Bases de données et Logique

M1 Informatique

Année 2018-2019

Philippe Pucheral

Contenu du module

- Qu'est ce qu'un SGBD et comment s'en sert-on ?
  - Modèle EA, modèle relationnel, SQL, programmation, intégrité, vues, droits d'accès
- Comment fonctionne un SGBD ?
  - Stockage et indexation, évaluation de requêtes, transactions, la base de l'administration de données
- Supports de cours
  - En ligne sur e-campus

Mais d'abord :

Introduction et Objectifs

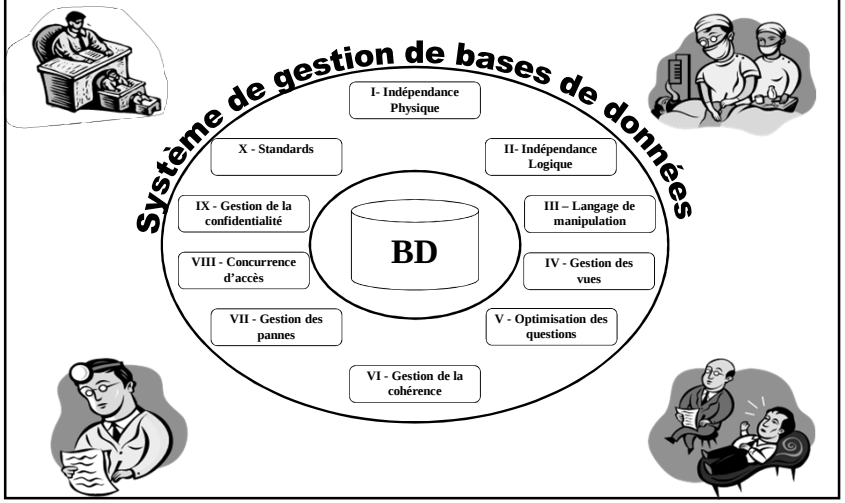
1. Introduction

- Les volumes de données à gérer sont de plus en plus grands
  - Giga (10^9), Tera (10^{12}), Peta (10^{15}), Exa (10^{18}), Zetta (10^{21}) – octets
  - Données créées : 1992 = 100 GB / jour, 2002 = 100 GB / sec, 2018 = 50.000 GB / sec
  - 269 milliards de mails échangés / jour en 2017
  - 3,3 milliards de requêtes / jour sur Google
  - 15.000 transactions/sec sur First American (Assurance)
- Les données sont multi-formes
  - Données numériques, textuelles, multimédia (images, films,...), flux (capteurs, RSS, …)
  - Elles sont plus faciles à gérer quand elles sont structurées et identifiées
- Il faut pouvoir facilement
  - Archiver ces données
  - Retrouver les données pertinentes à un traitement
  - Protéger les données contre les pannes
  - Protéger la confidentialité de ces données
  - Analyser ces données
  - …

Un peu d'histoire

- Années 60-70:
  - Avènement des Bases de Données Hiérarchiques et Réseaux
  - Fichiers reliés par des pointeurs, langage navigationnel
  - Invention liée au programme Apollo de la NASA
- Années 80:
  - Avènement des Bases de Données Relationnelles (BDR)
  - Relations entre ensemble de données, langage déclaratif
- Années 90:
  - Avènement des Bases de Données Objets (BDO) puis semi-structurées (XML)
  - Finalement intégré au relationnel
- Aujourd'hui:
  - Key-Value Stores, NoSQL … ou comment gérer le gigantisme
  - Mais toujours une place de choix pour les SGBD Relationnels

2. Objectifs des SGBD



O1: Description canonique des données

- Description cohérente, unique et centralisée des données manipulées par l'ensemble des applications constituant le système d'information.

— Perception globale du système d'information

=> augmentation du niveau d'informatisation

=> nouveaux traitements (aide à la décision, analyse de données, …)

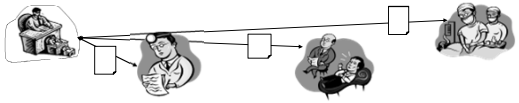
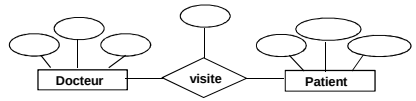
— Factorisation de la description des données et de leur comportement (contraintes d'intégrité …)

— Elimination de la redondance

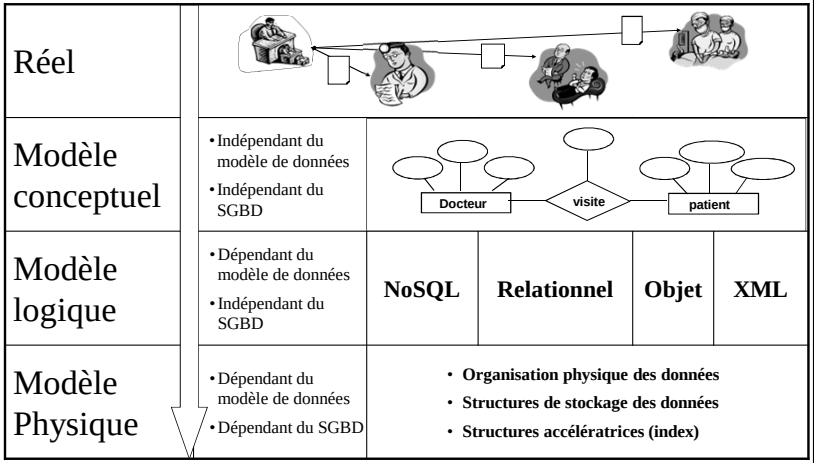
=> redondance coûteuse en place et source d'incohérence

=> redondance système reste nécessaire (fiabilité, performance, …)

Modélisation du réel

Réel		
Modèle conceptuel	• Indépendant du modèle de données • Indépendant du SGBD	
Modèle logique	• Dépendant du modèle de données • Indépendant du SGBD	**NoSQL** **Relationnel** **Objet** **XML**
Modèle Physique	• Dépendant du modèle de données • Dépendant du SGBD	• **Organisation physique des données** • **Structures de stockage des données** • **Structures accélératrices (index)**

Modèle logique



Modélisation Relationnelle (2)

Docteurs

Id-D	Nom	Prénom
1	Dupont	Pierre
2	Durand	Paul
3	Masse	Jean
….	……..	……

Visites

Id-D	Id-P	Id-V	Date	Prix
1	2	1	15 juin	250
1	1	2	12 août	180
2	2	3	13 juillet	350
2	3	4	1 mars	250

Prescriptions

Id-V	Ligne	Id-M	Posologie
1	1	12	1 par jour
1	2	5	10 gouttes
2	1	8	2 par jour
2	2	12	1 par jour
2	3	3	2 gouttes
….	….	….	……

Patients

Id-P	Nom	Prénom	Ville
1	Lebeau	Jacques	Paris
2	Troger	Zoe	Evry
3	Doe	John	Paris
4	Perry	Paule	Valenton
….	…….	…….	…….

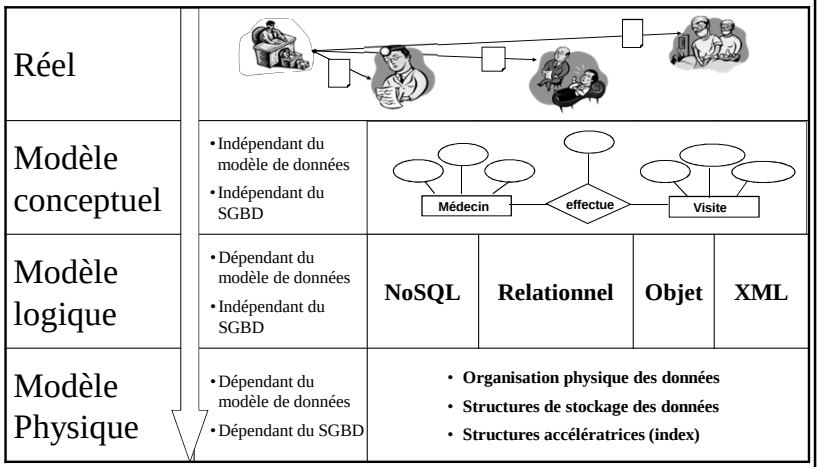
Médicaments

Id-M	Nom	Description
1	Aspegic 1000	……..
2	Fluisédal	……..
3	Mucomyst	……..
….	……..	……..

O2 - Indépendance Physique

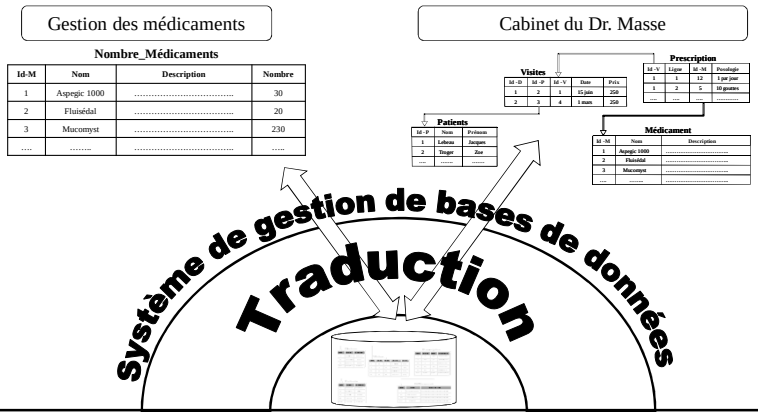
- **Indépendance des programmes d'applications vis à vis du modèle physique :**
  - Possibilité de modifier les **structures de stockage** (fichiers, index, chemins d'accès, …) sans modifier les programmes;
  - Ecriture des applications par des **non-spécialistes des fichiers** et des structures de stockage;
  - Meilleure **portabilité** des applications et **indépendance** vis à vis du matériel.

Modèle physique



O3 - Indépendance Logique

Les applications peuvent définir des **vues logiques** de la BD



O4 - Manipulation aisée

- Permettre, y compris à des non-informaticiens, de manipuler les données à partir de la seule connaissance du monde réel et de la modélisation qui en est faite
  - Mais les non-informaticiens font-ils des requêtes SQL ?
- La manipulation se fait via un langage déclaratif
  - La requête déclare l'objectif (assertion) sans décrire la méthode (programme)
  - Le langage suit une norme commune à tous les SGBD
  - SQL : Structured Query Langage
- Exemple

  Retrouver le nom et le n° de téléphone de tous les pédiatres

```
Select Nom, Tel
From Docteur
Where Specialite = 'Pédiatre'
```


O5 – Optimisation de requêtes

- Traduction automatique des requêtes déclaratives en programmes procéduraux (composition d'opérateurs élémentaires)
- Optimisation automatique de ces programmes
  - Exploitation des propriétés des opérateurs élémentaires
  - Gestion centralisée des chemins d'accès (index, hachage, …)
- Economie de l'astuce des programmeurs
  - milliers d'heures d'écriture et de maintenance de logiciels.
- Course aux performances mesurées en transactions par seconde (TPS) sur des "benchmark" standardisés (TPC).

O6 - Intégrité logique des données

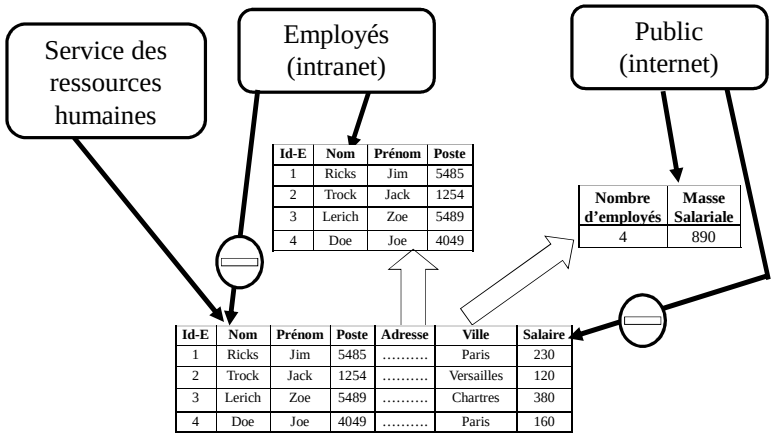
- Objectif : Détection automatique des mises à jour erronées
- Contrôle sur les données élémentaires
  - Contrôle de types: *Nom alphabétique*
  - Contrôle de valeurs: *Salaire mensuel entre 1 et 10k€*
- Contrôle sur les relations entre les données
  - Relations entre données élémentaires : *Prix de vente > Prix d'achat*
  - Relations entre objets : *Un électeur est inscrit sur une seule liste électorale*
- Avantages
  - simplification du code des applications
  - sécurité renforcée par l'automatisation
  - mise en commun des contraintes

O7: Confidentialité des données

— Objectif : garantir la confidentialité des informations et les protéger contre la dégradation
  - Données personnelles, procédé de fabrication, ...
  - Élément incontournable depuis la mise en oeuvre du RGPD !

— Plusieurs niveaux :
  - Authentification des usagers
  - Privilèges d'accès aux objets de la base
  - Chiffrement et hachage crytographique des données

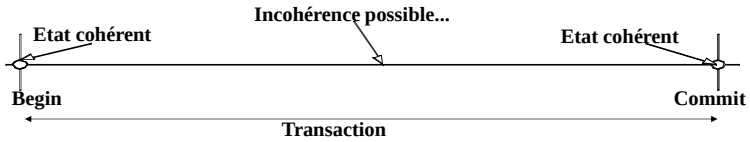
Confidentialité des données



O8 – Tolérance aux pannes

- **Motivations**
  - Transaction Failure : Contraintes d'intégrité, abandon de l'utilisateur
  - System Failure : Panne de courant, Crash serveur ...
  - Media Failure : Perte du disque
  - Communication Failure : Défaillance du réseau
- **Objectifs :**
  - Assurer l'**atomicité** des transactions
  - Garantir la **durabilité** des effets des transactions validées
- **Moyens :**
  - Journalisation : Mémorisation des états successifs des données
  - Mécanismes de reprise

Transaction



Begin
  CEpargne = CEpargne - 3000
  CCourant = CCourant + 3000
Commit T1

Atomicité et Durabilité

ATOMICITE

Begin
CEpargne = CEpargne - 3000
CCourant = CCourant + 3000
Commit T1

Panne

➔ **Annuler le débit !!**

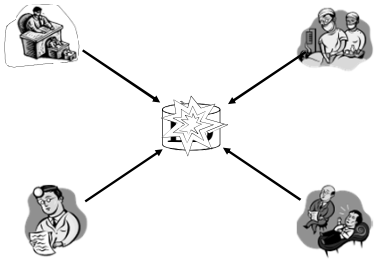
DURABILITE

Begin
CEpargne = CEpargne - 3000
CCourant = CCourant + 3000
Commit T1

Crash disque

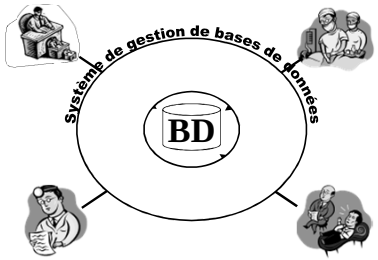
➔ **S'assurer que le virement a été fait !**

09 – Accès concurrents aux données



- Conflits d'accès ➔
  - **pertes de mises à jour**
  - **introduction d'incohérences**
  - **lectures non reproductibles**

Isolation et Cohérence



- Le SGBD gère les accès concurrents

➔ Chacun à l'*impression* d'être seul (Isolation)

➔ Cohérence conservée (Pas de maj conflictuelles)

➔ Sérialisabilité des exécutions

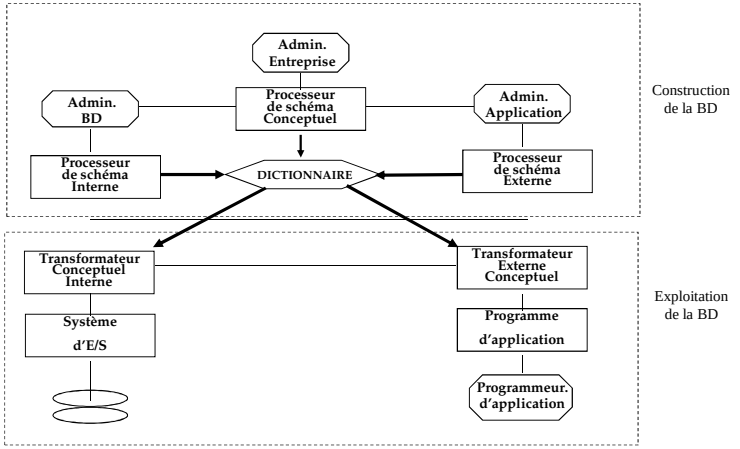
O10 - Standardisation

- L'approche bases de données est basée sur plusieurs standards
  - Langage SQL (SQL1, SQL2, SQL3)
  - Communication SQL CLI (ODBC / JDBC)
  - Transactions (X/Open DTP, OSI-TP)
- Force des standards
  - Portabilité des applications
  - Interopérabilité des systèmes

3. Architecture de référence des SGBD

- De nombreuses architectures fonctionnelles ont été proposées
- Ces architectures dépendent souvent du modèle de données utilisé
- ANSI/X3/SPARC est une architecture de référence mais sa normalisation a échouée.
- L'architecture ANSI/X3/SPARC repose sur un concept fondamental: la distinction de 3 niveaux de schémas

ANSI/X3/SPARC : principes



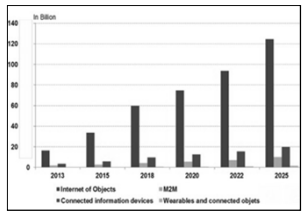
Dictionnaire et Méta-base sont synonymes

4. Applications traditionnelles des SGBD

- **OLTP** (On Line Transaction Processing)
  - Cible des SGBD depuis leur existence
  - Banques, réservation en ligne ...
  - Très grand nombre de transactions en parallèle
  - Transactions simples
- **OLAP** (On Line Analytical Processing)
  - Entrepôts de données, DataCube, Data Mining …
  - Faible nombre de transactions
  - Transactions très complexes

Mais les applications changent

- DB in the (very) large (Exa-base : 10^6 Tera)
  - NoSQL: sélections/màj simples sur tables géantes
    - ex: BigTable (Google), Cassandra (Facebook→Apache), Dynamo (Amazon) …
    - Performance/scalabilité au détriment de la cohérence
    - Une nouvelle forme d'OLTP
  - Big Data: analyse de gigantesques volumes de données faiblement structurées
    - Parallélisation/distribution massive des traitements : map/reduce, spark …
    - Une nouvelle forme d'OLAP
- DB in the 'very' small (Pico-base)
  - Des réseaux de capteurs à l'internet des objets
    - Gestion de données embarquée
    - Ressources très contraintes (RAM, énergie)
    - Calculs décentralisés et collaboratifs



Et la technologie aussi …

- Des SGBD disques aux SGBD en mémoire
  - Vers des performances temps-réel pour les hot-spot data

Barrette Samsung 128GB
pour serveurs extrêmes

- Des SGBD disques aux SGBD Flash
  - Une révolution dans la technologie du stockage

Condamné à tout réapprendre constamment ?

- Les usages et la technologie changent mais les principes fondamentaux restent
- Ces principes fondamentaux sont en nombre restreint
- Celui qui les maîtrise dans le contexte relationnel
  - Se rendra compte que les nouvelles architectures mettent souvent en œuvre de simples adaptions de ces principes
  - Il saura les comprendre rapidement et les administrer
- C'est le but de ce cours … *et du M2 DataScale* ☺