Introduction aux Bases de Données

Ulysse COUTAUD
ulysse.coutaud@gmail.com

Base de données (BDD) ?

Ensemble d'informations:

- Stockées.
- Consultées.
- Modifiées.

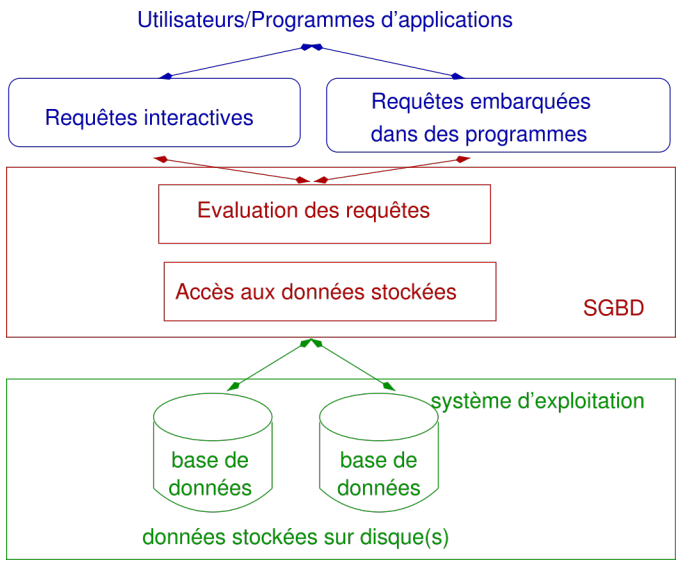
Exemples:

- Carnet d'adresses (quelques kilobytes).
- Comptes bancaires.
- Registre des personnels, des accès.
- Réservations des places dans les trains.
- Suivi des stocks, de la production, des commandes.
- Données de supervisions, programmes d'usinages, ...
- World Data Centre for Climate (6 petabytes).

Système de Gestion de Base de Données (SGBD) ?

Ensemble d'outils permettant **l'organisation, le contrôle, la consultation et la modification** d'une base de donnée.

- Accès et manipulation standardisés.
  - Indépendance physique des données
  - Indépendance logique des données
  - Réduire la complexité
  - Stockage et accès optimisés.
- Contrôle de cohérence des données
- Sécurité des données
- Accès ou transactions (massivement) concurrentes.
- Fiabilité (sauvegardes, restaurations, ...)



Un exemple

Un outil de gestion d'atelier: GEDIX.

Dans ce cours

Rudiments des Bases de Données:

- Le modèle "relationnel"
- Le langage SQL
- PostgreSQL
- Lire les données
- Modifier les données
- Concevoir une base de données simple

Un peu de vocabulaire

- **BDD** Base De Données
- **DB** Database

- **SGBD** Système de Gestion de Base de Données
- **DBMS** Data Base Management System

- **SGBDR** Système de Gestion de Base de Données Relationnelles
- **RDBMS** Relational Database Management System

- **SQL** Structured Query Language : Norme/langage de manipulation de données relationnelles.

Les principaux acteurs

- ORACLE :
  - Oracle DB
  - MySQL
- Microsoft
  - MS SQL: Microsoft SQL Server
  - Microsoft Access:
- IBM: IBM DB2
- SQLlite: SQLite

PostgreSQL

- **PostgreSQL**
- Open source
- Université de Berkeley
- Très populaire et répandu [a]

[a] https://db-engines.com/en/ranking_trend/relational+dbms

Le modèle relationnel

Une TABLE = une "RELATION"

Clé primaire

Colonne/attribut

TABLE_Employés(prénom,salaire,adresse,service)			
Prénom	Salaire	Adresse	Service
John	120	Randwick	Production
Mary	130	Wollolong	Maintenance
Peter	110	Randwick	Stock
Tom	120	Botany Bay	Production

Schéma de la relation

Ligne/"enregistrement"

Le modèle relationnel

Une TABLE = une "RELATION"

Clé primaire

Colonne/attribut

TABLE_Employés(prénom,salaire,adresse,service)			
Prénom	Salaire	Adresse	Service
John	120	Randwick	Production
Mary	130	Wollolong	Maintenance
Peter	110	Randwick	Stock
Tom	120	Botany Bay	Production

Schéma de la relation

Ligne/"enregistrement"

TABLE_Horaires(service,début,fin)		
Service	Début	Fin
Production	5	13
Maintenance	8	16
Stock	8	16

L'ensemble des schémas de relations + liens entre eux = le schéma de la base de données.

Se connecter à la base de données

TABLE Employes(prenom,salaire,adresse,service)

Prenom	Salaire	Adresse	Service
John	120	Randwick	Production
Mary	130	Wollolong	Maintenance
Peter	110	Randwick	Stock
Tom	120	Botany Bay	Production

TABLE Horaires(service,debut,fin)

Service	Debut	Fin
Production	5	13
Maintenance	8	16
Stock	8	16

Figure: BDD "atelier"

- Ouvrir ligne de commande (Win + cmd)
- Taper "psql fabrique postgres"
- Entrer le mot de passe "postgres"
- Lister les tables de la BDD courante \dt

Clause SELECT

BDD "atelier":

TABLE Employes(prenom,salaire,adresse,service)

Prenom	Salaire	Adresse	Service
John	120	Randwick	Production
Mary	130	Wollolong	Maintenance
Peter	110	Randwick	Stock
Tom	120	Botany Bay	Production

TABLE Horaires(service,debut,fin)

Service	Debut	Fin
Production	5	13
Maintenance	8	16
Stock	8	16

Afficher toute la table

```
SELECT * FROM nom_table;
```

Afficher certaines colonnes

```
SELECT colonne1, colonne2 FROM table;
```

Afficher les prénoms et salaires des employés.

Afficher toutes les informations des employés.

Clause SELECT avec contraintes

BDD "atelier":
employes(prenom,salaire,adresse,service) horaires(service,debut,fin)

SELECT colonne1, colonne2 FROM tableau WHERE condition1 AND/OR NOT condition2;

Nombres:

=, !=, <, <=, >, >=,
BETWEEN nombre1 AND nombre2,
IN (nombre1, nombre2, nombre3)

Texte:

Sensible à la casse: =, != ou <>
Non sensible à la casse: LIKE
(caractère joker: %)
IN ("mot1", "mot2", "mot3")

Afficher les prénoms des employés du service production.
Afficher les prénoms des employés qui gagne 110 ou moins.
Afficher les prénoms des employés dont l'adresse contient la chaine "an".
Afficher les prénoms et salaires des employés du service Production et Maintenance.

Filtrer les résultats

BDD "atelier":
employes(prenom,salaire,adresse,service) horaires(service,debut,fin)

Clause DISTINCT

```
SELECT DISTINCT colonne1 FROM table;
```

Afficher sans répétition la liste des services.
Afficher sans répétition les débuts de services.

Filtrer les résultats

BDD "atelier":
employes(prenom,salaire,adresse,service) horaires(service,debut,fin)

Clause ORDER BY

SELECT colonne1 FROM table ORDER BY colonne1 ASC/DESC;

Afficher toutes les informations sur les employes en les triants par ordre alphabétique sur les prénoms.
Afficher prénoms et adresses des employes en triant par salaires décroissants.

Filtrer les résultats

BDD "atelier":
employes(prenom,salaire,adresse,service) horaires(service,debut,fin)

Clauses LIMIT et OFFSET

SELECT colonne1 FROM table LIMIT num_limit OFFSET num_offset;

Afficher l'employé le moins bien payé.
Afficher le deuxieme employé le moins bien payé.

Requêtes sur plusieurs tables

BDD "atelier":
employes(prenom,salaire,adresse,service) horaires(service,debut,fin)

Clause INNER JOIN

SELECT colonne1 FROM table1 INNER JOIN table2 ON table1.colonneX=table2.colonneY;

Remarque

1: Le INNER JOIN est le JOIN par defaut, le mot clé "INNER" n'est pas nécessaire.
2: En cas d'ambiguité sur le nom de colonne, on ajoute en préfixe la table visé (ex: table1.colonne1). On peut également renommer la colonne résultat avec le mot clé "AS" (ex: table1.colonne1 AS T1C1)

Afficher la jonction de toutes les informations de la BDD
Afficher les horaires de John.
Afficher les noms et horaires de tous les employés du service production.

Requêtes sur plusieurs tables

BDD "atelier":
employes(prenom,salaire,adresse,service) horaires(service,debut,fin)

LEFT JOIN

Il y aura forcément chacune des lignes de table1. Si aucune correspondance avec table2: NULL.

RIGHT JOIN

Il y aura forcément chacune des lignes de table2. Si aucune correspondance avec table1: NULL.

Valeur NULL

Se filtre avec IS NULL ou IS NOT NULL

Afficher les prénoms des employés qui n'ont pas d'horaires.

Les expressions

BDD "atelier":
employes(prenom,salaire,adresse,service) horaires(service,debut,fin)

```
SELECT colonne1, colonne2 * 10 AS colonne2_pourcent FROM table;
SELECT colonne1, colonne2 FROM table WHERE colonne2 * 10 > 80;
```

Les salaires de la base sont journaliers. Afficher les prénoms et salaires mensuels (20 jours travaillés).

Les aggrégats

BDD "atelier":
employes(prenom,salaire,adresse,service) horaires(service,debut,fin)

```
SELECT COUNT(colonne1) from table;
```

Les fonctions d'aggrégats:

COUNT, MIN, MAX, AVG, SUM.

Afficher le nombre d'employés.
Afficher le salaire minimal.
Afficher le salaire moyen.
Afficher la masse salariale totale.

Les aggrégats

BDD "atelier":
employes(prenom,salaire,adresse,service) horaires(service,debut,fin)

Clause GROUP BY

SELECT colonne1, SUM(colonne2) FROM table GROUP BY colonne1; Applique la fonction d'aggrégat à chaque groupe ayant la même valeur en colonne1.

Afficher la masse salariale de chaque service.
Afficher le salaire moyen par adresse.

Clause HAVING

Equivalent de WHERE pour les expressions.

Afficher le salaire moyen des groupes d'employés ayant tous le même salaire à une même adresse.