

LICENCES MATHÉMATIQUES ET INFORMATIQUE
3ÈME ANNÉE - FORMATION INITIALE ET PAR APPRENTISSAGE

BASES DE DONNÉES RELATIONNELLES
POLYCOPIÉ DE COURS - INTRODUCTION

Maude Manouvrier

La reproduction de ce document par tout moyen que ce soit est interdite conformément aux articles L111-1 et L122-4 du code de la propriété intellectuelle.

Table des matières

1	Introduction	2
1.1	Définitions	2
1.2	SGBD vs Gestionnaire de fichiers	2
1.3	SGBD : Concepts généraux	3
1.3.1	Principaux composants	3
1.3.2	Principales fonctionnalités	3
1.3.3	Abstraction des données	4
1.3.4	Instances et schéma	5
	Bibliographie	6

Chapitre 1

Introduction

1.1 Définitions

Une **base de données** est une collection d'informations ou de données qui existent sur une longue période de temps [21]. Ces données décrivent les activités d'une ou plusieurs organisations [18].

Par exemple :

- Une base de données d'université peut contenir de l'information sur les étudiants (leurs coordonnées, leur cursus, leurs notes ...), sur les enseignements (les coordonnées de l'enseignant, le nombre d'heures, la formation ...) et sur les salles (le numéro, les dates de réservations ...).
- Une base de données de compagnie aérienne contiendra de l'information sur les clients (leurs coordonnées), les vols (date et heure de départ/d'arrivée, destination ...) et les réservations.
- Une base de données bancaire contient par exemple de l'information sur les clients (coordonnées, emploi, salaire ...), leurs comptes (solde), les transactions effectuées sur les comptes (retrait, dépôts ...).

Mais attention à l'abus de langage :

Une base de données est un *ensemble de données modélisant les objets d'une partie du monde réel et servant de support à une application informatique* [9].

Ces données sont gérées par un **Systèmes de Gestion de Bases de Données** (SGBD) (ou *DataBase Management Systems* - DBMS) [21] :

Un SGBD est un ensemble de logiciels systèmes permettant aux utilisateurs *d'insérer, de modifier, et de rechercher efficacement des données spécifiques dans une grande masse d'informations* (pouvant atteindre plusieurs milliards d'octets) *partagée par de multiples utilisateurs* [9].

1.2 SGBD vs Gestionnaire de fichiers

Prenons un exemple [18] :

On suppose qu'une entreprise possède 500Go de données sur ses employés, ses départements, sa production, etc.

- Les données sont accédées de manière concurrente par plusieurs employés de l'entreprise (le directeur, le comptable, les secrétaires, les responsables de département ...).
- Les recherches (requêtes) sur les données doivent être rapides.
- Les modifications réalisées par différentes personnes ne doivent pas rendre les données incohérentes.
- L'accès à certaines données doit être restreint à quelques personnes (ex. les salaires).

Si les données sont uniquement stockées dans des fichiers gérés par un gestionnaire de fichiers, plusieurs problèmes se posent :

1. Il n'est pas possible de stocker 500Go de données en mémoire. Il faut donc stocker ces données sur disque et placer en mémoire les parties nécessaires à l'application en fonction des opérations à effectuer.
2. Il faut trouver un moyen d'identifier chaque donnée (un employé, un département, etc.) sur le disque ou en mémoire.
3. Un programme est nécessaire pour chaque question. Ce programme est complexe car il va rechercher de l'information dans un grand volume de données.
4. Il faut protéger les données des incohérences dues à des modifications concurrentes.
5. Il faut pouvoir retrouver les données dans un état cohérent après une panne du système pendant des mises à jour.
6. Il faut sécuriser les données et permettre leur accès par un login et un mot de passe et gérer les droits de chaque utilisateur pour chaque type de données.

Un SGBD permet de simplifier toutes ces tâches.

1.3 SGBD : Concepts généraux

1.3.1 Principaux composants

Les principaux composants d'un SGBD sont : le *système de gestion de fichiers*, le *gestionnaire de requêtes* et le *gestionnaire de transactions* [21].

Le **Système de gestion de fichiers** a la charge du stockage et de l'accès des fichiers à la fois en mémoire secondaire (sur disque) et en mémoire principale. C'est lui qui place les données en mémoire à partir du disque et inversement. Un SGBD utilise des **index** pour accélérer l'accès aux données. Un index est une structure de données qui permet de trouver rapidement une ou plusieurs données en fonction d'une partie de leur valeur. Un index en base de données pour faire simple fonctionne un peu comme un index situé à la fin d'un ouvrage, excepté qu'il ne donne pas un numéro de page de livre en fonction d'un mot clé, mais l'adresse de l'information sur le disque, en fonction d'une partie de la valeur de l'information.

Le **gestionnaire de requêtes** traduit les requêtes d'interrogation ou de mise à jour, exprimées par l'utilisateur par un langage de haut niveau (SQL - *Structured Query Language*) en une séquence d'opérations sur les données stockées. Une des principales activités de ce gestionnaire est *l'optimisation de requêtes* qui consiste à trouver le meilleur algorithme (plan d'exécution des opérations de la séquence) pour répondre à la requête.

Les actions (requêtes d'interrogation, mises à jour, etc.) devant être exécutées ensemble séquentiellement sont regroupées dans une **transaction**. Par exemple, le virement d'une somme X d'un compte en banque A vers un compte en banque B est une transaction composée des actions suivantes : lire le solde du compte A, si le solde est supérieur à X alors supprimer X du solde de A, ajouter X au solde du compte B. Une transaction représente un tout, dans le sens où toutes ses opérations (ici l'opération de décrémentation du solde de A et celle de l'incrément du solde de B) seront exécutées ou aucune, même en cas de panne au milieu de l'exécution. Le **gestionnaire de transactions** permet de gérer les transactions et notamment permet à plusieurs transactions de s'exécuter de manière concurrente.

1.3.2 Principales fonctionnalités

Les principales fonctionnalités d'un SGBD sont les suivantes¹ :

1. Informations issues de <http://www-inf.int-evry.fr/COURS/BD/glossaire.html>

- **Contrôle de la redondance d'information** : La redondance d'informations pose différents problèmes (coût en temps, coût en volume et risque d'incohérence entre les différentes copies). Un des objectifs des bases de données est de contrôler cette redondance, voire de la supprimer ou de la limiter. On évitera donc au maximum de stocker plusieurs fois la même information lorsqu'il est possible de ne la stocker qu'une seule fois.
- **Partage des données** : Une base de données doit permettre d'accéder la même information par plusieurs utilisateurs en même temps. Le SGBD doit inclure un mécanisme de contrôle de la concurrence pour éviter par exemple qu'on puisse lire une information qu'on est en train de mettre à jour.
- **Gestion des autorisations d'accès** : Une base de données étant multi-utilisateurs, se pose le problème de la confidentialité des données. Des droits doivent être gérés sur les données, droits de lecture, mise à jour, création, ... qui peuvent différer d'un utilisateur à un autre.
- **Vérification des contraintes d'intégrité** : Un schéma de base de données se compose d'une description des données et de leurs relations ainsi que d'un ensemble de contraintes d'intégrité. Une contrainte d'intégrité est une propriété de l'application à modéliser qui renforce la connaissance que l'on en a. On peut classer les contraintes d'intégrité, en contraintes structurelles (un employé a un chef et un seul par exemple) et contraintes dynamiques (un salaire ne peut diminuer). Les SGBD commerciaux supportent automatiquement un certain nombre de contraintes structurelles, mais ne prennent pas en compte les contraintes dynamiques (elles doivent être codées dans les programmes d'application).
- **Sécurité et reprise après panne** : Une base de données est souvent vitale dans le fonctionnement d'une organisation, et il n'est pas tolérable qu'une panne puisse remettre en cause son fonctionnement de manière durable. Les SGBD fournissent des mécanismes pour assurer cette sécurité. Le premier mécanisme est celui de transaction qui permet d'assurer un comportement atomique à une séquence d'actions (elle s'effectue complètement avec succès ou elle est annulée). Ce mécanisme permet de s'affranchir des petites pannes (style coupure de courant). En ce qui concerne les risques liés aux pannes disques, les SGBD s'appuient sur un mécanisme de journalisation qui permet de régénérer une base de données automatiquement à partir d'une version de sauvegarde et du journal des mouvements.

1.3.3 Abstraction des données

Afin de faciliter l'accès aux données, les SGBD utilisent des structures complexes pour représenter les données dans la base de données. Ainsi, pour faciliter l'utilisation des SGBD [19], les données sont vues selon trois niveaux d'abstraction définies par la norme ANSI/SPARC :

1. Le **Niveau interne ou physique** : Il s'agit du plus bas niveau. Il est décrit par un **schéma physique** qui indique *comment* (avec quelles structures de données) sont stockées les données physiquement.
2. Le **Niveau logique ou conceptuel** : Ce niveau est décrit par un **schéma conceptuel** qui indique quelles sont les données stockées et quelles sont leurs relations. Le schéma conceptuel décrit la structure de la base indépendamment de son implémentation. Ce niveau est utilisé par l'administrateur de données qui décide du *quoi*, c'est-à-dire de ce qui est stocké dans la base de données.
3. Le **Niveau externe ou vue** : Ce niveau est propre à chaque type d'utilisateur. Il correspond à la partie des données de la base visibles par l'utilisateur. Il est décrit par un ou plusieurs **schémas externes**. Chaque schéma externe donne une vue sur le schéma conceptuel à une classe d'utilisateurs.

Ce découpage facilite la maintenance des applications [10] : (1) il permet de dissocier autant que possible les applications des contraintes liées au matériel ou au système d'exploitation (seul le niveau physique en dépend), (2) la séparation entre les niveaux externes et conceptuels permet de protéger les applications envers les modifications ou enrichissement du schéma conceptuel.

1.3.4 Instances et schéma

Le contenu d'une bases de données change constamment [19] : des données sont insérées, supprimées ou modifiées. la collection d'information stockée dans une base de données à un instant donné est appelée une **instance** de la base de données.

La description totale des données de la base est appelée **schéma** de la base de données. Le schéma correspond à un ensemble de définitions exprimées par un **langage de description de données** (DDL - *Data Definition Language*). Ce langage supporte un modèle (ensemble de règles et de concepts) et permet de décrire les données de la base de manière assimilable à la machine [9].

Les données sont ensuite manipulées par les utilisateurs à l'aide d'un **langage de manipulation de données** (DML - *Data Manipulation Language*).

Bibliographie

- [1] D. Austin, *Using Oracle8TM*, Simple Solutions - Essential Skills, QUE, 1998, ISBN : 0-7897-1653-4
- [2] R. Chapuis, *Oracle 8*, Editions Dunes et Laser, 1998, ISBN : 2-913010-07-5
- [3] P. Chen, *The Entity-Relationship Model-Toward a Unified View of Data*, ACM Transactions on Database Systems, Vol. 1, No. 1, March 1976, Pages 9-36, [http ://www.csc.lsu.edu/~chen/pdf/erd.pdf](http://www.csc.lsu.edu/~chen/pdf/erd.pdf)
- [4] T. Connolly, C. Begg et A. Strachan, *Database Systems - A practical Approach to Design, Implementation and Management*, Addison-Wesley, 1998, ISBN : 0-201-34287-1, disponible à la BU 055.7 CON
- [5] C.J. Date, *Introduction aux bases de données*, 6ème édition, Thomson Publishing, 1998, ISBN : 2-84180-964-1, disponible à la BU 005.7 DAT
- [6] R. Elamsri et S.B. Navathe, *Fundamentals of Database Systems*, 3ème édition, Addison Wesley-disponible à la BU 005.7 ELM
- [7] P. Delmal, *SQL2 - Application à Oracle, Access et RDB*, 2ème édition, Bibliothèque des Universités - Informatique, De Boeck Université, 1988, ISBN : 2-8041-2995-0, disponible à la BU 005.74 SQL
- [8] S. Feuerstein, B. Pribyl et C. Dawes, *Oracle PL/SQL - précis et concis*, O'Reilly, 2000, ISBN : 2-84177-108-3
- [9] G. Gardarin, *Bases de Données - objet & relationnel*, Eyrolles, 1999, ISBN : 2-212-09060-9, disponible à la BU 005.74 GAR
- [10] R. Grin, *Introduction aux bases de données, modèle relationnel*, Université Sophia-Antipolis, jan. 2000
- [11] R. Grin, *Langage SQL*, Université Sophia-Antipolis, jan. 2000
- [12] G. Gardarin et O. Gardarin *Le Client-Serveur*, Eyrolles, 1999, disponible à la BU
- [13] H. Garcia-Molina, J.D. Ullman et J. Widom, *Database System Implementation*, Prentice Hall, 2000, ISBN : 0-13-040264-8, disponible à la BU 005.7 GAR
- [14] H. Garcia-Molina, J.D. Ullman et J. Widom, *Database Systems - The Complete Book*, Prentice Hall, 2002, ISBN : 0-13-031995-3
- [15] S. Krakowiak, *Gestion Physique des données*, Ecole Thématique "Jeunes Chercheurs" en Base de Données, Volume 2, Port Camargue, mars 1997
- [16] D. Lockman, *Oracle8TM Développement de bases de données*, Le programmeur - Formation en 21 jours, Editions Simon et Schuster Macmillan (S&SM), 1997, ISBN : 2-7440-0350-6, disponible à la BU 005.74 ORA
- [17] P.J. Pratt, *Initiation à SQL - Cours et exercices corrigés*, Eyrolles, 2001, ISBN : 2-212-09285-7
- [18] R. Ramakrishnan et J. Gehrke, *Database Management Systems*, Second Edition ; McGraw-Hill, 2000, ISBN : 0-07-232206-3, disponible à la BU 055.7 RAM
- [19] A. Silberschatz, H.F. Korth et S. Sudarshan, *Database System Concepts*, Third Edition, McGraw-Hill, 1996, ISBN : 0-07-114810-8, disponible à la BU 005.7 DAT
- [20] C. Soutou, *De UML à SQL - Conception de bases de données*, Eyrolles, 2002, ISBN : 2-212-11098-7
- [21] J.D. Ullman et J. Widom, *A first Course in Database Systems*, Prentice Hall, 1997, ISBN : 0-13-887647-9, disponible à la BU 005.7 ULL