

Bases de Données Relationnelles

talel@infres.enst.fr

Complément de cours et transparents sur
www.bd.enst.fr/bdl

Supports de cours : (1) *Database Management Systems*, R. Ramakrishnan and J. Gehrke, ed. McGrawHill, 2000.
(2) *Bases de Données*, G. Gardarin, ed. Eyrolles, 2001.

1

Plan

✍ Bases de données et SGBD ?

✍ Modèle relationnel

2

Base de Données ?

- ✍ Une collection de données cohérentes entre elles, généralement de taille importante.
- ✍ **Modélise** une *entreprise* du monde réel
 - Entités (ex., étudiants, cours)
 - Associations (ex., Marc a choisi BD)
- ✍ Un *Système de Gestion de Bases de Données (SGBD)* est un logiciel destiné au stockage et à la manipulation de bases de données.

3

Modèle de données

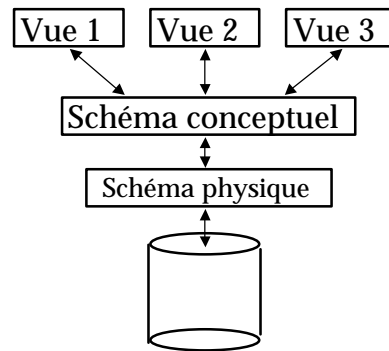
- ✍ Un *modèle de données* est un ensemble de concepts sur les données.
- ✍ Un *schéma* est une description d'un ensemble de données, s'appuyant sur un modèle de données.
- ✍ Le *modèle relationnel* est le plus répandu.
 - Concepts de base: *relation*, table avec tuples et des colonnes.
 - Chaque relation a un *schéma*, qui décrit ses colonnes.

4

Niveaux d'abstraction

☞ Plusieurs vues, un seul schéma conceptuel (logique) et schéma physique.

- Les vues décrivent comment l'utilisateur voit les données.
- Le schéma conceptuel définit la structure logique des données.
- Le schéma physique décrit les fichiers et les index utilisés.



☞ DDL : langage de définition des données; DML : langage de manipulation des données.

5

Exemple: Base de données d'une agence bancaire

☞ Schéma conceptuel :

- Clients(idl: integer, Nom: string, Prénom: string, Profession: string)
- Comptes(idc:string, type : string, solde:string, idl: integer)
- Opérations(idc:string, dte:date, type:string, montant : integer)

☞ Schéma physique :

- Les relations sont stockés dans des fichiers non séquentiels.

☞ Schéma externe (Vues):

- Cumul_mensuel(mois: string, idc:string, Nom: String, Prénom: string, total_crédits:integer, total_débits: integer)

6

Pourquoi un SGBD?

- ⌘ Indépendance des données/applications et sûreté d'accès aux données.
- ⌘ Temps de développement d'application réduit.
- ⌘ Intégrité des données et sécurité des accès.
- ⌘ Administration des données uniforme.
- ⌘ Concurrence des accès et reprise sur panne.

7

Indépendance des données

- ⌘ Les applications sont isolées des changements de structure et du mode de stockage des données.
- ⌘ Indépendance logique des données: Protection des changements de structure des données au niveau logique.
- ⌘ Indépendance physique des données: Protection des changements de structure au niveau physique.
- ⌘ *Un des plus importants bénéfices de l'utilisation des SGBD*

8

Contrôle de concurrence

- ⌘ L'exécution concurrente de programmes est essentielle pour un SGBD.
 - Les accès disque sont fréquents et relativement *lents*, il est important que l'unité centrale puisse exécuter de façon concurrente les programmes des utilisateurs.
- ⌘ L'exécution partielle des actions de différents programmes peut aboutir à des incohérences: ex., opération de débit en même temps que la réalisation d'un virement.
- ⌘ Les SGBD assurent que la concurrence soit réalisée sans problème: chaque utilisateur a l'impression d'être seul à travailler sur le système.

9

Exécution d'un programme au-dessus d'une BD

- ⌘ Concept clé : *transaction*, une séquence atomique d'actions sur une BD (lectures/écritures).
- ⌘ Chaque transaction est sensée laisser la BD dans un *état cohérent* après l'avoir prise dans un état cohérent.
 - Les utilisateurs peuvent spécifier des *contraintes d'intégrité* simples sur les données et le SGBD se charge de les garder inviolables.
 - En dehors de ça, le SGBD n'a pas conscience de la sémantique des traitement effectués.
 - Le fait qu'une transaction préserve la cohérence de la BD est au bout du compte de la responsabilité de l'utilisateur!

10

Principes de l'ordonnement des transactions

- ⌘ Les SGBD assurent une exécution concurrente de $\{T_1, \dots, T_n\}$ qui soit équivalente à une exécution *en série* des mêmes transactions.
- ⌘ Moyens :
 - Verrouillage des granules manipulés par les transactions.
 - ⌘ protocole **V2P strict** : avant de lire/écrire un granule, une transaction émet une demande de verrou et attend que le SGBD le lui accorde. Tous les verrous sont relâchés à la fin de la transaction.
 - ⌘ Possibilité de générer des situations de **Deadlock!**
 - Estampillage des transactions et des granules manipulés.
 - ⌘ Protocole d'estampillage : un TS est affecté à chaque transaction prise en compte par le système. Un TS en lecture et un TS en écriture sont gérés pour chaque granule.
 - ⌘ Protocole plus complexe, mais évite les deadlock.

11

Sûreté des traitements

- ⌘ Les SGBD assurent la cohérence des données même en cas de crash.
- ⌘ **Idée**: garder un *journal ou log* (historique) de toutes les actions élémentaires de m-à-j et de validation réalisées par le SGBD :
 - **Avant** qu'un changement ne soit réalisé, l'action est tracée dans un log file.
 - Après un crash, l'effet des transactions non abouties est annulé à l'aide du fichier log.

12

Les bases de données font «les affaires» de beaucoup de monde ...

- ✂ Les utilisateurs finaux et les éditeurs de SGBD
- ✂ Les développeurs d'applications BD

- ✂ Administrateurs de BD (DBA)

- conception logique / physique de schéma
- Gère la sécurité et les droits
- la disponibilité des données, reprise sur panne
- paramétrage (*tuning*) en fonction des besoins

Doit comprendre le fonctionnement des SGBD!

13

Résumé

- ✂ Les SGBD sont utilisés pour maintenir et interroger un volume de données important.
- ✂ Quelques bénéfices : reprise sur panne, accès concurrent, développement rapide d'applications, intégrité et sécurité des données.
- ✂ Les niveaux d'abstraction permettent l'indépendance des données.
- ✂ Les DBAs et les concepteurs de BD ont des jobs sensibles (de responsabilité) et sont **bien-payés!**
- ✂ La R&D en BD est très active (nouvelles techno, nouvelles applis, nouveaux modèles)

14

Plan

☞ Bases de données et SGBD ?

☞ Modèle relationnel

15

Concepts descriptifs

T. Codd (IBM San-Jose) en 1970

☞ **Domaine de valeurs** (ensemble de valeurs)

☞ Exemple : Entier, Réel, ... , Franc, Salaire = {5 000..100 000},
Point = {(X:Réel,Y:Réel)}

☞ Le **produit cartésien** $D_1 \times D_2 \times \dots \times D_n$ est l'ensemble des tuples (n-uplets) : $\langle V_1, V_2, \dots, V_n \rangle$ tel que $V_i \in D_i$

☞ Exemple : $D_1(\text{clients}) = \{\text{Dupond, Dupont, Tryphon}\}$,
 $D_2(\text{solde}) = \{100, 10000\}$

Dupond	100
Dupont	100
Tryphon	100
Dupond	10000
Dupont	10000
Tryphon	10000

16

Relation

- ✍ Sous-ensemble du produit cartésien d'une liste de domaines
- ✍ Chaque relation possède un nom
- ✍ Vision tabulaire
 - Une relation est une **table** à deux dimensions
 - Une ligne est un tuple
 - Un nom est associé à chaque colonne
- ✍ **Attribut** :
 - nom donné à une colonne d'une relation
 - prend ses valeurs dans un domaine

17

Exemple de relation

VINS	CRU	MILL	REGION	COULEUR
	CHENAS	1983	BEAUJOLAIS	ROUGE
	TOKAY	1980	ALSACE	BLANC
	TAVEL	1986	RHONE	ROSE
	CHABLIS	1986	BOURGOGNE	BLANC
	ST-EMILION	1987	BORDELAIS	ROUGE

18

Clé

- ⌘ Groupe d'attributs minimum d'une relation qui détermine chaque tuple de façon unique
 - Exemples : {Cru, Millésime} dans la table VINS, NSS pour des PERSONNES
- ⌘ **Clé primaire** : Clé choisie par le concepteur de la base de données
- ⌘ **Clé étrangère** : Groupe d'attributs d'une relation R_1 devant apparaître comme clé primaire dans une autre relation R_2
 - Les clés étrangères définissent les contraintes d'intégrité référentielles

19

Schéma

- ⌘ **Schéma d'une relation** : Nom de la relation, liste des attributs avec leurs domaines, (+Clé primaire)
- ⌘ **Exemple** :
 - VINS (NV : Entier, Cru : Texte, Mill : Entier, Degré : Réel, Région : Texte)
 - ⌘ Par convention, la clé primaire est soulignée
- ⌘ **Intention et extension**
 - Un schéma de relation définit l'intention de la relation
 - Une instance de table représente une extension de la relation
- ⌘ **Schéma d'une BD relationnelle**
 - C'est l'ensemble des schémas des relations composantes

20

Exemple de schéma

- ✍ **BUVEURS (NB, Nom, Prénom, Type)**
 - VINS (NV, Cru, Mill, Degré)**
 - ABUS (NB, NV, Date, Quantité)**
- ✍ clés étrangères
- ABUS.NV Référence VINS.NV**
 - ABUS.NB Référence BUVEURS.NB**

21

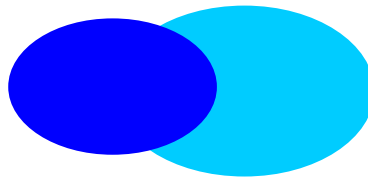
Résumé des concepts descriptifs

- ✍ Relation ou Table
- ✍ Attribut ou Colonne
- ✍ Domaine ou Type
- ✍ Clé
- ✍ Clé primaire
- ✍ Clé étrangère

22

Concepts de manipulation

- ⌘ Un ensemble d'opérations formelles (**algèbre relationnelle**), permettant d'exprimer les requêtes sous forme algébrique
 - Ces opérations sont utiles pour l'optimisation des requêtes
- ⌘ Opérations ensemblistes
 - UNION notée ?
 - INTERSECTION notée ?
 - DIFFERENCE notée –
- ⌘ Opérations algébriques :
 - Unaires : $O(\text{Relation}) \rightsquigarrow \text{Relation}$
 - Binaires : $\text{Relation} \times \text{Relation} \rightsquigarrow \text{Relation}$



23

Algèbre relationnelle

- ⌘ Cinq opérateurs de base :
 - Projection ?
 - Restriction ?
 - Produit cartésien \times
 - Deux opérateurs ensemblistes :
 - ⌘ Union ?
 - ⌘ Intersection ?
- ⌘ + opérateur de renommage ?
- ⌘ + opérateurs composés :
 - Jointure \bowtie
 - Division ?
 - Différence -

24

Projection

- ✍ Elimination des attributs non désirés et suppression des tuples en double
- ✍ Opérateur unaire, $\pi_{A_1, A_2, \dots, A_p}(R)$

VINS	Cru	Mill	Région	Qualité
	VOLNAY	1983	BOURGOGNE	A
	VOLNAY	1979	BOURGOGNE	B
	CHENAS	1983	BEAUJOLAIS	A
	JULIENAS	1986	BEAUJOLAIS	C

?
cru,Région

? (VINS)	Cru	Région
	VOLNAY	BOURGOGNE
	CHENAS	BEAUJOLAIS
	JULIENAS	BEAUJOLAIS

25

Restriction

- ✍ Obtention des tuples de R satisfaisant un critère Q
- ✍ Opérateur unaire, notée $\sigma_Q(R)$
- ✍ Q est le critère de qualification de la forme :
 $A_i \text{ ? } \text{Valeur}$
 $\text{?} \in \{=, <, >=, >, <=, ?\}$
- ✍ Il est possible de réaliser des "ou" (union) et des "et" (intersection) de critères simples

26

Exemple de restriction

VINS	Cru	Mill	Région	Qualité
	VOLNAY	1983	BOURGOGNE	A
	VOLNAY	1979	BOURGOGNE	B
	CHENAS	1983	BEAUJOLAIS	A
	JULIENAS	1986	BEAUJOLAIS	C

? MILL >= 1986

VINS	Cru	Mill	Région	Qualité
	JULIENAS	1986	BEAUJOLAIS	C

? CRU = "VOLNAY"

? CRU = "CHENAS"

27

Jointure \bowtie

- ⌘ Composition des deux relations sur un domaine commun
- ⌘ Relation binaire équivalente à une restriction sur un produit cartésien
 $R1 \bowtie_{?} R2$ équivalent à $?_{?}(R1 \times R2)$
- ⌘ Critère de jointure
 - Attributs de même nom égaux : Attribut = Attribut
 - ⌘ Jointure naturelle
 - Comparaison d'attributs : Attribut1 ? Attribut2
 - ⌘ Thêta-jointure

28

Exemple de jointure

VINS	Cru	Mill	Qualité
	VOLNAY	1983	A
	VOLNAY	1979	B
	CHABLIS	1983	A
	JULIENAS	1986	C



LOCALISATION	Cru	Région	QualMoy
	VOLNAY	Bourgogne	A
	CHABLIS	Bourgogne	A
	CHABLIS	Californie	B



VINSREG	Cru	Mill	Qualité	Région	QualMoy
	VOLNAY	1983	A	Bourgogne	A
	VOLNAY	1979	B	Bourgogne	A
	CHABLIS	1983	A	Bourgogne	A
	CHABLIS	1983	A	Californie	B

29

Résumé du modèle

- ✍ Un ensemble de concepts bien compris et bien formalisé
- ✍ Un modèle unique, normalisé et de plus en plus riche
- ✍ Un formalisme qui s'étend plutôt bien
 - algèbre d'objets
 - Algèbre pour le semi-structuré ?

30