



Département Génie des systèmes
Filière IMSI : 4^{ème} année ingénieur

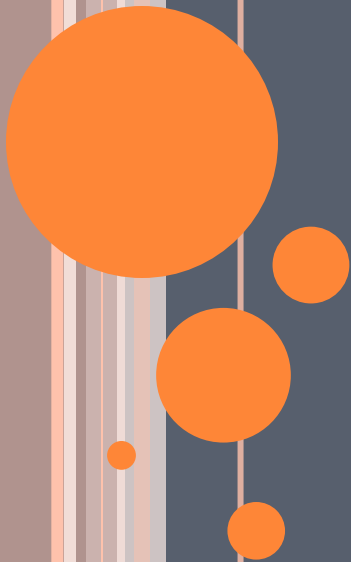
ARCHITECTURE ET ADMINISTRATION DES BASES DE DONNÉES

DR F.KABLI
Maître Conférence B

**[kablifatima47@
gmail.com](mailto:kablifatima47@gmail.com)**

Chapitre 1 :

BDD ET SGBD



BREF HISTOIRE.....

- ❖ Les disques durs, mémoire de masse de grande capacité, ont été inventés en **1956**.
- ❖ Les premières bases de données hiérarchiques sont apparues au début des années **1960**.
- ❖ En **1969**, base de données réseau au sein du **CODASYL**.
- ❖ La thèse de « F.Codden, **1970** » est à l'origine des bases de données relationnelles.
- ❖ **En 1990**, le début des bases de données objet.



BASES DE DONNÉES ?

- ❖ « Une base de données est un ensemble d'informations archivées dans les mémoires accessibles à des ordinateurs en vue de permettre le traitement des divers applications »
jacques chaumier , les banques de données, PUF 1984.
- ❖ Le concept de base de données permet de stocker et d'organiser une grande quantité d'information.
- ❖ Une base de données est une collection d'informations organisées afin d'être facilement consultables, gérables et mises à jour.



LES TYPES DE BASES DE DONNÉES

- 1. Les bases de données fichiers**
- 1. Les bases de données hiérarchiques**
- 1. Les bases de données réseau**
- 1. Les bases de données relationnelles**
- 1. Les bases de données objet (Exemple : orienté graphe)**



UTILISATIONS DES BDD:

- ❖ Compagnies aériennes pour gérer les réservations.
- ❖ Gestion de production.
- ❖ Enregistrements médicaux dans les hôpitaux.
- ❖ Enregistrements légaux dans les compagnies d'assurances.
- ❖ La gestion des grandes entreprises ou les universités.
- ❖ Etc.....

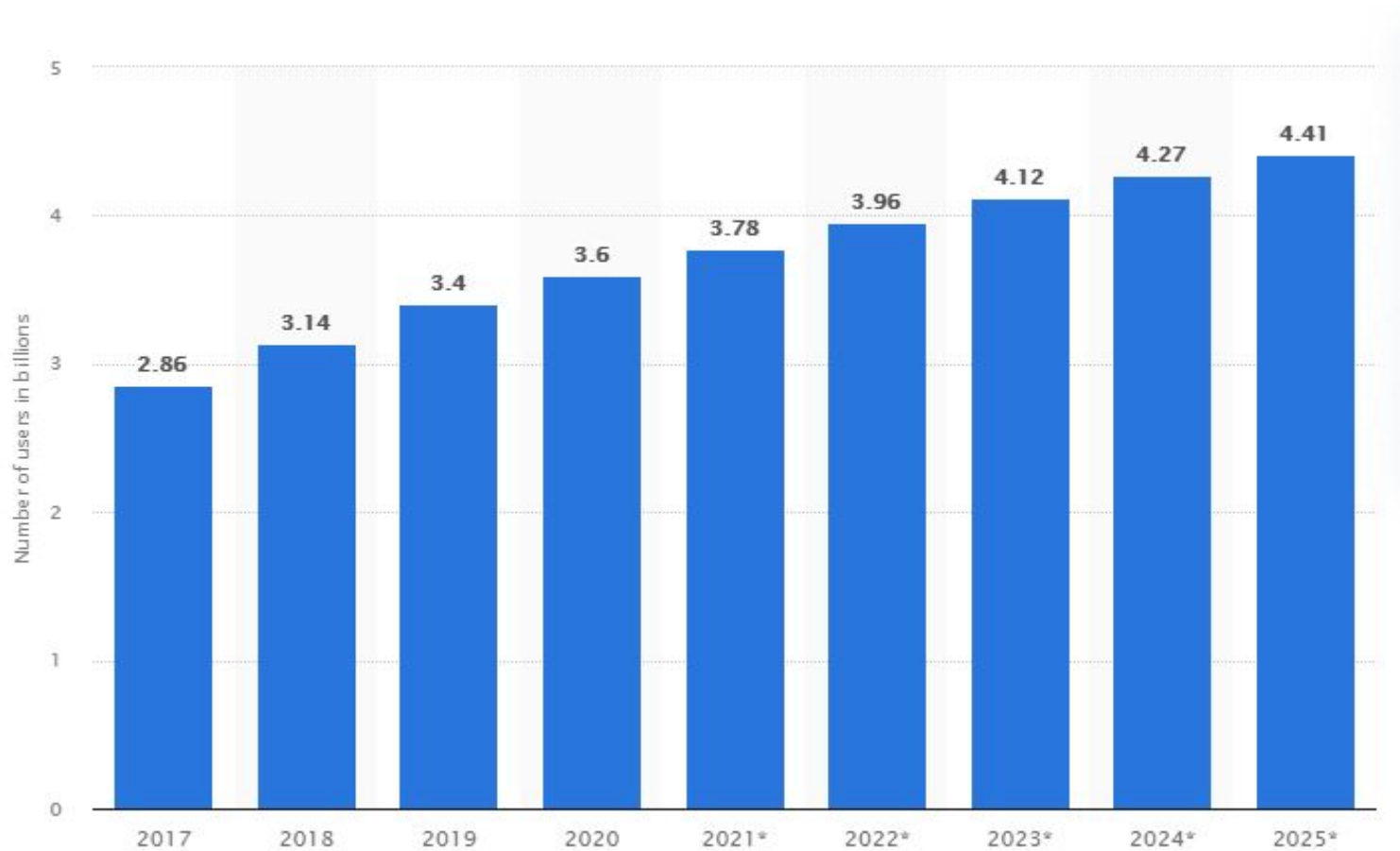


POURQUOI LES BASES DE DONNÉES ?

COMMENT GÉRER LES BASES DE DONNÉES ?



NOMBRE DES UTILISATEURS DES RS



POURQUOI LES BASES DE DONNÉES ?

COMMENT GÉRER LES BASES DE DONNÉES ?

❖ Base de données de **youtube** :

- ✓ 2007, 45 téraoctets (doublé tout les cinq mois).
- ✓ 2016, 100 heurs de vidéo mise en ligne chaque munit, en fixant 100 MO pour une heur, donc $100 * 100 = 10000 = 10GO$ par minute alors, **14,5 TO / Jour.**
- ✓ **2019** : 2 milliards d'utilisateurs actifs par mois, 300 heures de vidéo sont téléchargées sur YouTube chaque minute Près de 5 milliards de vidéos sont visionnées sur youtube chaque jour.
- ✓ **2021**: 2,3 milliards d'utilisateurs actifs par mois



POURQUOI LES BASES DE DONNÉES ?

COMMENT GÉRER LES BASES DE DONNÉES ?

❖ Base de données « **Réseaux sociaux** »:

✓ **Facebook :**

2019 : 300 petabytes par mois.

Généré 4 nouveau petabytes par jour

2021 : 2,8 milliards utilisateurs actifs par mois

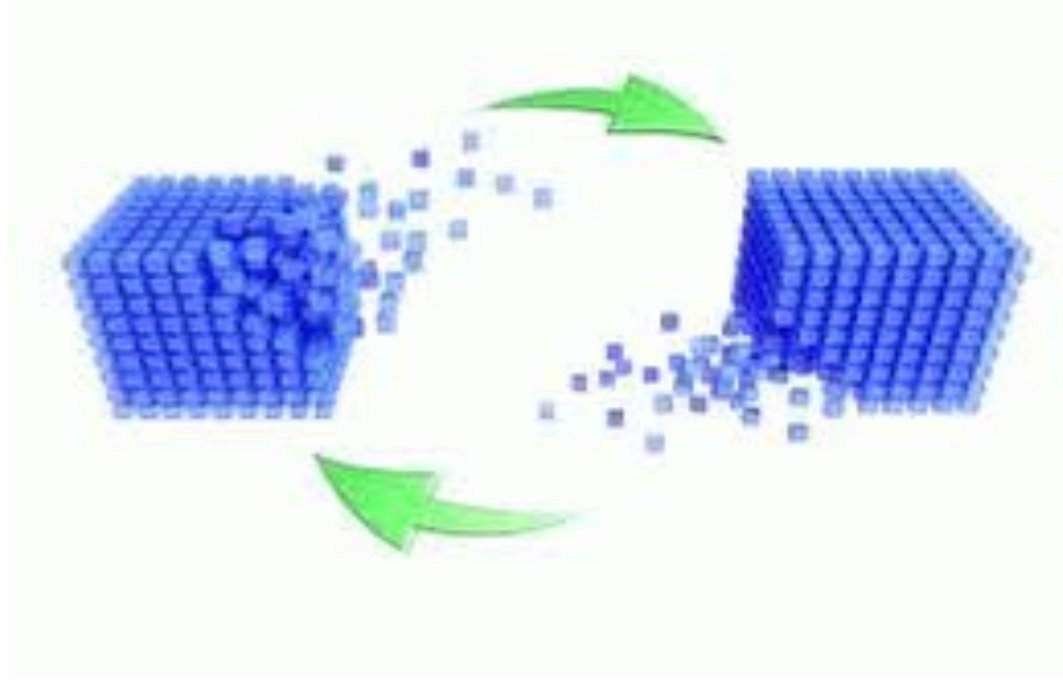
✓ **Twitter :**

2019 : 350 000 tweets envoyés par minute, 500 millions de tweets par jour et environ 200 milliards de tweets par an.

2021 : 206 millions utilisateurs par jours



SGBD



SYSTÈMES DE GESTION DE BASES DE DONNÉES

Définition :

Un Système de Gestion de Base de Données (**SGBD**) est un système qui permet de stocker des données dans une base de données. Un tel système permet de définir, modifier interroger et administrer les données d'une base de données.



LES PRINCIPAUX FONCTIONNALITÉS D'UN SGBD :

- 1. La définition des données
- 1. La manipulation des données
- 1. Le contrôle de données
- 1. L'intégrité des données
- 1. La gestion des accès concurrents
- 1. La confidentialité
- 1. La sécurité de fonctionnement



LES PRINCIPAUX FONCTIONNALITÉS D'UN SGBD :

- ❖ **La définition des données** : Grace au langage de définition de données (**LDD**), un SGBD offre la possibilité de définir les données d'une BDD.

CREATE : création d'une structure de données

ALTER : modification d'une structure de données

DROP : suppression d'une structure de données

RENAME : renommage d'une structure de données

- ❖ **La manipulation des données** : Grace au langage de manipulation de données (**LMD**), un SGBD offre la possibilité de manipuler les données

SELECT : sélection de données dans une table

INSERT : insertion de données dans une table

DELETE : suppression de données d'une table

UPDATE : mise à jour de données d'une table



LES PRINCIPAUX FONCTIONNALITÉS D'UN SGBD :

- ❖ **Le contrôle de données** : Grace au langage de contrôle de données (**LCD**), un SGBD offre la possibilité de contrôler les droits d'accès des utilisateurs (user 1, user 2, ...: consultation, modification).

GRANT : autorisation d'un utilisateur à effectuer une action

DENY : interdiction à un utilisateur d'effectuer une action

REVOKE : annulation d'une commande de contrôle de données précédente

COMMIT : validation d'une transaction en cours

ROLLBACK : annulation d'une transaction en cours

LOCK : verrouillage sur une structure de données



LES PRINCIPAUX FONCTIONNALITÉS D'UN SGBD :

- ❖ **L'intégrité des données :** Préserver la cohérence des données de la BDD

DEFAULT: quelle valeur par défaut à affecter à une colonne

NOT NULL : qu'une colonne ne peut pas être null

UNIQUE : qu'une colonne doit être unique

CHECK : un contrôle sur une colonne

PRIMARY KEY : préciser la clé primaire d'une table

FOREIGN KEY: définir la relation entre les tables

- ❖ **La gestion des accès concurrents :** Garantir l'accès simultanés par le SGBD (algorithmes)



LES PRINCIPAUX FONCTIONNALITÉS D'UN SGBD :

❖ **La confidentialité** : Authentification, droits d'accès.

le langage de gestion de la confidentialité est le sous
ensemble de SQL qui permet :

- ✓ À l'administrateur de : créer/détruire un utilisateur
- ✓ À chaque utilisateur de : donner/retirer une autorisation sur les objets qu'il possède

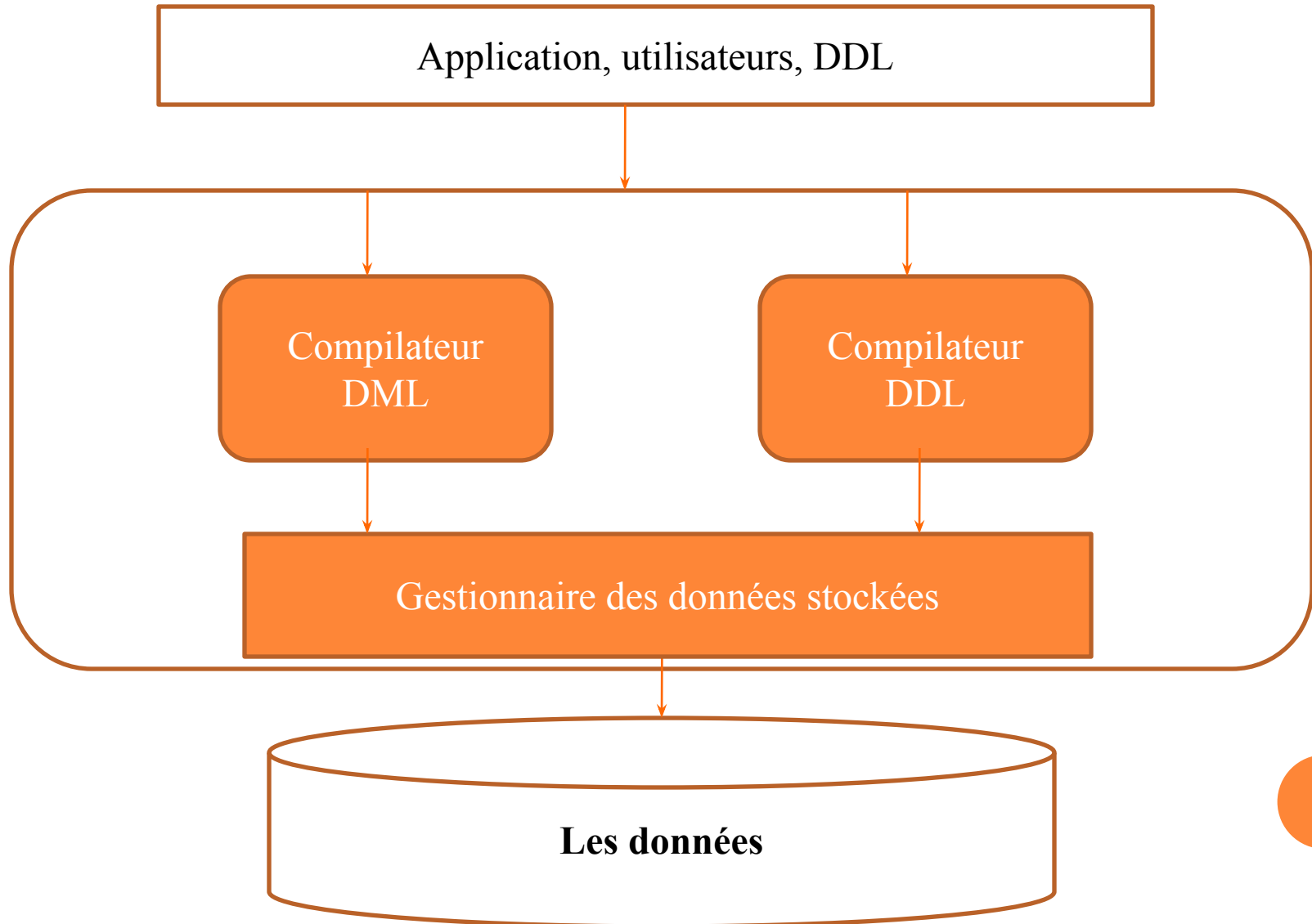


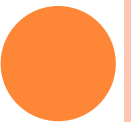
LES PRINCIPAUX FONCTIONNALITÉS D'UN SGBD :

- ❖ **La sécurité de fonctionnement** : Assure la persistance des données (copier de sauvegarde, journalisation).
- ❖ Exemple: injection SQL



STRUCTURE D'UN SGBD





LES COMPOSANTS D'UN GESTIONNAIRE DE DONNÉES :

- ❖ Gestionnaire d'autorisation
- ❖ Gestionnaire de transactions
- ❖ Gestionnaire de fichiers
- ❖ Gestionnaire du Buffer



LES COMPOSANTS D'UN SGBD

❖ Les principaux composants d'un SGBD sont les suivants:

1. Logiciel
2. Matériel
3. Données
4. Procédures
5. Langage d'accès à la BDD



LES COMPOSANTS D'UN SGBD

Langage d'accès à la BDD

- ❖ **SQL**: le langage des données structure signifie **Structured Query Language**, est un langage informatique normalisé permettant de communiquer avec une base de données relationnelle.
- ❖ **noSQL (not only SQL)** : destinée aux base de données non relationnelle.

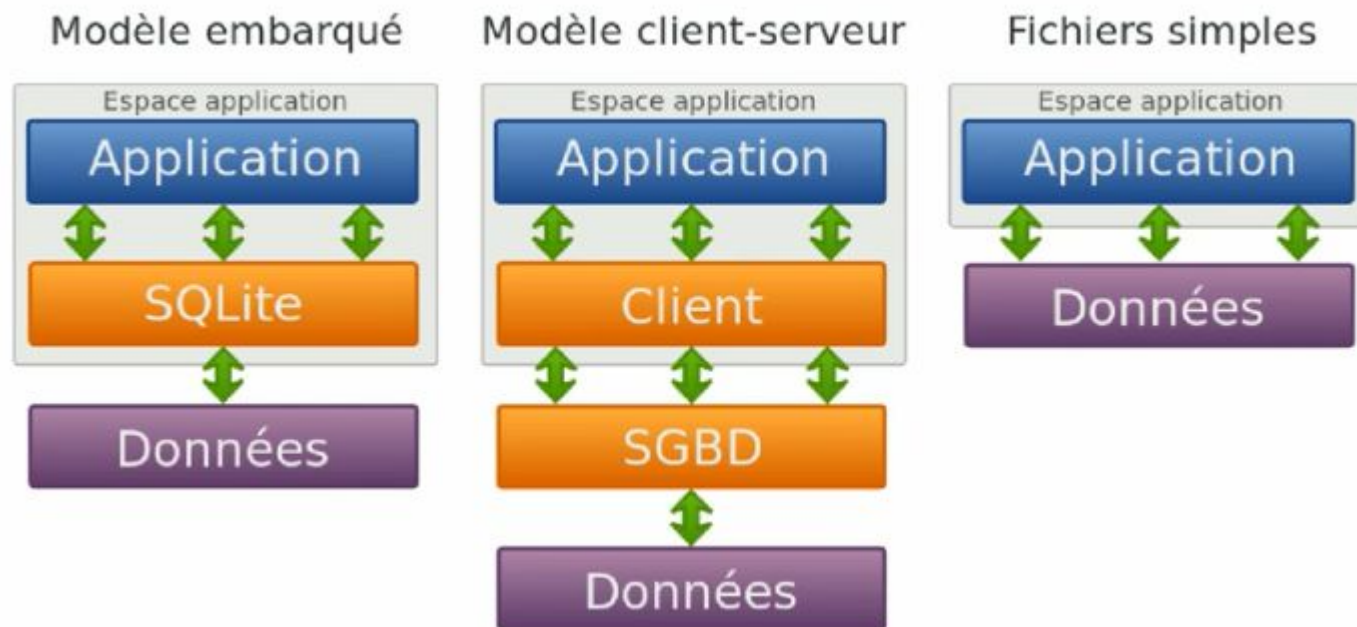
Les bases de données NoSQL sont largement utilisées dans les applications web en temps réel et le Big Data, car leurs principaux avantages sont une grande évolutivité et une haute disponibilité.

- ❖ Le NoSQL ne remplace pas le SQL, c'est juste une alternative.

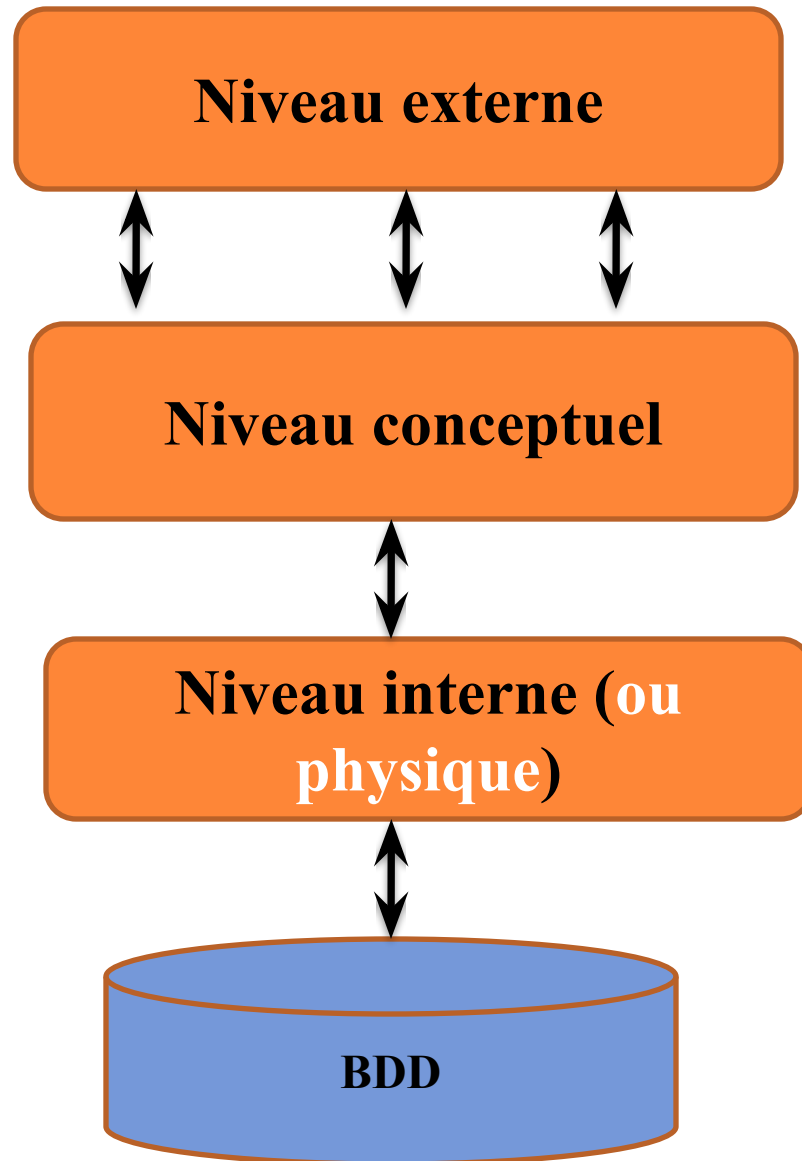


TYPES D'ARCHITECTURE DES SGBD

- ❖ Embarqué dans une application
- ❖ Connexion distante



L'ARCHITECTURE ANSI-SPARC



DÉFINITIONS:

- ❖ **Niveau Interne :** Description des données d'une base en termes de représentation physique en machine, correspondant à une spécification des structures de mémorisation et des méthodes de stockage et d'accès utilisées pour ranger et retrouver les données sur disques.
- ❖ **Niveau conceptuel :** Description des données en termes de types d'objets et de liens logiques indépendants de toute représentation en machine, correspondant à une vue canonique globale.
- ❖ **Niveau Externe:** Description d'une partie de la base de données extraite ou calculée à partir de la base physique, correspondant à la vision d'un programme ou d'un utilisateur, donc à un arrangement particulier de certaines données.



EXEMPLES DE SGBD

- ❖ **Système propriétaire:** Oracle Database, Microsoft SQL Server, DB2, MaxDB, 4D, dBase, Informix, Sybase.
- ❖ **Système libre** MySQL, PostgreSQL, MariaDB, Firebird, Ingres, HSQLDB, Derby, Apache Derby
- ❖ **Orienté objet :** ZODB, db4o
- ❖ **Embarqué :** SQLite Berkeley DB
- ❖ **NoSQL :** Cassandra, Redis? mONGOdb, SimpleDB, BigTable, CouchDB, Couchbase, HBase, LevelDB, RethinkDB, Memcached
- ❖ **Autre système :** Access, Openoffice.org FileMaker, HyperFileSQL, Paradox, Neo4j , Riak, Voldemort



STATIQTQUE DE

« [HTTPS://DB-ENGINES.COM/EN/RANKING](https://db-engines.com/en/ranking) »

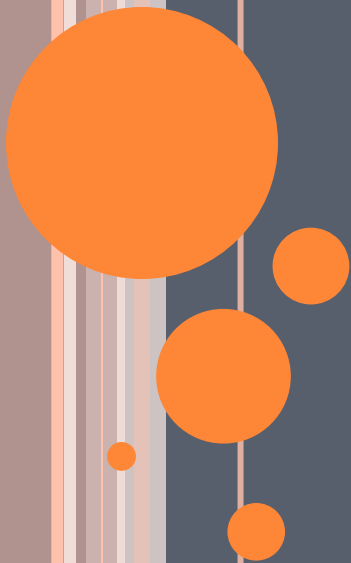
380 systems in ranking, October 2021

Rank			DBMS	Database Model	Score		
Oct 2021	Sep 2021	Oct 2020			Oct 2021	Sep 2021	Oct 2020
1.	1.	1.	Oracle +	Relational, Multi-model ⓘ	1270.35	-1.19	-98.42
2.	2.	2.	MySQL +	Relational, Multi-model ⓘ	1219.77	+7.24	-36.61
3.	3.	3.	Microsoft SQL Server +	Relational, Multi-model ⓘ	970.61	-0.24	-72.51
4.	4.	4.	PostgreSQL + ⓘ	Relational, Multi-model ⓘ	586.97	+9.47	+44.57
5.	5.	5.	MongoDB +	Document, Multi-model ⓘ	493.55	-2.95	+45.53
6.	6.	↑ 8.	Redis +	Key-value, Multi-model ⓘ	171.35	-0.59	+18.07
7.	7.	↓ 6.	IBM Db2	Relational, Multi-model ⓘ	165.96	-0.60	+4.06
8.	8.	↓ 7.	Elasticsearch	Search engine, Multi-model ⓘ	158.25	-1.98	+4.41
9.	9.	9.	SQLite +	Relational	129.37	+0.72	+3.95
10.	10.	10.	Cassandra +	Wide column	119.28	+0.29	+0.18



Chapitre 2 :

ARCHITECTURE DES SGBD DE PREMIERE GENERATION



BASE DE DONNÉES HIÉRARCHIQUE :

- ❖ Une **base de données hiérarchique** est une base de données dont le système de gestion lie les enregistrements dans une structure arborescente où chaque enregistrement n'a qu'un seul possesseur.
- ❖ Année 1970
- ❖ Gérer une grande quantité de données pour des projets de fabrication complexes.
- ❖ Exemple: Apollo Rocket en 1969.
- ❖ La structure logique de base est représentée par « **Upside-down tree** ».
- ❖ La structure hiérarchique contient des niveaux ou des segments.



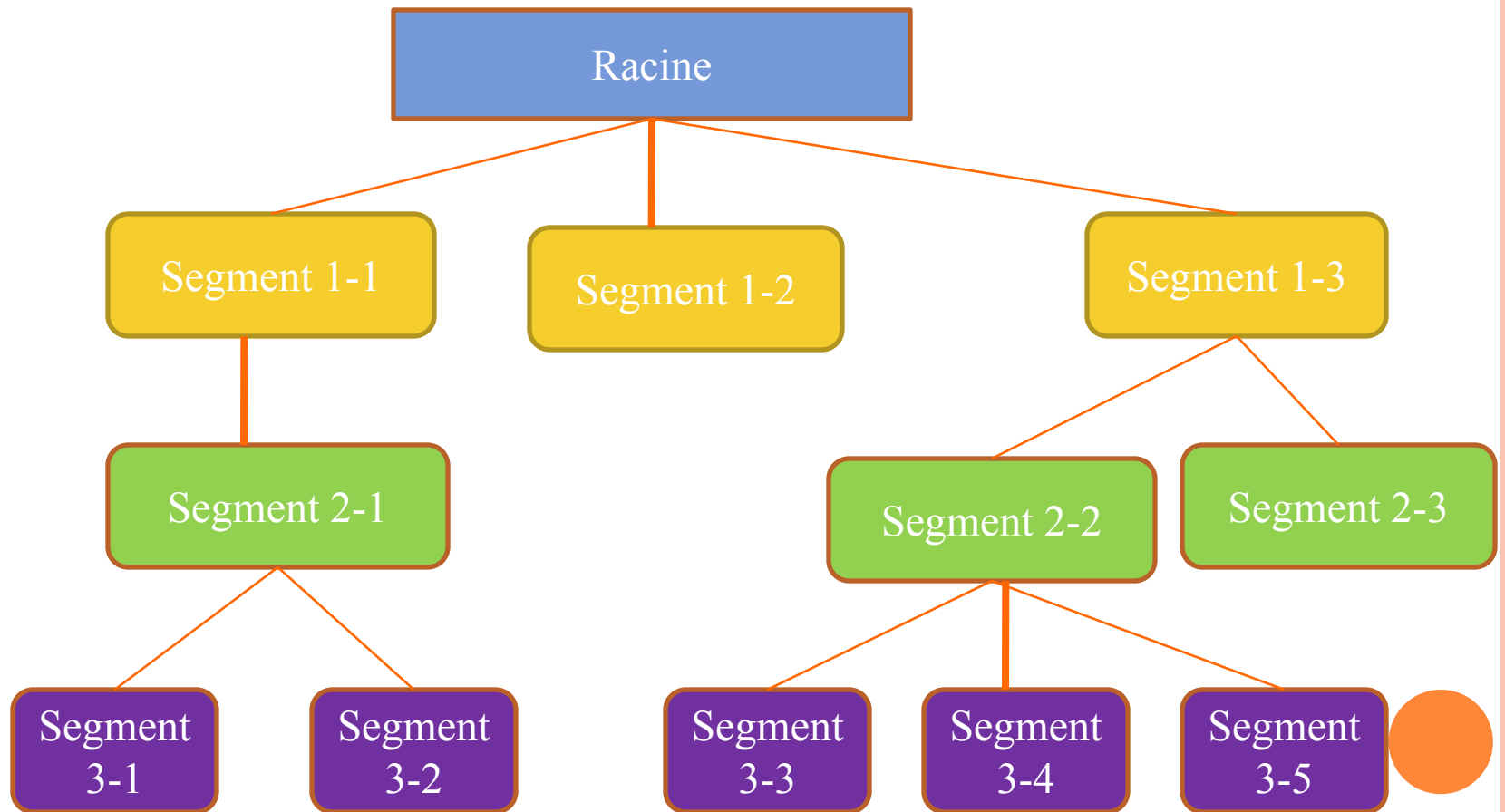
BASE DE DONNÉES HIÉRARCHIQUE :

- ❖ Les données sont stockés dans des fichiers
- ❖ Les relations entre ces données sont représentées sous la forme d'un arbre.
- ❖ (BDH) est composée d'une collection d'arbres
- ❖ Les segments sous d'autres segments sont les enfants du segment ci-dessus.
- ❖ Chaque parent a plusieurs enfants mais chaque enfant n'a qu'un seul parent.(Représentation 1-N)

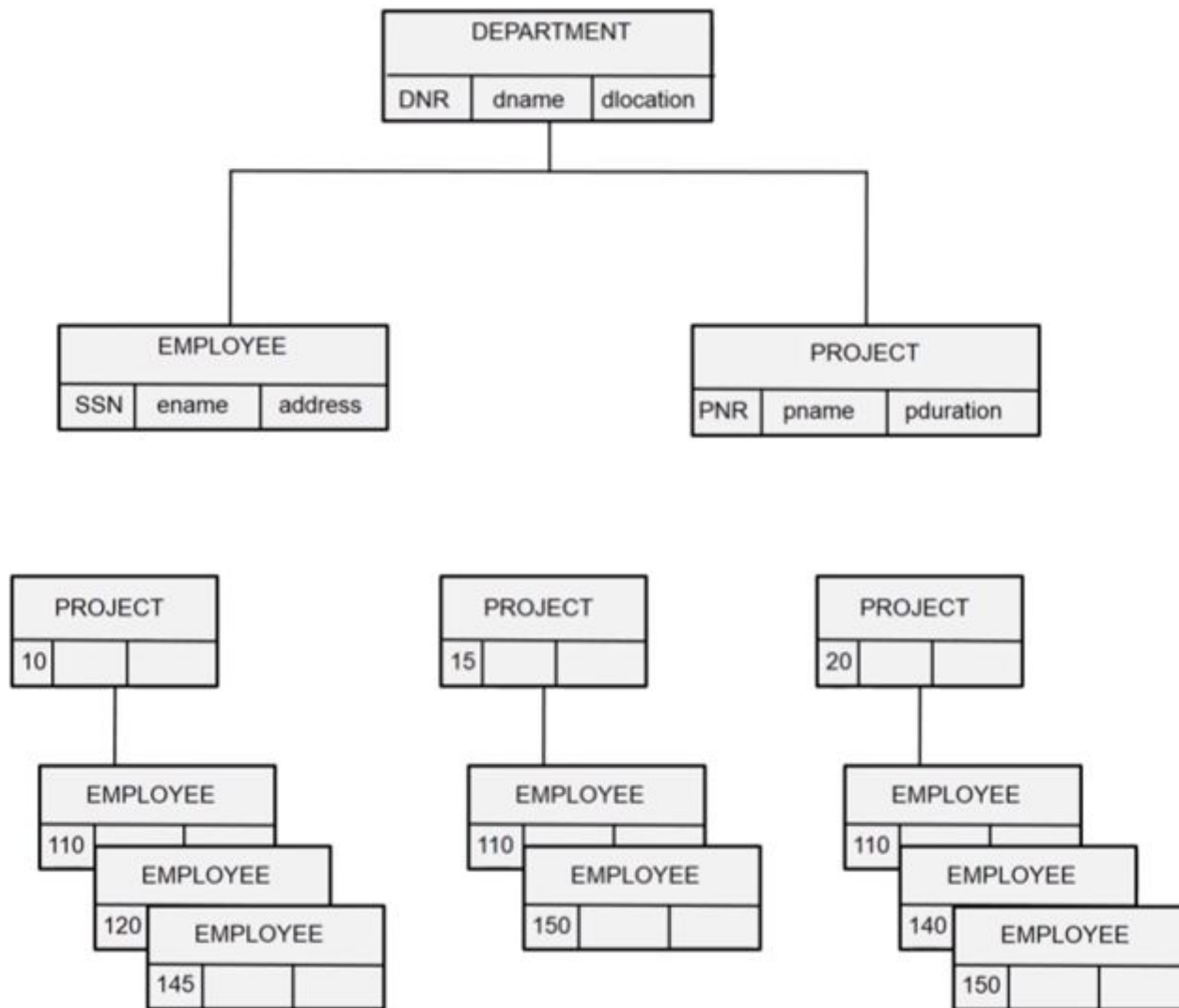


BASE DE DONNÉES HIÉRARCHIQUE :

❖ Structure de BDH



EXAMPLE



LES CONCEPTS DU MODÈLE

- ❖ Un **champ**: La plus petite unité de donnée, il s'agit d'une donnée élémentaire à valeur simple.
- ❖ Un **segment** : Collection de champs rangés consécutivement dans la base, portant un nom et le nombre d'occurrence.
- ❖ **Arbre de segments**: Collection de segments reliés par des associations père-fils, organisée sous la forme d'une hiérarchie.
- ❖ **Base de données hiérarchique**: Base de données constituée par une forêt de segments.



EXEMPLE « BDH »:

- ❖ **ADABAS** de software **AG**
- ❖ **Information Management System** de **IBM**
- ❖ Base de registre de **Windows**



Ses principaux **points faibles** sont :

- ❖ La recherche est difficile et coûteuse du fait que l'unique point d'entrée est la racine.
- ❖ Complexe à mettre en œuvre et difficile à gérer
- ❖ Manque de l'indépendance structurelle
- ❖ Une faible dépendance logique, qui induit un manque de sémantique.
- ❖ Le changement de la structure des données implique une modification des applications qui accèdent aux données.
- ❖ Le gaspillage de la mémoire (le contenu d'un enregistrement peut être répété).



BASE DE DONNÉES RÉSEAU

- ❖ Créer des relations complexes entre les données.
- ❖ Améliorer la performance des bases de données .
- ❖ Un segment fils peut avoir plusieurs parents.
- ❖ **Des liens multiples entre les segments** en plaçant des liens, ou des pointeurs, sur un ensemble d'enregistrements ou un autre.
- ❖ BDR a été Normalisée par le DBTG (**D**ata **B**ase **T**ask **G**roupe) de la **CODASYL**
- ❖ Il existe **deux types de BDR** , celles qui sont conformes aux spécifications rédigées par le DBTG, et celles qui ne s'y conforment pas.
- ❖ Les association **N-M** sont modélisées à l'aide d'un enregistrement supplémentaire.

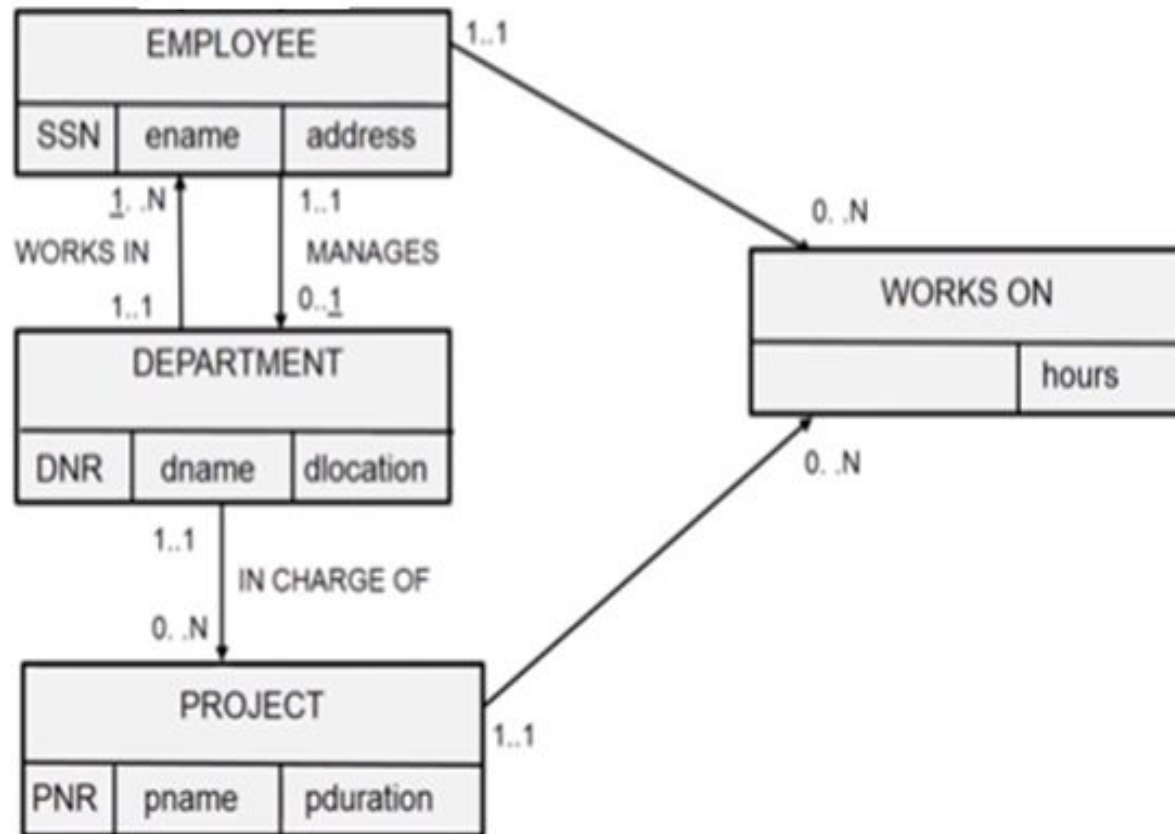


LES CONCEPTS DU MODÈLE

- ❖ **Atome:** Plus petite unité de données possédant un nom.
- ❖ **Groupe :** Collection d'atomes rangés consécutivement dans la base et portant un nom.
 - ✓ Les groupes simples
 - ✓ Les groupes répétitifs.
- ❖ **Article :** Collection d'atomes et de groupes rangés dans la base de données.



EXAMPLE



Ces principaux **défauts** sont :

- ❖ Trop complexe pour être utilisé efficacement.
- ❖ La navigation de la structure est encore plus complexe que dans le modèle hiérarchique.



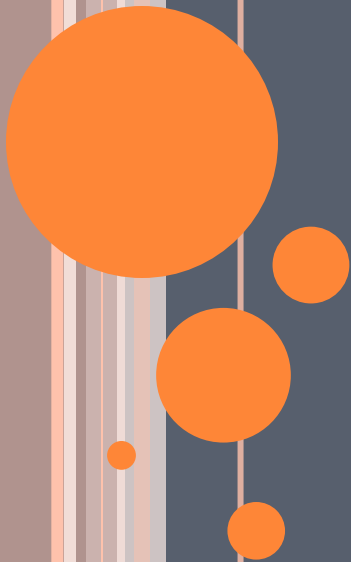
EXEMPLE « BDR » :

- ❖ IDMS
- ❖ DBMS
- ❖ EDMS
- ❖ IDS2
- ❖



Chapitre 3 :

ARCHITECTURE ET FONCTIONS DES SYSTEMES RELATIONNEL



LES BASES DE DONNES RELATIONNELLES

- ❖ Basé sur les travaux d'Edgar codd (Chercheur chez IBM).
- ❖ **Définition:** une **base de données relationnelle** est une base de données où la donnée est organisée dans des tableaux à deux dimensions appelés des *relations* ou *tables*



CONCEPTS DE BASE :

- ❖ **Relation** : ensemble de tuples.
- ❖ **Un Tuple** : ensemble d'attributs
- ❖ **Un attribut** : un nom donné à une colonne d'une relation, il prend ses valeurs dans un domaine.
- ❖ **Clé d'une relation** : un groupe d'attributs minimum qui détermine un tuple unique dans une relation
- ❖ **Clé étrangère** : groupe d'attributs devant apparaître comme clé dans une autre relation
- ❖ **Schéma de relation** : Nom de relation, liste des attributs avec domaines, et liste des clés d'une relation.



CONCEPTS DE BASE :

❖ **Modèle relationnel :**

L'information est organisée dans des tableaux à deux dimensions appelés **tables** avec une ou plusieurs relations.

❖ Les lignes de ces relations sont appelées des **enregistrements**, les colonnes sont appelées des attributs.

❖ **Algèbre relationnels** : ensemble d'opérations définies sur les relations, **8** opérateurs qui s'appliquent à des ensembles :

1. **4 Ensemblistes** (union, intersection, différence, produit cartésien)
2. **4 Relationnelles** (sélection, projection, jointure, division)



SGBD RELATIONNELLES

- ❖ **SGBDR** : SGBD manipulant des BDD structurées suivant le modèle relationnel.
- ❖ **Types de SGBDR:**
 1. **Client-Serveur**
 - ✓ Application web
 - ✓ Application distribuée
- ❖ **Définition :**
 - ❖ **Un client** est une machine (ordinateur) qui envoie des requêtes à un autre ordinateur appelé serveur.
 - ❖ **Un Serveur** : est un ordinateur puissant en terme de capacités d'entrée-sortie, offrant des nombreux services aux autres ordinateurs appelés **clients** via le réseau, et fonctionne en permanence.



SGBD RELATIONNELLES

❖ L'environnement CS est constitués de 3 parties:

1. Les nœuds d'ordinateur
2. Le réseau
3. Les applications

❖ Les caractéristiques Client-Serveur :

1. Attribut de client:

- ❖ Le client est proactif, il transmet les demandes aux serveur
- ❖ Il doit interagir avec au moins un traitement serveur
- ❖ Reprise après incident
- ❖ Les applications clients sont développés dans un langage hôte tel que: c, c++.....



SGBD RELATIONNELLES

❖ **Attribut de Serveur :**

stockage, organisation et MAJ des données partagées.

Il est déclenché à l'arrivée d'une demande du client

Il fonctionne sans arrêt

Les services peuvent être fournis directement ou indirectement

Exécuter plusieurs requêtes simultanément



SGBD RELATIONNELLES

- ❖ L'algèbre relationnelle fournit les opérateurs de base pour manipuler les extensions des relations (ensembles de n-uplets ou tables) d'une BD relationnelle.
- ❖ La relation : Voiture

NV	Puissance	marque	couleur
32	6	Renault	Blanc
40	7	Peugeot	Bleu
50	9	BMW	Blanc
17	8	BMW	Rouge



SGBD RELATIONNELLES

- ❖ Les principaux opérateurs de l'algèbre relationnelle sont les suivants :
- ❖ Renommage
- ❖ Union, intersection et différence , Division.
- ❖ Produit cartésien
- ❖ Sélection
- ❖ Projection
- ❖ Jointure



SGBD RELATIONNELLES

- ❖ **Renommage** : L'opérateur de renommage (ρ) permet de renommer les attributs d'une relation.
- ❖ **L'opérateur ρ est défini par :**

$\rho_{\text{nom, altitude}}$ (<table><tr><th>n</th><th>a</th></tr><tr><td>Everest</td><td>8848</td></tr><tr><td>Manaslu</td><td>8163</td></tr><tr><td>Hidden Peak</td><td>8086</td></tr></table>	n	a	Everest	8848	Manaslu	8163	Hidden Peak	8086)	=	<table><tr><th>nom</th><th>altitude</th></tr><tr><td>Everest</td><td>8848</td></tr><tr><td>Manaslu</td><td>8163</td></tr><tr><td>Hidden Peak</td><td>8086</td></tr></table>	nom	altitude	Everest	8848	Manaslu	8163	Hidden Peak	8086
	n	a																		
	Everest	8848																		
	Manaslu	8163																		
Hidden Peak	8086																			
nom	altitude																			
Everest	8848																			
Manaslu	8163																			
Hidden Peak	8086																			



SGBD RELATIONNELLES

❖ **Produit cartésien :**

❖ Soit p et q deux expressions telles que :

❑ $\text{schéma}(p) = (A_1, \dots, A_m)$

❑ $\text{schéma}(q) = (B_1, \dots, B_n)$

❑ $\{A_1, \dots, A_m\} \cap \{B_1, \dots, B_n\} = \emptyset$

❖ Le **produit cartésien** (\times) est défini par :

❑ $\text{schéma}(p \times q) = (A_1, \dots, A_m, B_1, \dots, B_n)$

n	a	\times	ns	p	$=$	n	a	ns	p
Everest	8848		Everest	Népal		Everest	8848	Everest	Népal
			Everest	Chine		Everest	8848	Everest	Chine



SGBD RELATIONNELLES

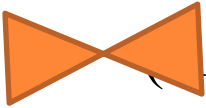
- ❖ **La sélection** : L'opérateur de sélection (σ) extrait l'ensemble des n-uplets d'une relation qui vérifient une condition donnée.
- ❖ σ (voiture) $\text{marque} = \text{BMW}$
- ❖ Sélectionner les voitures de la marque BMW

NV	Puissance	marque	couleur
32	6	Renault	Blanc
40	7	Peugeot	Bleu
50	9	BMW	Blanc
17	8	BMW	Rouge

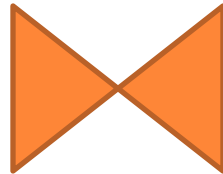


SGBD RELATIONNELLES

- ❖ L'opérateur de **Jointure** consiste à regrouper toutes les lignes de deux relations dont un ou plusieurs attributs sont identiques.

- ❖ Jointure : (R1)  (R2)
Id=Id-pays

Id	Pays
1	Cameron
2	Mali
3	Allemagne
4	USA
5	Argentine



Id	Nom	Prénom	Id-pays
1	Etto	samuel	1
2	Zuckerberg	Mark	4
3	Messi	lionel	5



SGBD RELATIONNELLES

- ❖ L'opérateur de **projection** (π) extrait l'ensemble des valeurs d'un constituant d'une relation.
- ❖ Π (voiture) NV, puissance

NV	Puissance	marque	couleur
32	6	Renault	Blanc
40	7	Peugeot	Bleu
50	9	BMW	Blanc
17	8	BMW	Rouge



SGBD RELATIONNELLES

- ❖ L'opérateur de **Union (U)**, **Différence (-)**, **Intersection (\cap)**
- ❖ Soit p et q deux expressions telles que : schéma(p) = schéma(q) = $(A_1: D_1, \dots, A_n: D_n)$

n	a
K2	8611
Everest	8848
Manaslu	8163

-

n	a
Everest	8848
Manaslu	8163

=

n	a
K2	8611

SGBD RELATIONNELLES

- ❖ L'opérateur de **division** (\div)
- ❖ Soit p et q deux expressions telles que : schéma(d)
schéma(D) \in

personne	age	métier
Dupont	20	Ingénieur
Dupont	20	Professeur
Durand	30	Professeur
Martin	40	Ingénieur
Martin	40	Professeur

\div

métier
Ingénieur
Professeur

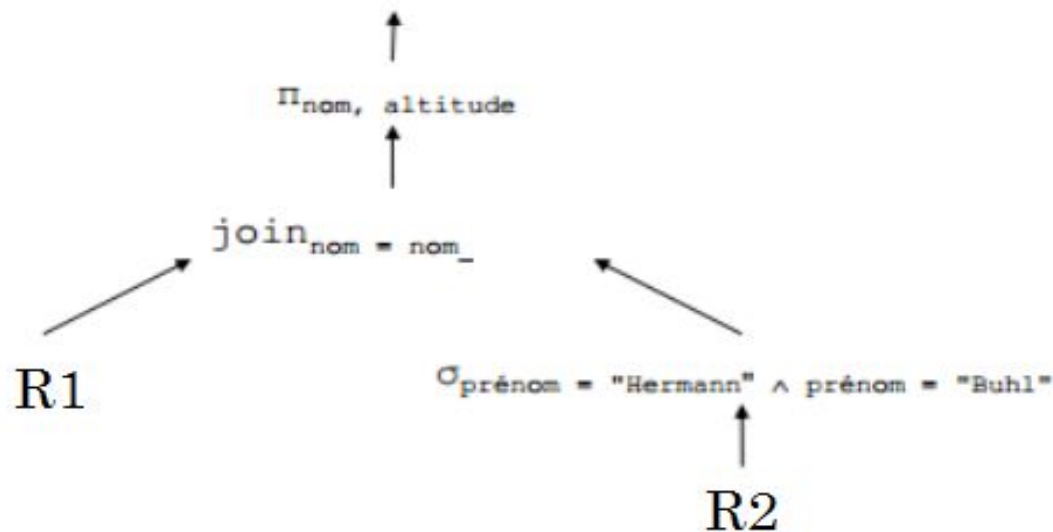


personne	age
Dupont	20
Martin	40



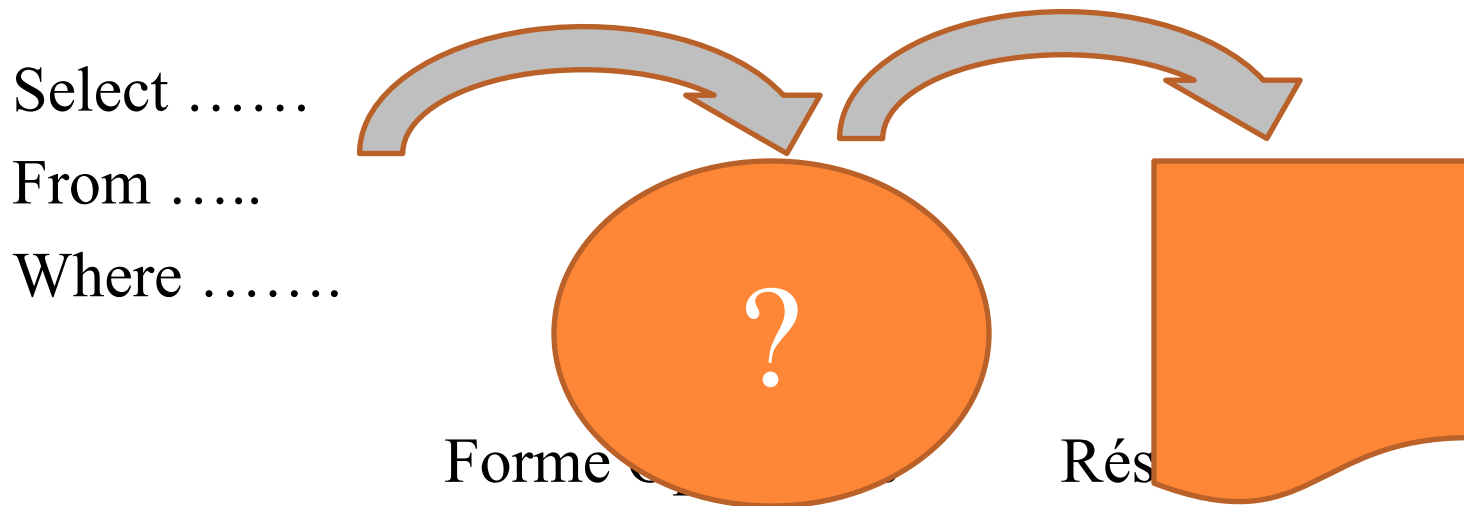
SGBD RELATIONNELLES

- ❖ **Arbre de requête** : Une requête de l'algèbre relationnelle peut être représentée par un arbre, appelé **arbre de requête**, dont les nœuds sont des opérateurs et les arcs des extensions de relations.



OPTIMISATION DES REQUÊTES

- ❖ La plupart des SGBDR sont dotés d'un optimiseur qui analyse de façon logique ou statistique la meilleure façon d'exécuter la requête.



OPTIMISATION DES REQUÊTES

- ❖ Est-ce que le SGBD crée un programme qui donne les bonnes performances ??

Réponse:

- ❖ Requête SQL
- ❖ Plans d'exécution logique
- ❖ Plans d'exécution physique
- ❖ Résultat



OPTIMISATION DES REQUÊTES

- ❖ Soit une requête Q :
- ❖ Comment ont passé du SQL qui est déclaratif à un arbre d'opérateurs (algébrique).
- ❖ Comment l'optimiseur obtient plusieurs plans d'exécution possibles
- ❖ Comment effectuer un choix parmi tous les plans
- ❖ Comment la requête est évalué par la suite.



- ❖ Ce sont des techniques de bases implémentés par les SGBD



OPTIMISATION DES REQUÊTES

- ❖ **Optimisation logique** : Elle permet de réécrire les requêtes sous une forme simplifiée est logiquement optimisé (sans prends en compte les couts d'accès aux données).
- ❖ **Optimisation physique** : Effectuer le choix des meilleurs algorithmes pour les opérateurs , prends en compte de la taille des données et des chemins d'accès disponibles.



OPTIMISATION DES REQUÊTES

❖ **Optimisation logique :**

1. **Analyse de requête :** procéder par la modalisation des graphes des relations et des attributs :
 - ❖ **Syntaxique** (noms, types)
 - ❖ **Sémantique** (non contradictoire)
 - ❖ **Graphes de connexion de relations et d'attributs**



OPTIMISATION DES REQUÊTES

❖ Graphes de connexion de relations :

- a) Un sommet représente chaque occurrence de relation
- b) Une jointure est représenté par un arc entre les deux sommets représentant les relations de type jointure
- c) Une restriction est représentée par une boucle sur la relation a la quelle elle s'applique
- d) La projection finale est représentée par un arc d'un nœud relation vers un nœud résultat



OPTIMISATION DES REQUÊTES

❖ **Graphes de connexion d'attributs :**

- ❖ un sommet est associé à chaque référence d'attribut ou constante
- ❖ Une jointure est représentée par un arc entre les attributs participants
- ❖ Une restriction est représentée par un arc entre un attribut et une constante

