

Bases de données

© Olivier Caron

Bureau F-016

Olivier.Caron@polytech-lille.fr

Systeme d'information et bases de données

- ✓ Une formation sur 3 ans !
 - ▶ Systeme d'information → Analyse, conception
 - ▶ Bases de données → structuration, requêtes
 - ▶ Domaines liés : programmation, réseau, sécurité, statistiques, . . .

GIS 1 : introduction aux bases de données

- ✓ Les objectifs des bases de données
- ✓ La modélisation de systèmes d'information
- ✓ Le modèle relationnel
 - ▶ modèle de données
 - ▶ requêtes
- ✓ Programmation et bases de données
- ✓ Introduction à l'analyse

Le programme (1/2)

- ✓ 22 heures de cours
- ✓ 14 heures de TD
- ✓ 10 heures de TP
- ✓ 1 projet tutoré (25 heures)
- ✓ Contrôle des connaissances :
 - ▶ 1 DS (2 heures), un contrôle TP individuel (1 heure) et projet
 - ▶ Tous documents **papiers** autorisés

Le programme (2/2)

✓ Plates-formes :

- ▶ SGBD : Postgres sous unix (<http://www.postgresql.org>)
- ▶ le langage C (TP)
- ▶ En projet : HTML, le langage PHP, serveur web Apache
<http://www.w3c.org>,
<http://www.apache.org>,
<http://www.php.net>

Bibliographie

- ✓ *Introduction aux bases de données*, Serge Miranda et José-Maria Busta
- ✓ *Les bases de données relationnelles*, Serge Miranda et José-Maria Busta
- ✓ *Bases de données*, George Gardarin
- ✓ *Des Bases de données à l'Internet*, Philippe Mathieu
(2 exemplaires au centre de documentation Polytech'Lille)
- ✓ et. . .le **support** de cours à
<http://www.polytech-lille.fr/~ocaron>

Historique (1/4)

- ✓ Définition base de données :
juste un ensemble d'informations. . .
- ✓ Ne définit pas : la structure, le nombre, les besoins. . .

Historique (1/4)

- ✓ Définition base de données :
juste un ensemble d'informations. . .
- ✓ Ne définit pas : la structure, le nombre, les besoins. . .
- ✓ Première structuration : le fichier
- ✓ Définition : c'est un ensemble de données qui peut être manipulé par plusieurs utilisateurs ayant une vue unique de ces données.
C'est un ensemble d'enregistrements physiques.

Historique (2/4)

- ✓ Fichier, version support papier :
 - ▶ Information conservée dans les différents services.
 - ▶ Localisation de l'information (Quel bureau ? Quel tiroir ?).
 - ▶ Est-elle sous la forme souhaitée ?
 - ▶ Éclatée entre plusieurs services.
 - ▶ Les informations importantes sont dupliquées. Sont-elles identiques ?

Historique (3/4)

- ✓ Évolution fichier, support informatique (1960) :
 - ▶ Fichiers de natures différentes, manipulés par des programmes différents, dans des langages différents avec des formalismes différents.
 - ▶ Centralisation physique, mais pas logique.
 - ▶ Pour toute nouvelle application, où sont les informations utiles ?

Historique (4/4)

- ✓ Évolution : SGBD (Systèmes de Gestion de bases de données)
 - ▶ Archivage sous un modèle commun.
 - ▶ Centralisation (pas de duplication, donc pas d'incohérence possible).
 - ▶ Langage commun de manipulation.
 - ▶ Sécurité (Qui a le droit de lire ou d'écrire, contrôle d'intégrité).

Définitions

- ✓ Base de données :
ensemble d'informations

Définitions

- ✓ Base de données :
ensemble d'informations
- ✓ Système de gestions de bases de données :
ensemble de logiciels et matériels permettant de gérer des bases de données

Les objectifs des bases de données

- ✓ Les objectifs principaux :
 - ▶ Indépendance physique des données
 - ▶ Indépendance logique des données
 - ▶ Manipulation des données par des non-informaticiens
 - ▶ Efficacité des accès aux données
 - ▶ Administration centralisée des données
 - ▶ Non redondance des données
 - ▶ Cohérence des données
 - ▶ Partageabilité des données
 - ▶ Sécurité des données

Objectif : Indépendance Physique (1974)

- ✓ *Représentation du monde réel sous forme indépendante de la structure physique*
- ✓ Contre-exemple : telle valeur est située à telle adresse en mémoire.
- ✓ En informatique, les utilisateurs de SGBD n'ont pas à se soucier de :
 - ▶ Structures de stockage, fichiers, listes, tableaux, disque dur, système de sauvegarde, . . .

Objectif : Indépendance Logique (1974)

- ✓ Idée : Chaque utilisateur doit pouvoir ne connaître qu'une partie de la sémantique des données et ne voir qu'une partie des données

Objectif : Indépendance Logique (1974)

- ✓ Idée : Chaque utilisateur doit pouvoir ne connaître qu'une partie de la sémantique des données et ne voir qu'une partie des données
- ✓ Avantages :
 - ▶ On peut modifier (virtuellement) la structure

Objectif : Indépendance Logique (1974)

- ✓ Idée : Chaque utilisateur doit pouvoir ne connaître qu'une partie de la sémantique des données et ne voir qu'une partie des données
- ✓ Avantages :
 - ▶ On peut modifier (virtuellement) la structure
 - ▶ On ne dispose que des informations pertinentes

Objectif : Indépendance Logique (1974)

- ✓ Idée : Chaque utilisateur doit pouvoir ne connaître qu'une partie de la sémantique des données et ne voir qu'une partie des données
- ✓ Avantages :
 - ▶ On peut modifier (virtuellement) la structure
 - ▶ On ne dispose que des informations pertinentes
 - ▶ Sécurité (on ne peut pas accéder à des données qu'on ne connaît pas.

Objectif : Indépendance Logique (1974)

- ✓ Idée : Chaque utilisateur doit pouvoir ne connaître qu'une partie de la sémantique des données et ne voir qu'une partie des données
- ✓ Avantages :
 - ▶ On peut modifier (virtuellement) la structure
 - ▶ On ne dispose que des informations pertinentes
 - ▶ Sécurité (on ne peut pas accéder à des données qu'on ne connaît pas.
 - ▶ Plusieurs représentations sont possibles destinées à un utilisateur ou un groupe d'utilisateurs
- ✓ Architecture Ansi-SPARC

Objectif : manipulation des données par des non-informaticiens

- ✓ Les utilisateurs (non informaticiens) peuvent :
 - ▶ consulter les données
 - ▶ découvrir les données (aspect Méta)
 - ▶ mettre à jour (fixer des barrières !)
- ✓ Il faut donc définir un langage le plus simple et le plus intuitif possible :
 - ▶ QBE (Query By Example) : simple pour...des requêtes simples (graphique)
 - ▶ SQL (Structured Query Language) : standard de fait, très puissant

Objectif : efficacité des accès aux données

- ✓ Réservé aux utilisateurs informaticiens :
 - ▶ Offrir un langage de programmation et une bibliothèque (API) permettant la gestion de bases de données.
 - ▶ Bibliothèques propriétaires : PL-SQL (Oracle), VBA (Ms Access) , libPostgres (C) , php et postgres, . . .
 - ▶ Standards (de fait) : ODBC (Microsoft), JDBC (Sun JavaSoft)

Objectif : efficacité des accès aux données

- ✓ Réservé aux utilisateurs informaticiens :
 - ▶ Offrir un langage de programmation et une bibliothèque (API) permettant la gestion de bases de données.
 - ▶ Bibliothèques propriétaires : PL-SQL (Oracle), VBA (Ms Access) , libPostgres (C) , php et postgres, . . .
 - ▶ Standards (de fait) : ODBC (Microsoft), JDBC (Sun JavaSoft)
- ✓ Algorithmes d'optimisation de requêtes
- ✓ Support réseau

Objectif : Administration cohérente des données

- ✓ L'administration est souvent centralisée. Le rôle du DBA :
 - ▶ Contrôle efficace des données, sauvegarde
 - ▶ Résoudre les conflits
 - ▶ Optimiser les accès aux données
 - ▶ Prise en compte de la charge (nombre d'utilisateurs connectés, charge disque, . . .)

Objectif : non redondance des données

- ✓ Eviter la perte de place en mémoire
- ✓ Eviter les mises à jour multiples
- ✓ Exemple :
 - Personne (nom, prenom, age, qualite)
 - Adresse (nom, prenom, rue , ville, code postal)

Objectif : non redondance des données

- ✓ Eviter la perte de place en mémoire
- ✓ Eviter les mises à jour multiples
- ✓ Exemple :
 - Personne (nom, prenom, age, qualite)
 - Adresse (nom, prenom, rue , ville, code postal)

 - Personne (numID, nom, prenom, age, qualite)
 - Adresse (numID, rue , ville, code postal)
- ✓ A prendre en compte lors de la conception de la base

Objectif : Cohérence des données

- ✓ Notion d'intégrité (de domaine, référentielle, . . .)
- ✓ Fournir des outils de contrôle automatique

Partageabilité des données

- ✓ Permettre aux applications de partager les données de la base dans le temps mais aussi simultanément.
- ✓ L'utilisateur n'a pas à se soucier des autres accès !
- ✓ → systèmes de transactions

Objectif : sécurité des données

- ✓ Fournir des mécanismes pour définir des droits d'accès.
- ✓ Les données doivent être protégées contre les accès non autorisés ou mal intentionnés.
- ✓ Les droits d'accès peuvent également dépendre de la valeur des données :
 - ▶ Exemple : un employé pourra connaître les salaires des personnes qu'il dirige mais pas des autres employés de l'entreprise.

La qualité d'un SGBD, c'est aussi

✓ Gérer des données importantes (datawarehouse, datamining)

La qualité d'un SGBD, c'est aussi

- ✓ Gérer des données importantes (datawarehouse, datamining)
- ✓ Gérer des données réparties sur plusieurs serveurs

La qualité d'un SGBD, c'est aussi

- ✓ Gérer des données importantes (datawarehouse, datamining)
- ✓ Gérer des données réparties sur plusieurs serveurs
- ✓ Accepter un grand nombre de connexions

La qualité d'un SGBD, c'est aussi

- ✓ Gérer des données importantes (datawarehouse, datamining)
- ✓ Gérer des données réparties sur plusieurs serveurs
- ✓ Accepter un grand nombre de connexions
- ✓ Les temps de réponse !

La qualité d'un SGBD, c'est aussi

- ✓ Gérer des données importantes (datawarehouse, datamining)
- ✓ Gérer des données réparties sur plusieurs serveurs
- ✓ Accepter un grand nombre de connexions
- ✓ Les temps de réponse !
- ✓ La reprise après panne (dans un état cohérent), système de réplication, . . .

La qualité d'un SGBD, c'est aussi

- ✓ Gérer des données importantes (datawarehouse, datamining)
- ✓ Gérer des données réparties sur plusieurs serveurs
- ✓ Accepter un grand nombre de connexions
- ✓ Les temps de réponse !
- ✓ La reprise après panne (dans un état cohérent), système de réplication, . . .
- ✓ L'ergonomie

La qualité d'un SGBD, c'est aussi

- ✓ Gérer des données importantes (datawarehouse, datamining)
- ✓ Gérer des données réparties sur plusieurs serveurs
- ✓ Accepter un grand nombre de connexions
- ✓ Les temps de réponse !
- ✓ La reprise après panne (dans un état cohérent), système de réplication, . . .
- ✓ L'ergonomie
- ✓ Le prix :-)

Les SGBD relationnels

✓ SGBDR représentent 80% du marché

Les SGBD relationnels

- ✓ SGBDR représentent 80% du marché
- ✓ Le SGBD parfait n'existe pas !

Les SGBD relationnels

- ✓ SGBDR représentent 80% du marché
- ✓ Le SGBD parfait n'existe pas !
- ✓ Les Pros :
 - ▶ Oracle, Informix, Sybase, DB2, . . .

Les SGBD relationnels

- ✓ SGBDR représentent 80% du marché
- ✓ Le SGBD parfait n'existe pas !
- ✓ Les Pros :
 - ▶ Oracle, Informix, Sybase, DB2, . . .
- ✓ Les semi-pros :
 - ▶ Postgres, MS-SQL server, . . .

Les SGBD relationnels

- ✓ SGBDR représentent 80% du marché
- ✓ Le SGBD parfait n'existe pas !
- ✓ Les Pros :
 - ▶ Oracle, Informix, Sybase, DB2, . . .
- ✓ Les semi-pros :
 - ▶ Postgres, MS-SQL server, . . .
- ✓ et les autres :
 - ▶ Ms Access, MySQL, . . .

Les SGBD relationnels

- ✓ SGBDR représentent 80% du marché
- ✓ Le SGBD parfait n'existe pas !
- ✓ Les Pros :
 - ▶ Oracle, Informix, Sybase, DB2, . . .
- ✓ Les semi-pros :
 - ▶ Postgres, MS-SQL server, . . .
- ✓ et les autres :
 - ▶ Ms Access, MySQL, . . .

Les SGBD relationnels

- ✓ SGBDR représentent 80% du marché
- ✓ Le SGBD parfait n'existe pas !
- ✓ Les Pros :
 - ▶ Oracle, Informix, Sybase, DB2, . . .
- ✓ Les semi-pros :
 - ▶ Postgres, MS-SQL server, . . .
- ✓ et les autres :
 - ▶ Ms Access, MySQL, . . .