

# Chapitre I

## Introduction à SQL

### I - Introduction à SQL

- 1.1 Introduction
- 1.2 Répertoires
- 1.3 Principes des BD relationnelles
- 1.4 Contraintes d'intégrité
- 1.5 Les composants SQL
- 1.6 Conclusions

### 1.1 Définitions

- **BD** = ensemble structuré d'informations accessibles simultanément par plusieurs utilisateurs
- **Bases de Données / Banques de Données**
- **Systèmes d'information**
- **SGBD** = Système de Gestion de Bases de Données
- « *La vérité est dans la base de données* »

### Les types abstraits de données

- **Alphanumérique**
  - entier, flottant, chaîne de caractères
- **Multimédia**
  - signal, image, dessin,
  - image animée, audiovisuel,
  - document

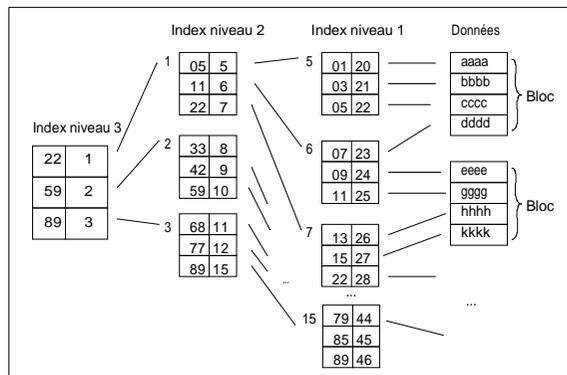
## 1.2 Répertoires

- Répertoire / index
- Accélérateur d'accès
- Sans index :
  - balayage séquentiel de toute la BD
  - très consommateur de temps
- Nécessité de structure de données adéquates et de procédures d'accès

## Indexation

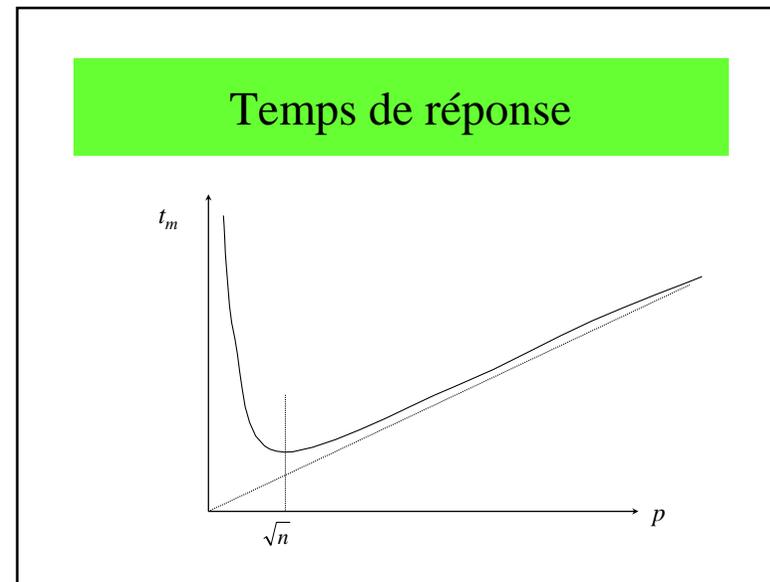
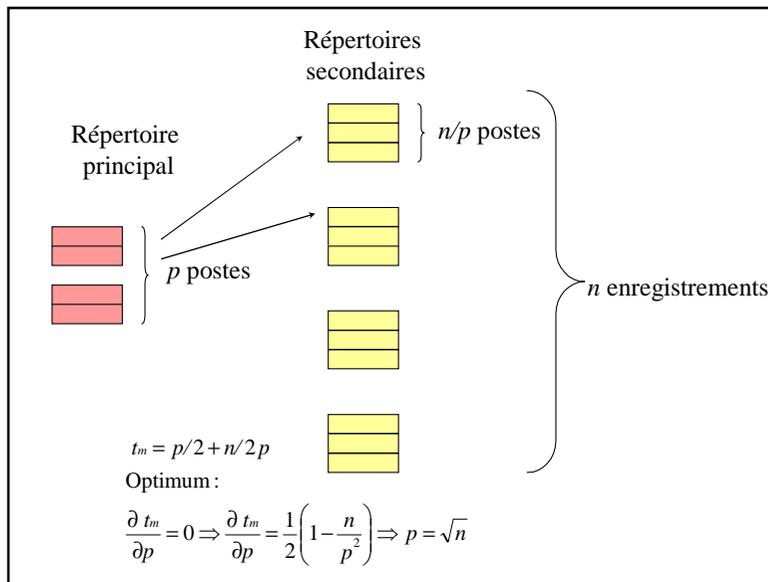


## Hierarchie de répertoires



## Temps de réponse

- Supposons  $n$  enregistrements
- sans index (balayage séquentiel)
  - $t_m = n/2$
- sans index (balayage dichotomique)
  - $t_m = \log_2(n)$



### 1.3 Principes des BD relationnelles

- Créé par Edgar CODD fin des années 60
- « Tout est table »
- 1 table == 1 relation
- Opérateurs sur des tables
- ORACLE, ACCESS, SYBASE, MySQL, etc.

### Tables et relations

*VEHICULE (Nom-Propriétaire, Marque, Numéro)*

<i>VEHICULE</i>	<i>Nom-Propriétaire</i>	<i>Marque</i>	<i>Numéro</i>
	Pierre	Renault	1111AA69
	Anne	Ford	2222BB69
	Marie	VW	3333CC69
	Jules	Fiat	4444DD69

## Définitions

TABLE

Colonne

Ligne  
n-uplet  
t-uple  
tuple



## Colonne

- Correspondant à un attribut
- Parfois clé ou identificateur
- Parfois valeur inconnue (*NULL*) Définie selon un domaine (entier, chaîne, image, etc.)
  - Dom (entier) =  $N \cup NULL$
- "*Order is immaterial*"

## Valeur NULL

- NULL = non renseigné
- valeur inconnue
- valeur non encore connue, non encore rentrée
- valeur impossible
- valeur inapplicable

## Lignes

- Correspond à un enregistrement / article
- Possède tous les attributs
- Identifiée par une clé
- "*Order is immaterial*"

## Clés primaires (primary keys)

- s'agit d'un identifiant unique à valeur obligatoire
- **attention** : dans une table, une clé unique peut être créée à partir de plusieurs attributs, ou plusieurs identifiants
- permet l'accès à une ligne d'une table

## Clés étrangères (foreign keys)

- Se dit d'une clé qui est référencée dans une autre table
- Exemple

PERSONNE (NOID, NOM, NOTEL)

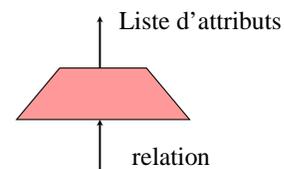
TELEPHONE (NOTEL, LIEU)



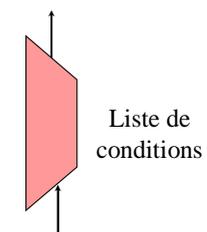
## Symbolisation des opérateurs

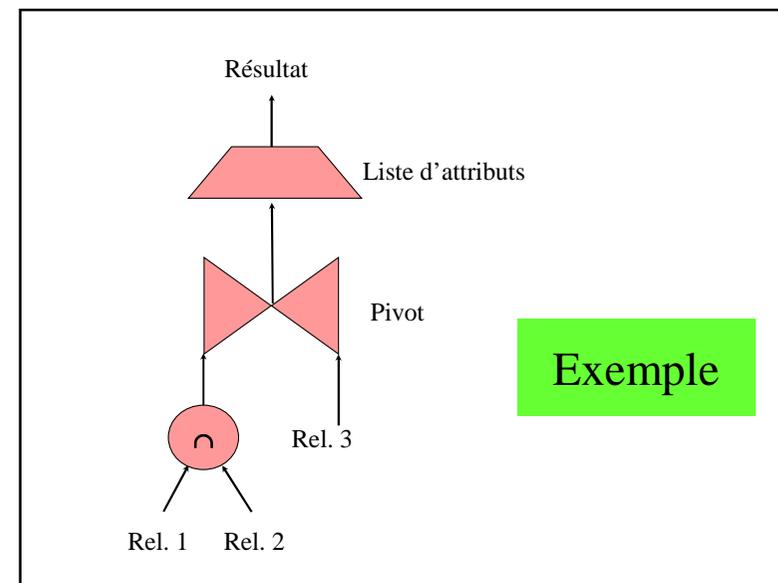
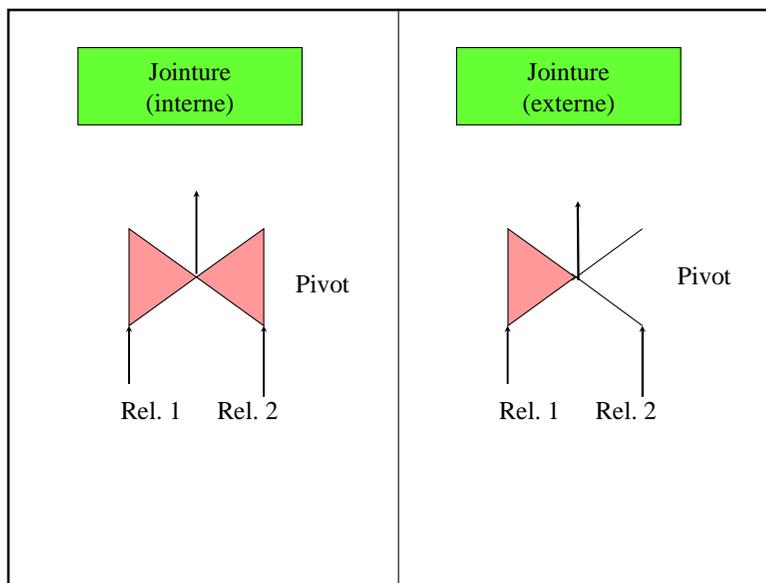
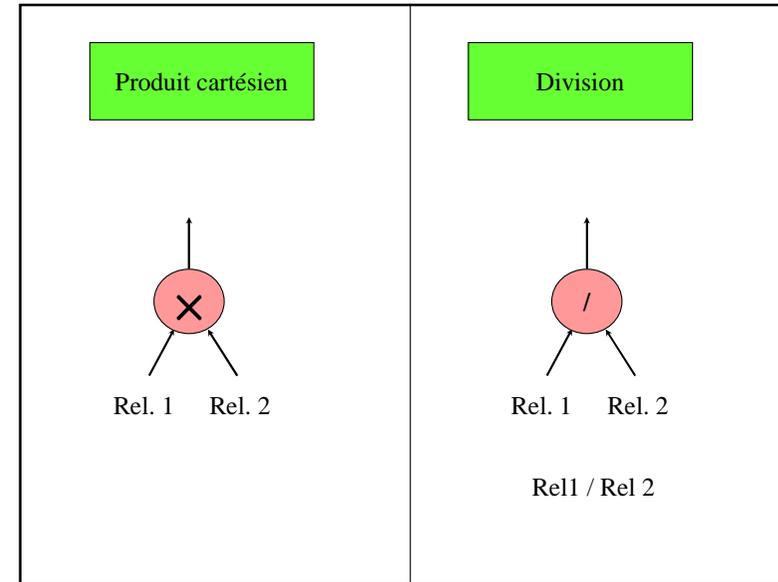
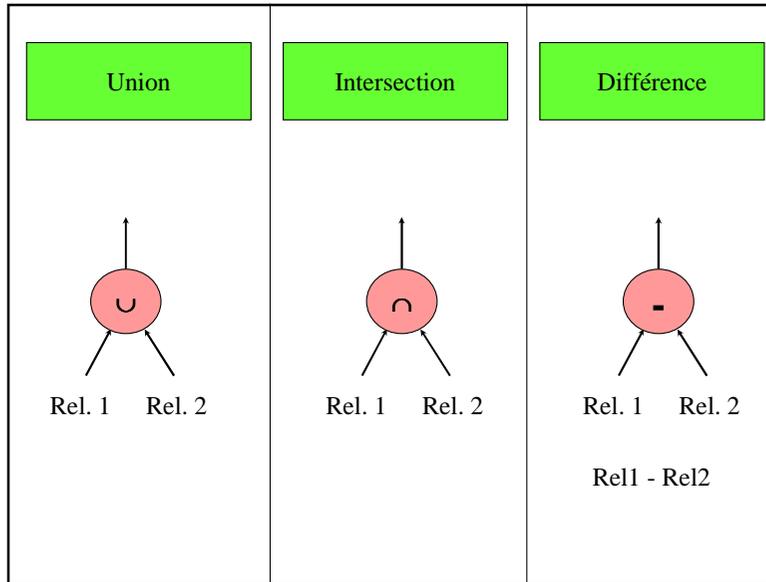
- Objectifs : graphe d'enchaînement des opérateurs afin de répondre à une requête

Projection



Restriction





## Remarques sur les jointures

- Pivot de la forme :
  - Relation1.Toto = Relation2.Titi
  - Autojointure possible
- Attention aux valeurs NULL dans les jointures internes
- Jointure externe : les attributs manquants sont forcés à NULL

## 1.4 Contraintes d'intégrité

- Contrôle de qualité du contenu
- Conditions ou prédicat
- Contraintes d'intégrité de base
  - existentielle (primary key)
  - référentielle (foreign key)
- Autres contraintes
  - contrôle de vraisemblance
  - CI topologiques

## Exemple

- Ensembles d'objets appelés « triangles »
  - 3 côtés
  - 3 angles
  - etc.

## Contraintes d'intégrité

- CI statiques :
  - $S \subseteq BD$
  - $C(S) = true \forall t$
- CI dynamiques : avant et après MAJ
  - $S(t^-) \subseteq BD(t^-)$
  - $S(t^+) \subseteq BD(t^+)$
  - $C(S, t^-) = true$
  - $C(S, t^+) = true$

## Remarques 1

- Si le prédicat est faux, il existe une incohérence qu'il va falloir corriger.
  - Etat stable / instable de la BD
- Plusieurs attitudes en cas d'incohérence
  - blocage temporaire de la BD : aucun traitement possible tant que l'incohérence subsiste
    - cas des CI longues
  - certains traitements sont malgré tout possibles durant un faible temps
    - corrections dites « au burin »

## Remarques 2

- Aucune contradiction entre les CI
- Nécessité d'outils de vérification de la monotonie des CI après intégration

## Vérification des CI

- Déclarative
  - à la création de la BD, on donne une liste de prédicats qui doivent toujours être satisfaits
  - vérification à la création et en mise à jour
- Procédurale
  - programmes de vérification
    - lancés régulièrement (procédures de contrôles)
    - ou sur événements (triggers).

## Contrainte existentielle

- Liée aux *primary keys*
- Détails de la contrainte
  - toutes les valeurs doivent être différentes
  - les valeurs « NULL » non autorisées
- En cas de difficultés, créer un identificateur provisoire
- Vérification déclarative (création)
- Vérifiée à chaque insertion et MAJ de clé

## Contrainte référentielle

- Liée aux *foreign keys*
- Détails de la contrainte
  - doit toujours avoir une contrepartie de type *primary key*
  - les NULL peuvent être tolérés
- Vérification déclarative (création)
- Vérifiée à chaque insertion et MAJ

## Contrôle de vraisemblance

- Vérification des ordres de grandeurs
- Par exemple : dates, âges, codes postaux, immatriculations de voitures, etc
- Petites vérifications : déclaratives
- Grosses vérifications : procédure/trigger

## Contrôle topologique

- Dans le cas des BD géographiques (vecteur)
  - polygones non dégénérés
  - deux parcelles ne doivent pas se chevaucher
  - surface totalement couverte
  - etc.
- Généralement par procédures ou triggers

## Ordonnement des ordres de création des tables

- 1 - sans ALTER
  - Commencer par les tables sans clés (rares)
  - Puis les tables avec PRIMARY KEYS
  - Puis terminer par les tables avec FOREIGN KEYS
- 2 - avec ALTER
  - rentrer dans n'importe quel ordre
  - puis introduire les clés (avec l'ordonnement précédent)

## 1.5 Les composants SQL

- Attention bien distinguer :
- **Les composants du système SQL** (ex schéma, index, cluster, etc.)
- **Les structures de composants logés dans la BD** (au sens object-oriented)

## Composants de SQL

- Composants schéma
- Composants globaux
- Partitions

## Composants Schéma

- Tables
- Contraintes
- Index
- Vues
- Types
- Séquence
- Procédures
- Fonctions
- Triggers
- Packages
- Synonymes
- Clusters
- etc.

## Composants globaux

- Contexte
- Utilisateurs
- Profils
- Rôles
- Tablespace
- Segments de rollback

## Partitions

- Sur les tables, sur les vues matérialisées
- Index partitionnés sur les tables

## Références à des parties de composants

- Schéma
- Partition
- BD distribuées

## Table DUAL

- Table fictive permettant d'assurer la régularité de la grammaire
- Ex :
  - `SELECT SYSDATE FROM DUAL;`

## 1.6 Conclusions

- Importance des BD et des SI
- Importance du modèle relationnel
- Petites et grosses BD
- Standardisation
- Importance des BD relationnelles
- Simplicité du modèle en tables
- Requête effectuée par combinaison d'opérateurs relationnels