

SGBD(R)

Système de Gestion de Bases de Données (Relationnels)

OLIVARÈS Georges

25/05/2016

© Groupe CGI inc. CONFIDENTIEL

CGI

La force de l'engagement[MD]

SGBDr
Modèles de données



CHAPITRE 2

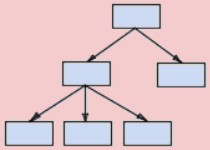
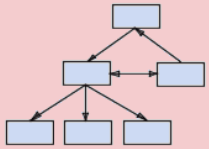
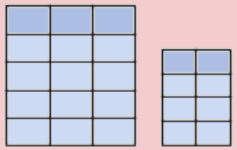
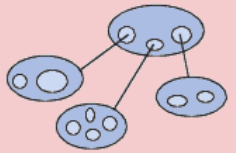
CGI

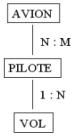
La force de l'engagement[MD]

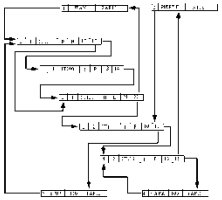
Plan – Chapitre 2 – Modèles de données

1. Les différents modèles
2. SGBDr - Règles d’intégrité
3. Concepts
4. Algèbre

Autres modèles

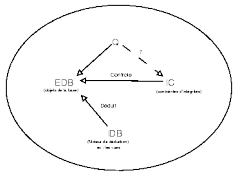
Hierarchique	Réseau	Déductif	Objet
			
Données classées hiérarchiquement. Selon une arbo. descendante Ce modèle utilise des pointeurs entre les différents enregistrements. Il s'agit du premier modèle de SGBD	Ce modèle utilise des pointeurs vers des enregistrements Toutefois la structure n'est plus forcément arborescente dans le sens descendant	Les données sont représentées sous forme de table, mais leur manipulation se fait par calcul de prédicats	Les données sont stockées sous forme d'objets, c'est-à-dire de structures appelées *classes* présentant des données membres. Les champs sont des instances de ces classes
+ Organisation à modèle économiques/ social + Simplicité du modèle + Implémentation facile + Besoin de l'utilisateur dans la structure	+ Représentation naturelle + Elimination des redondances + Chemin d'accès multiples + Pas d'anomalies de stockage	+ Modèle pour les données factuelles et règles de comportement et de déduction (BI, biologie, …)	+ Grande souplesse de stockage + Représentation d'entitées complexes
-Pas de liens N:M -> redondance -Anomalies lors des opérations (nœuds enfants, création=parent, modif pour toutes occurrences) - Chemin d'accès unique -Logique très réduite	- Pas d'indépendance vis à vis des stratégies d'accès.		-Moins de contrôles de données -N'offres pas toutes les possibilités des autres





```
RECORD NAME IS VINS;
02 NV TYPE IS SIGNED PACKED DECIMAL 5;
02 CRU TYPE IS CHARACTER 10;
02 MILLESIME OCCURS 5 TIMES;
03 ANNEE TYPE IS SIGNED UNPACKED DECIMAL 4;
03 DEGRE TYPE IS BINARY 15.

RECORD NAME IS PRODUCTEURS;
02 NP TYPE IS SIGNED PACKED DECIMAL 5;
02 NOM TYPE IS CHARACTER 10;
02 PRENOM TYPE IS CHARACTER 10;
02 ADRESSE
     03 RUE TYPE IS CHARACTER 30;
     03 CODE TYPE IS SIGNED PACKED DECIMAL 5;
     03 VILLE TYPE IS CHARACTER 10;
```



Le modèle relationnel

- Proposer des schémas de données faciles à utiliser
- Améliorer l'indépendance logique & physique
- Mettre à disposition des utilisateurs des langages haut niveau pouvant être utilisé par des non informaticiens
- Optimiser l'accès à la BDD
- Améliorer l'intégrité et la confidentialité
- Fournir une approche méthodologique dans la construction des schémas

Pour Codd

- Les données sont présentes dans des tables
- Des opérateurs permettent de manipuler ces tables
- Il y a des règles d'intégrité sur ces tables

CGI

Les règles d'intégrité (les 12)

- Règle 1 - Règle d'information
  - Représentation uniforme des données
    - Toutes les informations de la base de données doivent être représentées d'une seule et même manière, notamment par des valeurs dans les colonnes à l'intérieur de lignes
- Règle 2 - Règle de garantie d'accès
  - Chaque information stockée dans une base de données relationnelle doit pouvoir être accessible en précisant uniquement le nom de la table, le nom de la colonne et la valeur de la clé primaire
- Règle 3 - Règle des informations manquantes
  - Moyen d'expression des valeurs manquantes (*NULL*)
    - Il doit y avoir un moyen d'exprimer les valeurs manquantes dans la base de données d'une façon homogène non dépendante du type de donnée
    - Cette expression doit être supportée dans les opérations logiques
- Règle 4 - Catalogue système
  - Représentation dynamique de la description de la base
    - La description de la base au niveau logique doit être représentée dynamiquement par un ensemble de données normalisé de telle sorte que ce catalogue soit accessible aux utilisateurs autorisés depuis leur langage de requête

Les règles d'intégrité (les 12)

- Règle 5 - Règle de sous-langage
  - Mise à disposition d'un langage relationnel
    - Le système doit mettre à disposition au moins un langage relationnel qui :
      - a une syntaxe linéaire
      - peut être utilisé à la fois de manière interactive et dans les programmes d'application
      - supporte les opérations de définition de données (dont les vues), de manipulation de données (mise à jour et accès), les contraintes de sécurité et d'intégrité, les transactions (début, validation, annulation)

- Règle 6 - Règle de mise à jour des vues
  - Définition de vues et mises à jour des tables depuis les vues
    - Le système doit supporter la définition des vues
    - Les informations des tables sous-jacentes doivent pouvoir être mises à jour directement depuis la vue

- Règle 7 - Insertion, mise à jour et suppression
    - Le système doit permettre aux opérateurs d'insertion, de mise à jour et de suppression d'être actifs en même temps

- Règle 8 - Indépendance des données physiques
  - La modification de la structure physique de la base ne modifie pas les programmes
    - Les programmes d'application ou les opérations interactives ne doivent pas être modifiés si la structure physique de la base est modifiée



CGI

Les règles d'intégrité (les 12)

- Règle 9 - Indépendance des données logiques
  - Les programmes d'application et les opérations n'ont pas à être modifiés si la base de données est restructurée
    - au moins s'il n'y a pas de perte d'information
- Règle 10 - Indépendance par rapport aux contraintes d'intégrité
  - Les contraintes d'intégrité doivent être spécifiées séparément des programmes d'application et rangées dans le catalogue
  - Elles doivent pouvoir être modifiées sans devoir retoucher les applications existantes
- Règle 11 - Indépendance par rapport à la distribution des données
  - Les applications existantes se déroulent normalement quand
    - une version distribuée du SGBD est introduite pour la première fois
    - les données distribuées sont redistribuées sur le système
- Règle 12 - Règle de non-subversion
  - Si le système dispose d'une interface de bas niveau, elle n'a pas la possibilité de pervertir le système (passer outre une directive de sécurité ou une contrainte d'intégrité)



CGI

Concepts

Domaine: est un ensemble de valeurs (distinctes) caractérisées par un nom

Pays = (‘France’, ‘Italie’, ‘Japon’)

T°C = [-10; 20]

Fibonacci = {1, 2, 3, 5}

Relation: est un sous ensemble du produit cartésien appliqué à une liste finie de domaines, caractérisé par un nom.

DeviseCourante = {<France, Euro>, <Italie, Euro>, <Japon, Yen>}

On visualise une relation dans une table

Attribut: est une colonne de relation, caractérisé par un nom unique dans cette relation.

PaysDepart x PaysArrivé -> Pays x Pays

On visualise une relation dans une table

Tuple: est un n-uplet de valeurs

PaysDepart x PaysArrivé x Vol -> tuple est <France, Italie, B79865>

On visualise une relation dans une table

Concepts

Domaine ?

Relations ?

Attributs ?

Tuples ?

Identifiant	Prénom	Nom	Email
1	Jean	Dupont	jdupont@email.com
2	Marie	Malherbe	mama@email.com
3	Hadrien	Piroux	happi@email.com
4	Nicolas	Jacques	Jacques.nicolas@email.com

Afficher | Structure | SQL | Rechercher | Insérer | Exporte

#	Nom	Type	Interclassement	Attributs	Null	Défaut	Extra
1	**Identifiant**	int(11)			Non	*Aucune*	AUTO_INCREMENT
2	**Prénom**	varchar(50)	latin1_swedish_ci		Non	*Aucune*	
3	**Nom**	varchar(50)	latin1_swedish_ci		Non	*Aucune*	
4	**Email**	varchar(255)	latin1_swedish_ci		Non	*Aucune*	

Concepts

Le modèle relationnel n'apporte pas une réponse satisfaisante à tous les problèmes:

- Prendre en compte des « objets » structurés ainsi que les opérations qui leur sont associées
  - SGBD Orienté objet
- Prendre en compte des données peu structurées: textes, sons, images, …
  - SGBD Multimédia
- Faire le point avec l'intelligence artificielle afin de pouvoir déduire de nouvelles données à partir de celles existant déjà
  - SGBD déductives

Algèbre – Produit cartésien

$$R \times S = \{(x_1, \ldots, x_k, x_{k+1}, \ldots, x_{k+n}) \mid (x_1, \ldots, x_k) \in R \wedge (x_{k+1}, \ldots, x_{k+n}) \in S\}$$

- R et S de schémas quelconques
- Un même attribut doit être renommé si il apparait dans les deux tables

Relation r

A	B	C
a	b	c
d	a	f
c	b	a

Relation s

D	E	F
b	g	a
d	a	f

r x s

A	B	C	D	E	F
a	b	c	b	g	a
a	b	c	d	a	f
d	a	f	b	g	a
d	a	f	d	a	f
c	b	a	b	g	a
c	b	a	d	a	f

Produit cartésien des domaines
D_1 = { 0, 1 } et D_2 = { a, b, c }
D_1 x D_2 = { (0, a), (0, b), (0, c), (1, a), (1, b), (1, c) }

0	a
0	b
0	c
1	a
1	b
1	c

Produit cartésien des domaines
(Dupond, Durant, Martin), (Jean, Paul)

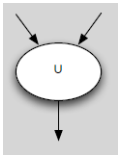
Dupond	Jean
Dupond	Paul
Durant	Jean
Durant	Paul
Martin	Jean
Martin	Paul

Algèbre – Union / Différence

Union

$$R \cup S = \{(x_1, \ldots, x_k) \mid (x_1, \ldots, x_k) \in R \vee (x_1, \ldots, x_k) \in S\}$$

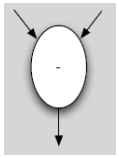
- R et S de schémas compatibles
- Opérateur ensembliste
- Permet de réaliser l'insertion de nouveaux tuples dans une relation



Différence

$$R - S = \{(x_1, \ldots, x_k) \mid (x_1, \ldots, x_k) \in R \wedge (x_1, \ldots, x_k) \notin S\}$$

- R et S de schémas compatibles
- Opérateur ensembliste
- Permet de réaliser la suppression de tuples dans une relation



Relation r

A	B	C
a	b	c
d	a	f
c	b	a

Relation s

D	E	F
b	g	a
d	a	f

r ∪ s

A	B	C
a	b	c
d	a	f
c	b	a
b	g	a

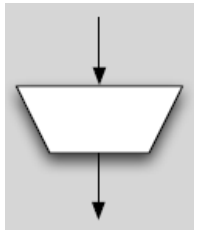
r – s

A	B	C
a	b	c
c	b	a

Algèbre – Projection

$$\pi_{A_{i_1},\cdots,A_{i_p}} : \begin{array}{ccc} R(A_1,\cdots,A_k) & \rightarrow & R'(A_{i_1},\cdots,A_{i_p}) \\ (x_1,\cdots,x_k) & \mapsto & R'(x_{i_1},\cdots,x_{i_p}) \end{array}$$

- R de schéma quelconque
- Permet de choisir un sous-ensemble des colonnes



Relation r

A	B	C
a	b	c
d	a	f
c	b	a

Relation s

D	E	F
b	g	a
d	a	f

$\prod_{AC}(r)$

A	C
a	c
d	f
c	a

Algèbre – Restriction

$$\sigma_{\varphi}(R) = \{(x_1, \ldots, x_k) \mid (x_1, \ldots, x_k) \in R \wedge \varphi(x_1, \ldots, x_k)\}$$

- R de schéma quelconque
- Permet de sélectionner des lignes, selon une formule logique
- Utilise des opérateurs :
  - logiques $\wedge$ (et), $\vee$ (ou), $\neg$ (non)
  - comparaison $\leq, <, =, >, \geq, \neq$

Relation r

A	B	C
a	b	c
d	a	f
c	b	a

Relation s

D	E	F
b	g	a
d	a	f

$\sigma_{B=b}(r)$

A	B	C
a	b	c
c	b	a

Algèbre – Intersection

$$R \cap S = \{(x_1, \ldots, x_k) \mid (x_1, \ldots, x_k) \in R \wedge (x_1, \ldots, x_k) \in S\}$$

$$R \cap S = R - (R - S)$$

- R et S de schéma identiques
- Permet de retrouver les tuples identiques

Relation r

A	B	C
a	b	c
d	a	f
c	b	a

Relation s

A	B	C
b	g	a
d	a	f

r ∩ s

A	B	C
d	a	f

Algèbre – Quotient

Soient R une relation de schéma $R(A_1, \cdots, A_k)$ et S de schéma $S(B_1, \cdots, B_p)$, telles que $\pi_{A_{k-p+1}, \cdots, A_k}(R)$ et S sont des relations de schémas compatibles.

R/S est le plus grand ensemble E tel que $E \times S \subseteq R$.

- R et S de schéma quelconques

Relation r

A	B	C	D
a	b	c	d
a	b	e	f
b	c	e	f
e	d	c	d
e	d	e	f
a	d	d	e

Relation s

C	D
c	d
e	f

r ÷ s

A	B
a	b
e	d
b	c

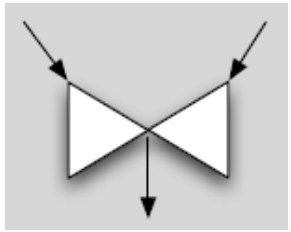
CGI

Algèbre – Jointure

$R \bowtie_\varphi S = \sigma_\varphi(R \times S)$

φ est une formule logique utilisant des opérateurs de comparaison $(=, \leq, <, >, \geq, \neq)$.

- **Équi-jointure:** égalité entre les deux colonnes
- **Jointure naturelle**: égalité sur les colonnes de même nom



Relation r

A	B	C
1	2	3
4	5	6
7	8	9

Relation s

D	E
3	1
6	2

$r \underset{B<D}{\bowtie} s$

A	B	C	D	E
1	2	3	3	1
1	2	3	6	2
4	5	6	6	2

Algèbre – Jointure naturelle

- **Jointure naturelle**: égalité sur les colonnes de même nom

$$r \bowtie s = \prod_{R \cup (S - R)} (\sigma (r \times s))$$
$$r.A_1 = s.A_1 \wedge$$
$$r.A_2 = s.A_2 \wedge$$
$$\ldots$$
$$r.A_k = s.A_k$$

Relation r

A	B	C
a	b	c
d	b	c
b	b	f
c	a	d

Relation s

B	C	D
b	c	d
b	c	e
a	d	b

r ⋈ s

A	B	C	D
a	b	c	d
a	b	c	e
d	b	c	d
d	b	c	e
c	a	d	b