Dominante
Informatique

Systèmes de Gestion de Bases de Données (SGBD)

Introduction

http://www.bd.enst.fr/

saglio@enst.fr

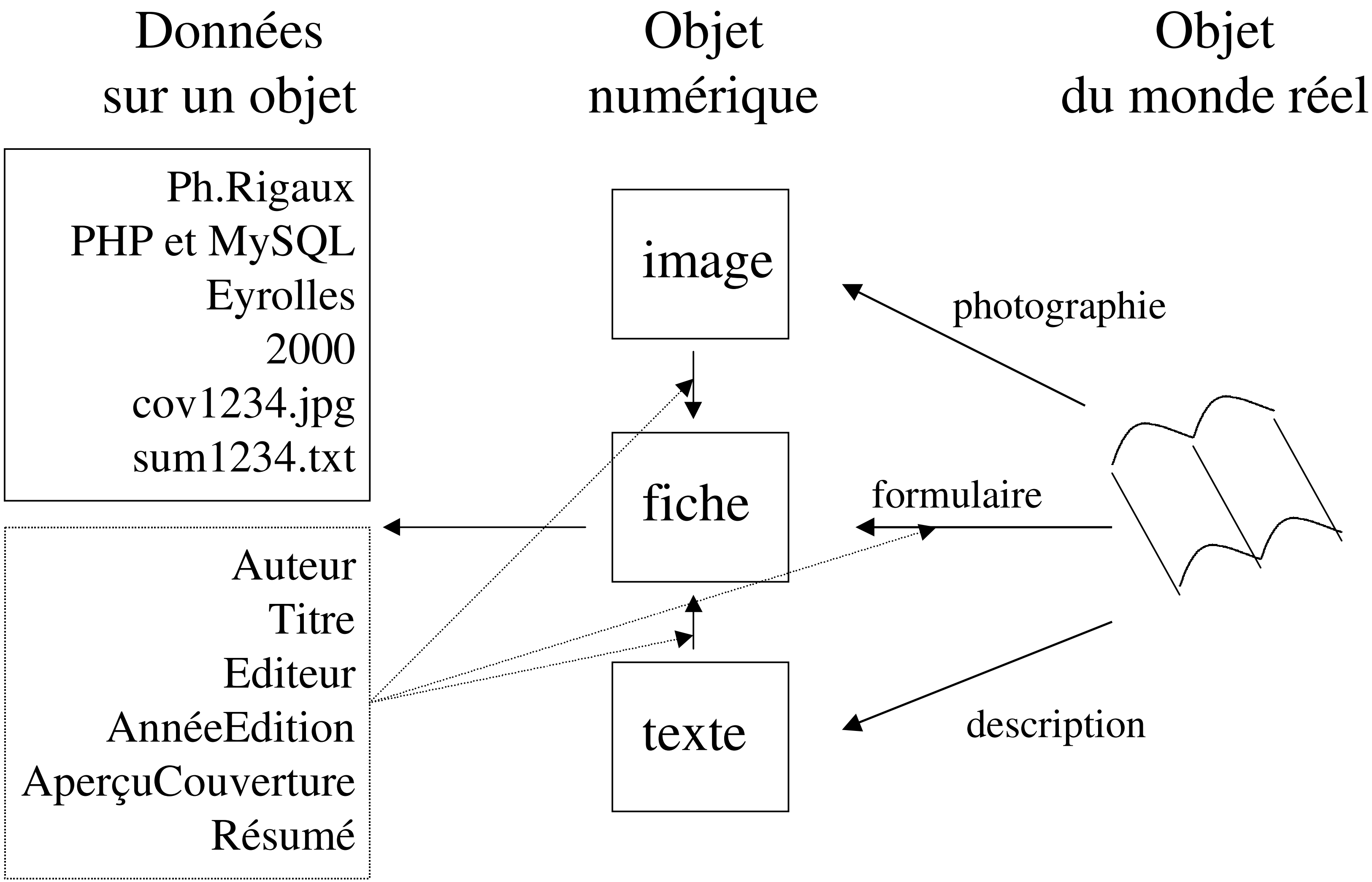
Données

- **nommées et typées** (= information)
  - ce que manipulent les programmes
  - en particulier l'input et l'output
  - types simples ou complexes (structures)
- **persistantes**
  - input et output doivent être en mémoire resp. avant et après l'exécution du programme,
  - et, au delà des arrêts du système, stockés en mémoire secondaire

Données et fichiers

- **fichiers binaires** (blocs de données, sons, images…): programmes de manipulation spécialisés
- **fichiers d'articles** (données de même noms et structures): procédures de manipulation standard (get_first, get_next, get_by_key_value, get_by_index_value…)
- **fichiers de texte** (caractères, mots, semi-structures textuelles): procédures de manipulation de texte, sans ou avec balises sans ou avec définitions de type de données (TXT, HTML, XML)

Données et objets



Base de données

- ensemble, en général assez **volumineux**, de données,
- représentant de façon **cohérente** un univers de classes et d'objets instances de ces classes,
- en **partage** pour tous les programmes / applications définis sur cet univers,
- **distribuées** sur différents fichiers éventuellement sur différents systèmes
- avec le **minimum de redondances**

Systèmes de BD

- systèmes offrant les services d'accès aux données d'une ou plusieurs bases : **définition**, de **manipulation** (consultation, mise à jour) et de **contrôle** (concession/retrait de droits)
- selon un ou plusieurs protocoles d'accès et/ou langages (LDD, LMD, LCD)
- de façon **performante**: temps de réponse optimisé
- en gérant correctement (ordonnancement) la **concurrence** d'accès sur les données ouvertes au partage (isolation)

Appli / SGBD *vs.* Appli spécifique

	Modéliser	Manipuler
C	**int, char, tableaux, struct imbrication, pointeurs**	**affectation, if, while, for**
SGBD relationnel	**int, char, tableaux et types abstraits**	requêtes SQL et procédures stockées

Différences :

- **volume de données : Go -> To**
- **nombre d'utilisateurs simultanés :100 -> 10 00**

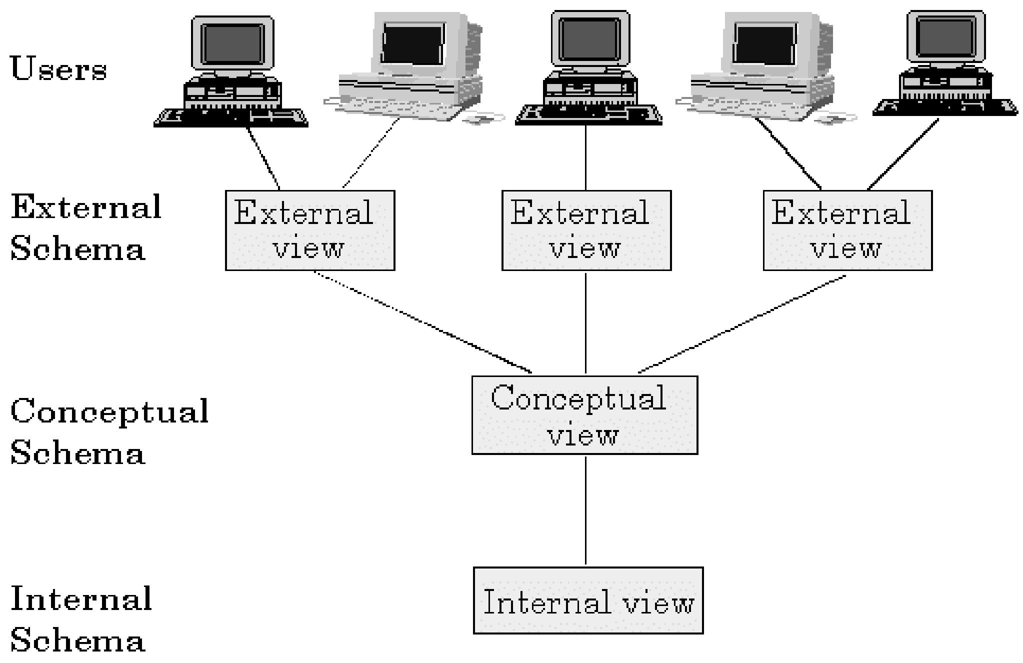
mais surtout **=> optimisation - intégrité - gestion des accès concurrents à la charge du SGBD**

Exemples de SBD

- Gestion d'annuaires : X500 ou LDAP
- Administration de ressources distribuées : CMIP ou SNMP
- Guichet automatique de banque
- Réservation de billets de transport
- E-commerce
- Méta-catalogues

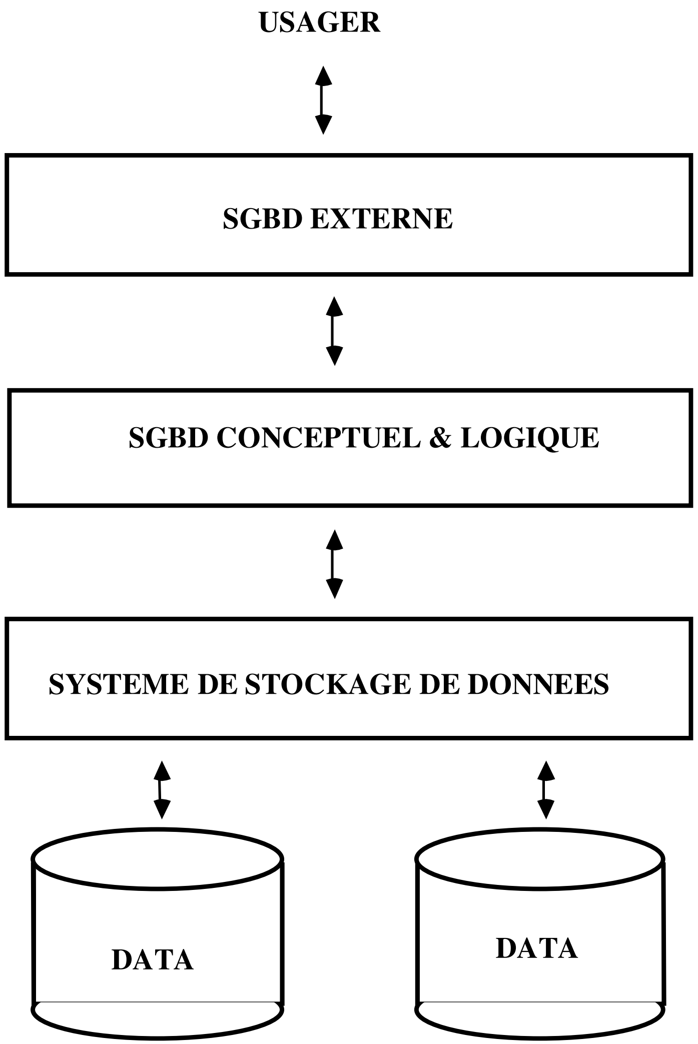
SGBD architecture logicielle

ANSI/SPARC Architecture



Distributed DBMS © 1998 M. Tamer Özsu & Patrick Valduriez Page 4. 3

Architecture fonctionnelle d'un SGBD



SGBD externe

assure l'analyse et l'interprétation des requêtes de l'usager, la mise en forme des résultats.

SGBD conceptuel et logique

Intégrité des données, optimisation de l'accès aux données, placement efficace des données

Système de stockage de données

Partage des données entre les différentes applications, sécurité des données, protection des données contre les pannes, gestion efficace de la mémoire secondaire

saglio@enst.fr

SGBD progiciels

- plus ou moins portables / systèmes
- plus ou moins inter-opérables / protocoles
- plus ou moins « dimensionnables » (« scalable »)
- Oracle, DB2, SQL Server, Informix, Ingres...

commercialisation des systèmes relationnels

Les grands SGBD relationnels ont l'architecture client-serveur

ORACLE	**ORACLE FRANCE**
DB2, SQL/DS	**IBM**
SQL Server	**Microsoft**
SQL Server	**SYBASE inc.**
INFORMIX	**INFORMIX**
INGRES	**COMPUTER ASSOCIATE**
POSTGRES	**Berkeley**

An 2001 :

4 SGBD relationnels se partagent - encore - l'essentiel du marché

Plan du cours - BD dom. informatique

- Introduction (c’est fait!)
- Modèle Entité-Relation (v. UML, modèles sémantiques, ontologies)
- Modèle et langages relationnels (SQL interprêté)
- Le développement d'application (SQL embedded, CLI, procédures stockées)
- TP1
- La conception et l'intégrité logique
- TP2
- Les services système : transactions, performances
- Les BD en 2001. Et après?

Bibliographie

Polycopié de Bases de Données - **J.P. Cheiney, P. Picouet, J.M. Saglio - Télécom Paris**

Maîtriser les bases de donnée **- G. Gardarin - ed Eyrolles**

Foundations of Databases **- Abiteboul, Hull, Vianu - Ed Addison Wesley**

Principles of Databases and Knowledge Based Systems **- J. D. Ullman - Computer Science Press.**

et

http://www.bd.enst.fr

Langages de Bases de Données

Un *langage de définition de données* **(LDD) permet de définir dans le SGBD les objets représentés à l'aide du modèle de données**

- Définition des entités
- Définition des associations
- Définition des contraintes d'intégrité
- Définition de vues

Un *langage de manipulation de données* **(LMD) qui est un langage de requêtes et non un langage de programmation**

- *Optimisation possible*

SGBD modèles et langage

- navigationnels

  (hiérarchique et en réseau - CODASYL)

  1965 à « hier »

- relationnels

  (classique, étendu, objet-relationnel - SQL3)

  1970 à ce jour

- objet

  (ODMG)

  1980 à ce jour

Les SGF : quand l'informatique débarque dans l'entreprise

1955-60 développement des systèmes de gestion de fichiers

les SGF **permettent de stocker et d'organiser les données sur mémoire secondaire**

différentes organisations :

séquentielles

séquentielles triées

indexées

séquentielles indexées

Objectif des SGF : Les performances

Exemple d'utilisation d'un SGF

Exemple : la base viticole

un fichier par entité :

vins, viticulteurs, buveurs

un fichier par association

productions, commandes

Manipulation **:**

ouvrir le fichier vins

ouvrir le fichier productions

prendre le premier vin

1) move nv dans ww-nv

lire dasn productions ww-nv

afficher le numero de producteur

prendre le vin suivant

retourner en 1

fin boucle

fermer les fichiers

saglio@enst.fr

1965 : Systèmes navigationnels

Première génération de Systèmes de Gestion de Bases de Données

- **Hiérarchiques (ex : IMS…)**
- **Réseaux (CODASYL, IDS2)**

Principe **:**

- **relier les données provenant de différents fichiers**

caractéristiques **:**

- **liens logiques entre les données :**
  - **hiérarchiques (modèle hiérarchique)**
  - **hiérarchiques ++ (modèle réseau)**
- **= liens physiques entre les enregistrements informatiques (pointeurs)**
- **langages dits 'navigationnels'**

Exemple d'un SGBD hiérarchique

amélioration du modèle hiérarchique :

• lien de cardinalité 1-N

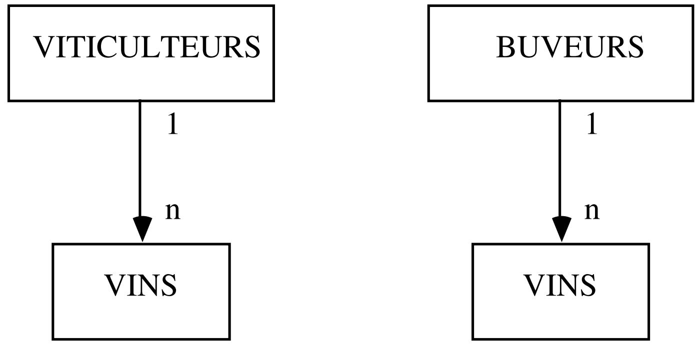
• un type de fils ne peut pas avoir plusieurs types de pères

Association m-n non modélisable

duplication de structures de données

accès par la racine de la hiérarchie

la base viticole représentée dans le modèle hiérarchique :



Exemple d'un SGBD réseau

amélioration du modèle hiérarchique :

• lien de cardinalité 1-N

• un type de fils peut avoir plusieurs types de pères

notions d'articles et d'ensembles

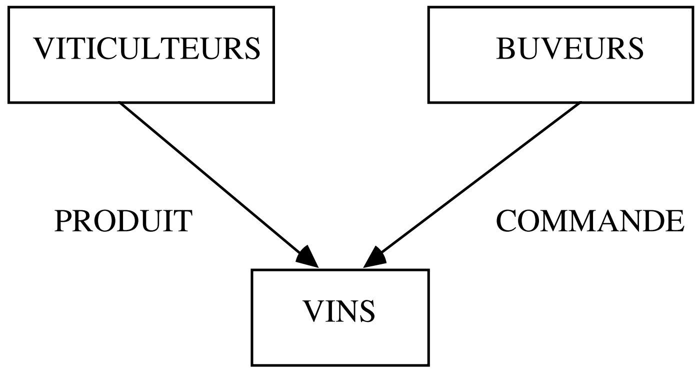
Utilisation d'articles intermédiaires pour représenter les associations m-n

Mais

• Pas d'ensemble récursif

• Pas de partage d'obje pour un même ensemble

la base viticole représentée dans le modèle réseau :



Les langages relationnels

Principe : Langages assertionnels vs. langages procéduraux

(quoi vs. comment)

SQL **: langage de définition et de manipulation de bases de données**

normes SQL1, SQL2, …-> SQL3

SQL est sur tous les SGBD, parfois même s'ils ne sont pas relationnels

c'est un argument de vente!

Exemple d'utilisation de SQL

Langage de definition de donnees :

create table vins (n°v number, cru char(20),
millesime number, n°vt number)

Langage de manipulation de donnees :

```
select n°v, cru
from vins
where  millesime = 1984

select distinct nom
from viticulteurs, vins, productions
where  viticulteurs.n°vt = productions.nvt
and productions.nv = vins.n°v
and cru = 'muscadet'

insert into vins values (8, 'beaujolais', 1981, 4)
```

+ **Embedded SQL**

Et le futur ?

SI guidés par les contraintes applicatives

Annuaires, SIG, ...

OLAP vs. OLTP

Nouvelles architectures de SGBD

BD parallèles

BD distribuées - homogènes v. hétérogènes

BD wrappers / web - semi-structuré

Nouveaux modèles / langages de SGBD

Bases de données temporelles, spatiales (ADT)

Bases de données orientées objets

Bases de données déductives