

Chapitre 1

Introduction aux BD relationnelles

I.	INTRODUCTION.....	1
A.	BASE DE DONNEES	1
B.	SYSTEMES DE GESTION DE BASES DE DONNEES	1
C.	SGBD ET NORME ANSI/SPARC	3
D.	IMPLANTATION PHYSIQUE DES DONNEES (NIVEAU INTERNE).....	4
1.	Pages.....	4
2.	Index.....	4
3.	Index sous forme de B+ arbre	5
E.	LE LANGAGE SQL.....	6

I. Introduction

A. Base de données

Une **BASE DE DONNEES** (*anglais : database*) est un ensemble de données structurées, généralement stocké sur un support de persistance (*disque dur*).

Une **BASE DE DONNEES RELATIONNELLE** (*anglais : relational database*) est une base de données organisée sous forme de tables (colonnes et lignes) mises en relation.

Les bases de données ont comme support de stockage un ou plusieurs fichiers. Ces fichiers sont organisés spécifiquement selon le SGBD.

B. Systèmes de gestion de bases de données

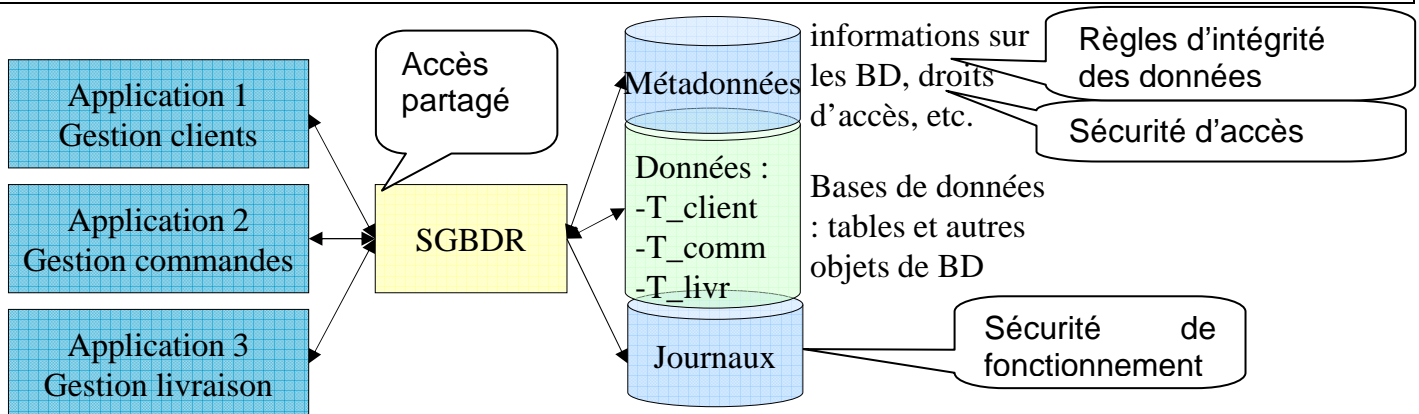
Un **SGBD**, Système de Gestion de Bases de Données, (*anglais : DataBase Management System*) est un ensemble de programmes qui permet la gestion du contenu d'une base de données, de manière

- sécurisée : l'accès aux données est contrôlé ; l'accès est géré par utilisateur ou groupe d'utilisateurs auxquels des droits d'accès spécifiques peuvent être attribués ;
- partagée : l'accès simultané aux données par plusieurs utilisateurs est permis : des mécanismes de contrôle sont mis en œuvre afin de garantir la cohérence des données accédées par chacun des utilisateurs) ;
- cohérente : les données sont contrôlées en utilisant divers mécanismes comme le contrôle de valeurs en double dans une colonne, le contrôle qu'une donnée d'une table fait bien référence à une données identique dans une autre table, etc. ;
- fiable : la garantie de stabilité des données est assurée, même après incident ; des systèmes de journalisation de toutes les modifications permettent le retour à un état antérieur cohérent de la base de données ;
- performante : l'accès aux informations – mémorisation et lecture – est réalisé en un temps optimal grâce à des structures de données performantes pour stocker les données, ainsi que des optimisations réalisées sur les requêtes des utilisateurs ;

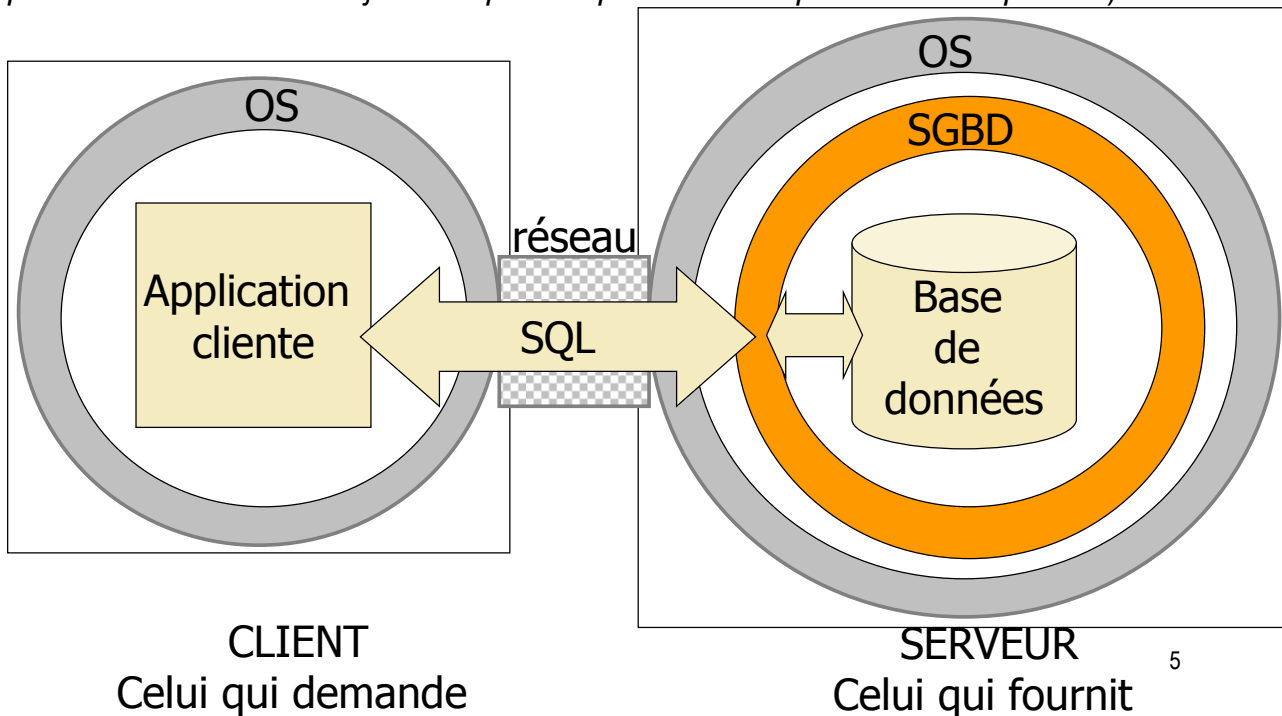
Langage SQL

- assure une indépendance des données au regard des applications et des utilisateurs qui y accède (indépendance logique) et des systèmes qui les hébergent (indépendance physique vis-à-vis des systèmes de gestion de fichiers)
- offre, grâce à un langage standard, la possibilité de décrire tous les objets d'une base de données et définir les liens existant entre eux, d'ajouter, modifier et supprimer des données, d'accéder aux données stockées

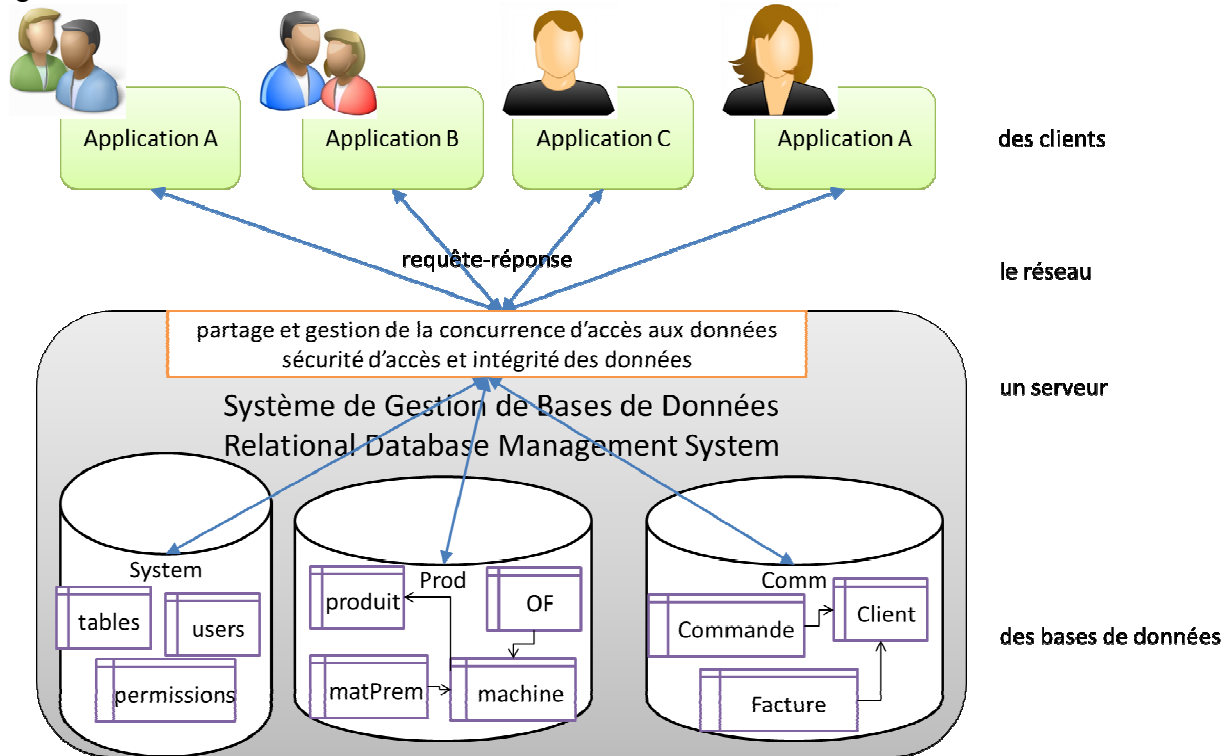
Les **SGBD Relationnels** (*anglais : Relational DBMS*) sont les SGBD des bases de données relationnelles.



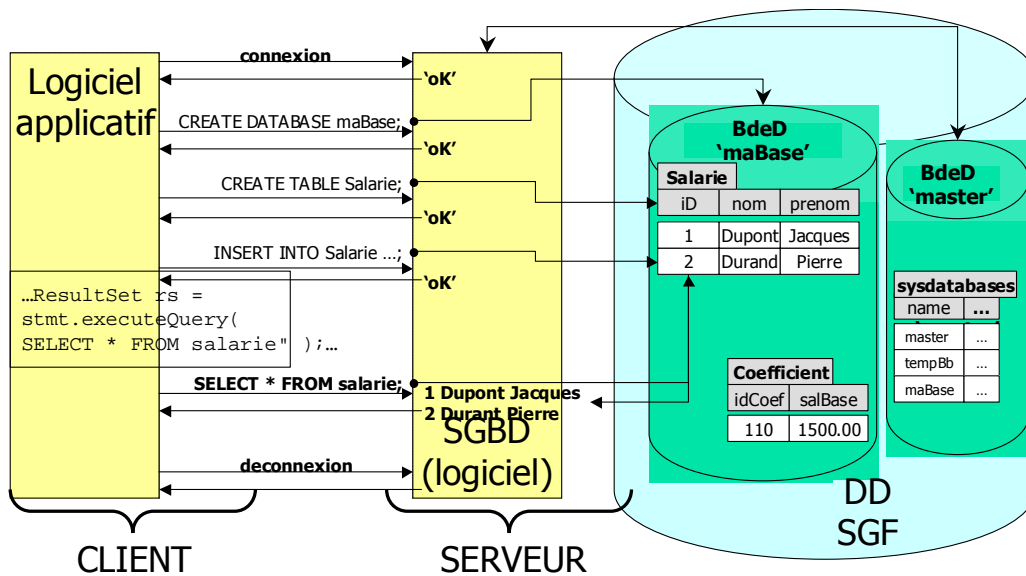
Les interactions entre un logiciel applicatif (client, demandeur) et le SGBD (fournisseur, serveur) fonctionnent selon un dialogue défini par une forme de protocole (*indépendamment des protocoles réseaux sous-jacents qui transportent les requêtes et les réponses*) :



Langage SQL



Le dialogue instauré pourrait être de cette forme (exemple basé sur Microsoft SQL Server 7) :

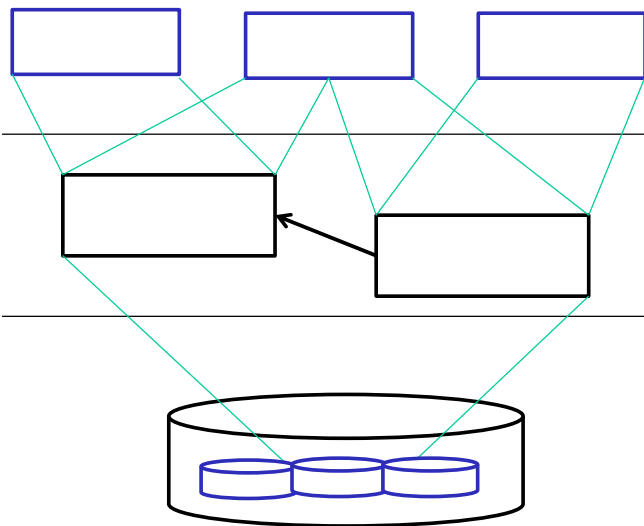


C. SGBD et norme ANSI/SPARC

La norme ANSI/SPARC¹ définit l'architecture fondamentale sur laquelle reposent les SGBD, en 3 couches (ou niveaux) :

¹ Groupe de travail mis en œuvre en 1975 pour définir l'organisation des données des BD : American National Standards Institute's Standards Planning and Requirements Committee (ANSI/SPARC)

Langage SQL



Niveau **externe** : vues utilisateurs d'un sous-ensemble du schéma de la BD (vues)

Niveau **conceptuel** (ou logique) : manière d'organiser des données et les liens entre elles (schéma de bases de données : tables, contraintes, index)

Niveau **interne** (ou physique) : manière d'implémenter les bases de données sur les supports physiques (fichiers)

Chacun des niveaux est indépendant des autres. Ainsi l'administrateur de la base de données peut-il modifier

- le stockage des données (sur plusieurs disques par exemple) sans que cela n'influe sur la représentation conceptuelle des données.
- le schéma logique (ajout de tables, par exemple) sans que cela n'influe sur les vues qu'ont les utilisateurs sur la base de données.

Les utilisateurs finaux peuvent accéder aux données en utilisant les vues sans se préoccuper ni du schéma conceptuel de la base ni de son implémentation physique.

D. Implantation physique des données (niveau interne)

Le stockage des données est essentiellement lié au système d'exploitation sur lequel le SGBD est installé. Les administrateurs de bases de données (*anglais* : *DBA, DataBase Administrator*) peuvent définir certains paramètres comme le choix de placement de fichiers de bases de données des unités de disques, etc.

Les unités de disques durs sont organisées logiquement en plateau et cylindre, format des pistes, découpées en secteurs angulaires. Le secteur est l'unité de stockage physique (un secteur a une capacité théorique de 512 octets).

Les systèmes d'exploitation, afin d'optimiser l'accès aux disques, définissent la notion de bloc composé d'un certain nombre de secteurs. Ainsi, sous Windows, un bloc regroupe 8 secteurs et a une capacité théorique de 4096 octets, 4kio.

1. Pages

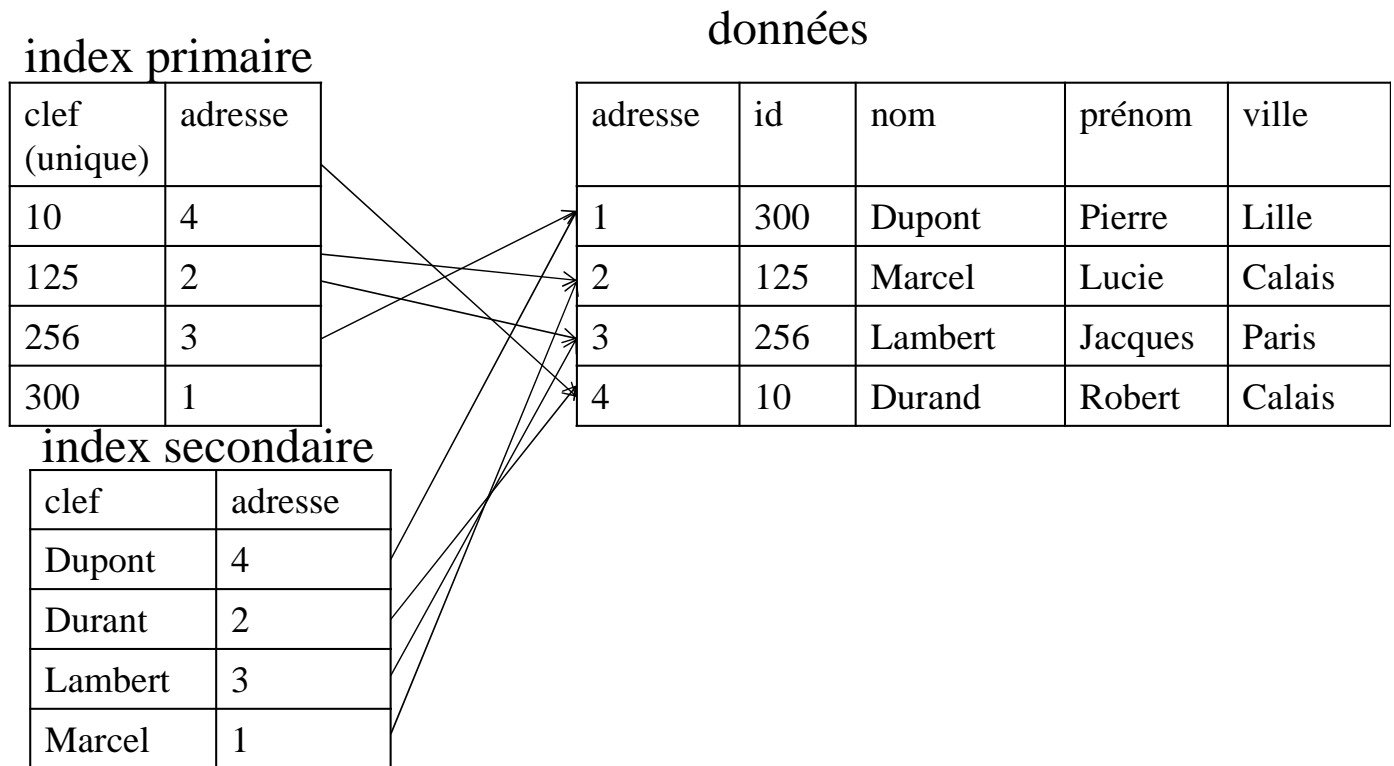
Afin de garantir l'indépendance vis-à-vis des unités de stockage, les SGBD définissent la notion de page : par exemple, sous MySQL, la taille des pages est par défaut de 16 ko. La structure des pages est spécifique à chaque SGBD.

La page est l'unité de gestion des entrées/sorties du SGBD. Une page permet le stockage de données d'une table ou d'une partie d'un index

2. Index

Au-delà du stockage des données, des mécanismes d'optimisation du placement des données au sein des fichiers ont été élaborées afin de garantir un temps d'accès optimal : ce sont les index.

Un index comporte généralement un ensemble des clefs (clefs primaires ou secondaires) faisant référence à l'ensemble des valeurs qui lui correspondent.



A chaque nouvelle insertion de ligne :

- les données sont ajoutées généralement à la fin de l'espace des données : une adresse est donc attribuée
- le couple (clef – adresse) est ajouté à la zone d'index, au bon endroit afin de maintenir l'ordre croissant des clefs, les clefs suivants étant décalées vers le bas.

Le temps de recherche d'une clef dépend du nombre N de clef :

- dans le cas d'une recherche séquentielle, sur une valeur de clef approximative, le nombre de clefs consultées sera en moyenne de $N/2$ (au pire N) ; ainsi pour 1000 clefs, 500 comparaisons
- dans le cas d'une recherche dichotomique, sur une valeur de clef exacte, sur une séquence de clefs triées, le nombre moyen de clefs consultées sera de $\log_2 N^{(2)}$; ainsi pour 1000 clefs, 10 comparaisons ($2^{10} = 1024$)

L'index est un mécanisme de recherche efficace. Cependant l'insertion d'une nouvelle clef peut nécessiter la réorganisation de l'ensemble de l'index stocké sur plusieurs pages.

3. Index sous forme de B³+ arbre

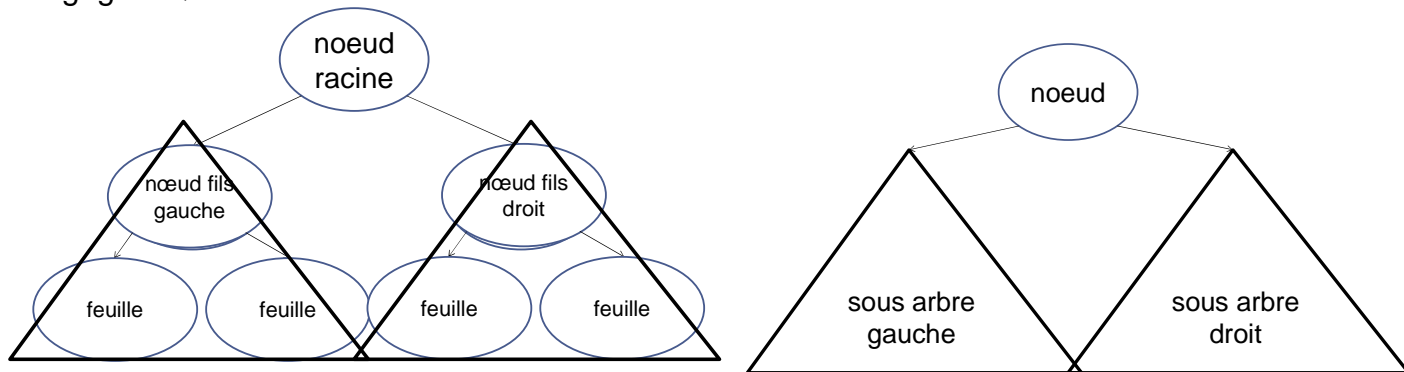
L'arbre est une structure de donnée très souple. Il est composé d'un ensemble de nœuds

- un nœud racine, point de départ de la recherche
- des nœuds internes qui définissent la hiérarchie de recherche
- des nœuds feuilles qui contiennent les données associées à une clef

² Le logarithme d'une base B d'un nombre X ($\log_B X$) est la puissance à laquelle il faut élever B pour obtenir X : $\log_{10} 1000$ vaut 3, car X vaut B^3 , soit 1000 vaut 10^3

³ B pour Balanced, c'est-à-dire équilibré

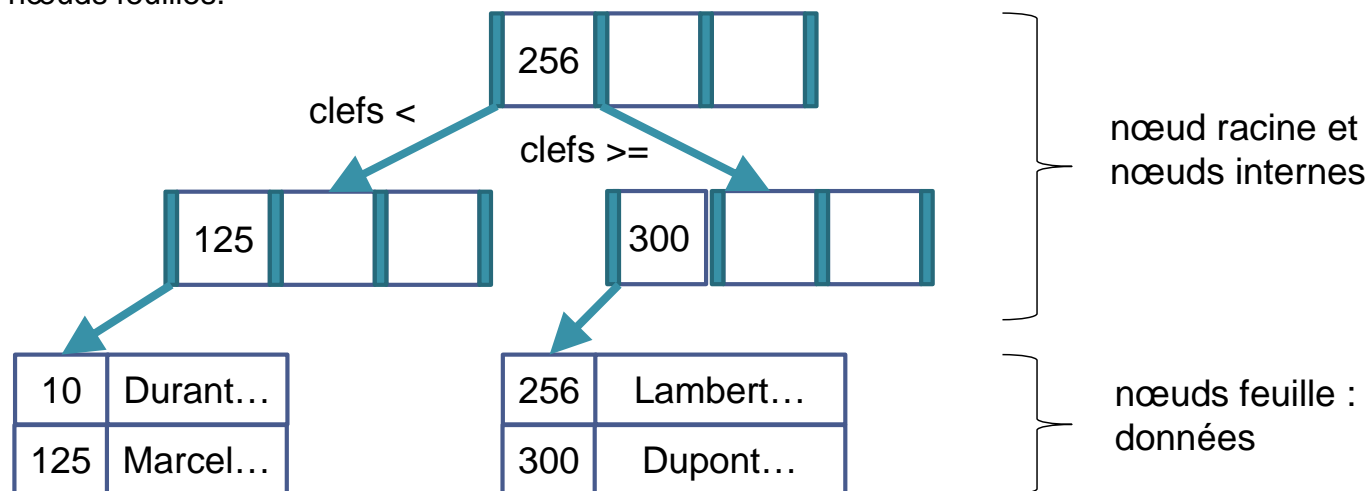
Langage SQL



La notion de hauteur définit, pour chacun des sous arbres gauche et droit d'un nœud, la distance maximale qui sépare le nœud de sa feuille la plus éloignée. Un B arbre doit être équilibré : la différence de hauteur de chacun des sous arbres d'un nœud ne doit pas dépasser 1 (*valeur absolue (hauteur sous arbre gauche – hauteur de sous arbre droit) <= 1*).

Dans un B+ arbre, chaque nœud peut contenir un nombre maximal de clefs (ici 3, qui définissent 4 intervalles de valeurs).

L'exemple qui suit montre l'organisation, à la partir du nœud racine, des nœuds internes et des nœuds feuilles.



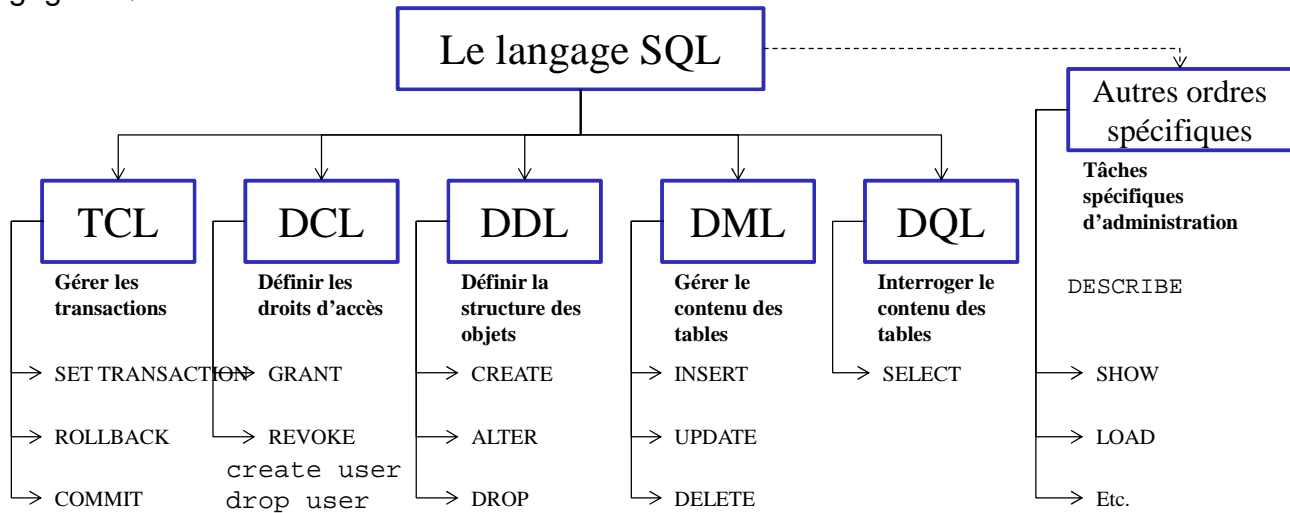
E. Le langage SQL

SQL (Structured Query Language) est LE langage permettant de donner des ordres aux SGBDR afin de créer les tables et de gérer leur contenu, mais aussi d'assurer la sécurité d'accès aux tables et le maintien de la cohérence des données.

Le langage SQL s'appuie sur les opérateurs de l'algèbre relationnelle définis en 1970 par Ted Codd, mathématicien, chercheur chez IBM. Le langage SQL est basé sur le concept de relation de la théorie des ensembles (l'équivalent des relations étant les tables pour SQL).

Le langage SQL a subi plusieurs évolutions : la version la plus utilisée actuellement est SQL92 (norme ANSI X3.135-1992 – www.ansi.org).

Il permet de faire abstraction de l'organisation physique des données.



Le langage SQL permet de couvrir tous les besoins d'accès aux bases de données relationnelles grâce à 5 groupes d'ordres SQL :

- **Data Definition Language (DDL)**, le Langage de Définition des Données (LDD), permet la gestion des tables de la base de données (création, ajouter des colonnes, supprimer des colonnes, etc.)
- **Data Control Language (DCL)**, le Langage de Contrôle des Données (LCD), permet la gestion des droits d'accès aux tables par utilisateur
- **Transaction Control Language (TCL)**, le Langage de Contrôle des Transactions, permet la gestion de l'intégrité des données dans la base de données
- **Data Manipulation Language (DML)**, le Langage de Manipulation des Données (LMD) donne accès au contenu des tables : ajouter des lignes, modifier des valeurs de colonnes, supprimer des lignes.
- **Data Query Language (DQL)**, le Langage d'Interrogation des Données (LID) permet de construire des requêtes d'interrogation des données dans les tables.

D'autres ordres spécifiques peuvent compléter ce jeu pour réaliser des tâches généralement liées à l'administration de la base, en utilisant le système de requêtes classique.

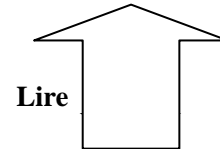
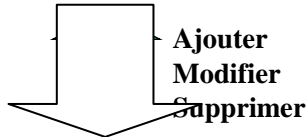
Parmi ces groupes, 3 sont plus directement liées aux tables :

DDL = définir la STRUCTURE

numero	nom	prenom	ville	salaire	entree
---------------	------------	---------------	--------------	----------------	---------------

DML = gérer le CONTENU

DQL = interroger le CONTENU



numero	nom	prenom	ville	salaire	entree
1	dupont	max	arras	1000	01/01/07
2	durant	tim	aix	1500	15/03/07
3	lambert	betty	pau	1350	20/04/07
4	bradford	jean	arras	1250	30/09/07

nom	prenom
dupont	max
bradford	jean