo
36	C
22	système
22	SI
44	C
44	BD

Prérequis

<u>nomC</u>	<u>nomCprérequis</u>
système	algo
système	C
BD	algo
SI	algo
SI	C

Exercices - Série 6

Traduire en schémas relationnels les diagrammes entité association suivants en utilisant l'algorithme de traduction proposé dans le chapitre suivant (chapitre 5, p. **Erreur ! Signet non défini.**). Préciser les identifiants et les identifiants externes des relations.

Exercice 1: Bibliothèque

(Chapitre 3 - Série 2 : exercice 1)

Exercice 2: Editeur

(Chapitre 3 - Série 2: exercice 2)

Exercice 3: Les courses de chevaux (solutions 1 et 2)

(Chapitre 3 - Série 2: exercice 3)

Exercice 4: Aéroport

(Chapitre 3 - Série 3: exercice 1)

Exercice 5: Personnel

(Chapitre 3 - Série 3: exercice 3)

Exercice 6: Généalogie (solutions 1 et 2)

(Chapitre 3 - Série 3: exercice 4)