

Chapitre I : Introduction

I. Qu'est-ce qu'une base de données ?

I. 1. Introduction

Une base de données est un ensemble d'informations structurées définissant un système d'information en vue de son stockage dans un support de masse pour faciliter son exploitation. Elle modélise une situation réelle à laquelle elle doit être fidèle selon un certain niveau d'abstraction. Son objectif principal est donc de structurer et sauvegarder des informations d'un système pour y accéder facilement en cas de besoin.

Elle est constituée de relations représentées sous forme de tables dont les colonnes sont des attributs et les lignes des n -uplets (n étant le nombre d'attributs de la base). Elle est modélisée et administrée via un système de gestion de base de données (SGBD).

I. 2. Instance et schéma d'une base de données

I. 2. 1. Schéma d'une base de données

La structure d'une base de données bien conçue change très peu durant son cycle de vie. Elle est constituée de relations dont chacune est décrite par des attributs, une clé primaire et éventuellement une/des clé(s) étrangère(s). Le domaine de chaque attribut doit également être spécifié. Ces informations (nom de relation, attributs, domaines des attributs, clé primaire, clés étrangères) décrivent le schéma d'une relation. L'ensemble des schémas des relations qui composent une base de données donne son schéma.

I. 2. 2. Instance d'une base de données

Le contenu d'une base de données change constamment durant son exploitation. Des données sont ajoutées, modifiées et/ou supprimées fréquemment par les utilisateurs. Il est donc nécessaire de connaître ce que contient la base à chaque instant. L'instance d'une table est constituée de l'ensemble des n -uplets qui y sont enregistrés à un instant donné. L'instance d'une base de données est constituée de l'ensemble des instances de ses tables à un instant donné.

II. Qu'est-ce qu'un SGBD ?

II. 1. Introduction

Un SGBD est une application permettant de gérer des bases de données. Pour ce faire il contient des modules permettant d'implémenter des bases de données, de les sécuriser, d'y insérer des informations, de les exploiter, etc. L'exploitation d'une base consiste en grande partie à y effectuer des mises à jour et à y chercher des informations respectant un ou plusieurs critères de recherche. Un SGBD est composé essentiellement de trois grandes parties que sont le gestionnaire de fichiers, le gestionnaire de transactions et le gestionnaire de requêtes.

II. 2. Les composants d'un SGBD

II. 2. 1. Le gestionnaire de fichiers

Le rôle du gestionnaire de fichiers est d'assurer la gestion des données sur les supports de stockage de masse. Il permet de sauvegarder les données sous forme de tables avec des lignes (n_uplets) et des colonnes (attributs). Il donne la possibilité de lier les différentes tables pour permettre l'implémentation des liens entre elles. Il assure donc la création et la modification de la base et de ses tables en plus de l'enregistrement (sauvegarde) des données, de leurs mises à jour et de leur recherche sur les supports de stockage. Il gère également la restriction à l'accès des données suivant les privilèges des utilisateurs. Il est le seul gestionnaire capable d'accéder aux données enregistrées sur le disque dur.

II. 2. 2. Le gestionnaire de transactions

II. 2. 2. 1. Qu'est-ce qu'une transaction ?

Une transaction est une suite d'opérations indivisible vérifiant les propriétés **ACID** (Atomicité, Cohérence, Isolation, Durabilité).

- ✓ **Atomicité** : Toutes les opérations composant une transaction doivent être entièrement exécutée ou totalement annulée. C'est la loi du tout ou rien.
- ✓ **Cohérence** : Une base de données devant être toujours cohérente, une transaction validée doit laisser la base de données dans un état cohérent.
- ✓ **Isolation** : Les modifications apportées à la base par les opérations d'une transaction en cours ne doivent être visibles qu'après leur validation. Ainsi, une donnée en cours de modification ne doit pas être accessible.

- ✓ **Durabilité** : Lorsqu'une transaction est exécutée et validée, les modifications apportées par ses différentes opérations sur la base doivent y rester.

Si une partie des opérations d'une transaction est exécutée et que pour une raison ou une autre le système n'arrive pas à exécuter le reste, les effets de ces opérations doivent être annulés.

II. 2. 2. Rôle du gestionnaire de transactions

Le rôle du gestionnaire de transactions dans un SGBD est d'assurer l'exécution correcte des transactions lancées par les utilisateurs. Il assure leur exécution en veillant à ce que les différentes tâches de chacune d'elles suivent le bon ordre, tout en vérifiant le respect des propriétés ACID. Il coordonne aussi l'exécution des différentes transactions lancées simultanément.

II. 2. 3. Le gestionnaire de requêtes

Il sert d'intermédiaire entre les données sauvegardées dans la base et les utilisateurs. Il traduit les requêtes d'interrogation et de mise à jour exprimées par les utilisateurs dans un langage de haut niveau (SQL par exemple) en une séquence d'opérations basiques sur les informations enregistrées dans la base. Ces requêtes sont souvent intégrées dans des programmes applicatifs pouvant accéder à la base.

III. Dépendance fonctionnelle

Il y'a dépendance fonctionnelle entre deux attributs d'une entité si à chaque valeur de l'un correspond une et une seule valeur de l'autre.

Par exemple connaissant le nom du bâtiment et le numéro d'une salle, on connaît sa capacité, si oui ou non elle climatisée, son type (Salle de TP, TD, etc.), etc.

Bâtiment, Salle → Capacité, Climatisé, Type.

L'ensemble contenant la/les propriété(s) de gauche (Bâtiment, Salle) est appelé source et celui contenant la/les propriété(s) de droite (Capacité, Climatisé, Type) est appelé but. Une dépendance fonctionnelle dont la source est composée d'un seul attribut est appelée DF simple. Si la source est composée de plusieurs attributs on a une dépendance fonctionnelle composée.

III. 1. Dépendance directe

Une dépendance fonctionnelle $X \rightarrow Y$ est dite directe s'il n'existe pas un attribut Z tel que les dépendances fonctionnelles $X \rightarrow Z$ et $Z \rightarrow Y$ soient correctes.

III. 2. Dépendance élémentaire

Une dépendance fonctionnelle est dite élémentaire si sa source ne contient pas d'attribut superflu. Si $X, Y \rightarrow Z$ et $X \rightarrow Z$ alors Y est superflu dans la dépendance $X, Y \rightarrow Z$ qui n'est pas élémentaire.

IV. Objectifs des bases de données

Les bases de données jouent un rôle très important dans la structuration, la sauvegarde et l'exploitation des données d'un système d'information. Elles permettent de stockées des données avec le format le plus approprié et d'y accéder le plus aisément possible en cas de besoin. Elles sont très utiles lorsque l'on cherche des données satisfaisant une condition dans un ensemble d'information assez volumineux. Pour ce faire, une base de données peut centraliser les données ou les répartir sur plusieurs sites différents. Il y a aussi la possibilité de répliquer les mêmes données sur des sites différents. L'administrateur doit veiller à ce qu'il n'y ait ni redondance, ni incohérence, ni perte de mise à jour. Il doit également mettre en place les politiques de sécurité tout en faisant de sorte d'avoir les meilleures performances possibles. Le SGBD doit pouvoir autoriser l'accès simultané aux informations par plusieurs utilisateurs tout en sauvegardant la cohérence et en assurant la sécurité de la base.

V. Conception d'une base de données

La conception d'une base de données est une étape très importante et délicate qui demande beaucoup d'attentions et de créativité. Elle commence par la mise en place du cahier des charges s'il n'existe pas encore. Il contient les informations importantes fournies par le client et qui doivent être prises en compte tout au long de la conception de la base. A partir du cahier des charges, le schéma de la base est créé.

V. 1. Mise en place du cahier des charges

Le cahier des charges est le document contenant les informations qui seront stockées dans la base, les traitements qui s'y porteront, les utilisateurs qui y accéderont, les privilèges de chacun d'eux, les règles de gestion, etc. Pour cela des interviews sont nécessaire car toutes les

informations du cahier des charges doivent venir du client et de ses collaborateurs. Le concepteur de la base se chargera d'exploiter ces informations pour créer le schéma de la base de données. Les principales questions que l'on peut poser lors de l'interview sont :

- Quelles sont les données que l'on doit enregistrer dans la base ?
- Quels sont les traitements qui doivent porter sur les données ?
- Quelles sont les règles de gestion de la structure concernée ?
- Où est ce que les données seront stockées ?
- Qui est ce qui pourra accéder aux données de la base ?
- Pour chaque utilisateur, quels sont les traitements qu'il peut lancer et sur quelle(s) partie(s) de la base ?
- Etc.

V. 2. Analyse des données

Elle consiste à faire une étude approfondie du contenu du cahier des charges pour structurer les données selon leurs caractéristiques et leur importance. Cette structuration se fait de manière descendante. On commence par identifier les entités, ensuite leurs attributs et leurs domaines, l'identifiant de chaque entité, etc. Les répétitions de données ainsi que les incohérences sont à éviter au maximum. L'identifiant de chaque entité est choisi et les liens entre les différentes entités sont établis. Ces liens permettent d'inter-relier les différentes entités pour faciliter la recherche d'informations basée sur plusieurs entités.

V. 3. Création du schéma d'une base de données

Après une bonne analyse des informations contenues dans le cahier des charges, les entités, leurs attributs, leurs identifiants, les associations qui les lient sont clairement identifiés. Le modèle Entité/Association est donc complété avec le choix des cardinalités des associations et de leurs attributs éventuels.

Le modèle Entité/Association n'est pas directement implémenté dans le SGBD. Il est transformé en modèle relationnel en tenant compte principalement des associations et de leurs cardinalités. Le schéma de la base de données est obtenu en ajoutant les domaines des attributs au modèle relationnel.

VI. Architecture d'une base de données

Une base de données est manipulée le plus souvent par les utilisateurs via des interfaces utilisateurs d'applications informatiques. Les utilisateurs accèdent ainsi au serveur de données qui peut être très distant du site où ils se trouvent. Les composants logiciels peuvent être exécutés par une même machine ou par plusieurs machines différentes.

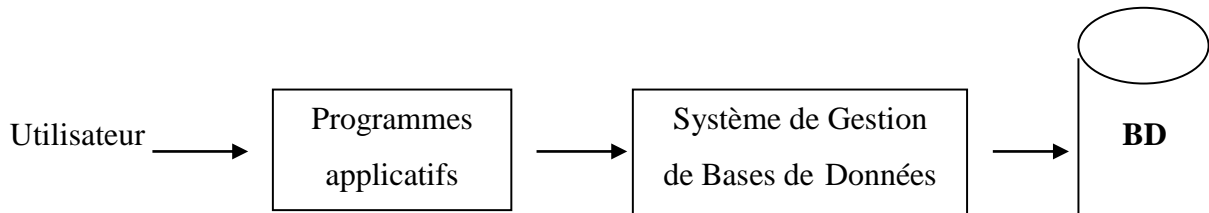


Figure 1 : Environnement d'utilisation d'une base de données

VI. 1. Architecture 2-tiers

On parle d'architecture 2-tiers si une seule machine (un seul serveur) se charge de l'exécution complète des demandes des utilisateurs. Elle sert alors de serveur de données et de serveur de traitement. Les logiciels (programmes applications) qui accèdent à la base sont exécutés par la machine sur laquelle la base de données est implémentée.

VI. 2. Architecture 3-tiers

Dans une architecture 3-tiers, le serveur de données est distinct du serveur de traitement. L'utilisateur envoie sa requête au serveur de traitement, qui à son tour passe par le serveur de données pour obtenir les informations sur lesquelles doivent porter les traitements.