

1. Historique des SGBD

Les SGBD ont déjà une longue histoire. Les années 60 ont connu un premier développement des systèmes de fichiers qui tendent essentiellement à faire apparaître les mémoires secondaires comme idéales, partagées, banalisées, directement adressables, de capacité infinie. Ceux-ci, après avoir évolué, composent aujourd'hui le cœur des SGBD.

Le milieu des années 60 a vu la naissance de la première génération de SGBD ; celle-ci a été marquée par la séparation de la description des données des programmes d'application et l'avènement de langages d'accès navigationnels, c'est-à-dire permettant de se déplacer dans des structures de type graphe et d'obtenir, un par un, des groupes de données appelés articles. Cette première génération, caractérisée par les systèmes obéissant aux premières recommandations du CODASYL est basée sur des modèles d'accès, c'est à dire des modèles de données visant essentiellement à optimiser les méthodes de placement des données sur les mémoires secondaires afin de réduire les temps d'accès.

La deuxième génération de SGBD a grandi dans les laboratoires depuis 1970, à partir du modèle relationnel. Elle vise essentiellement à enrichir le SGBD externe afin de simplifier l'accès aux données pour les utilisateurs externes. Ainsi, la deuxième génération offre des langages assertionnels basés sur la logique, permettant de spécifier les données que l'on souhaite obtenir sans dire comment les accéder.

La troisième génération de SGBD basée sur des langages d'accès plus puissants et plus naturels, supportant des types de données plus variés, incluant des possibilités de déduction et de répartition.

Ce chapitre a pour objet de définir les concepts de base de données (BDD) et de système de gestion de base de données (SGBD).

2. Concept de base de données

Une base de données est faite pour enregistrer des faits, des événements qui surviennent dans la vie d'un organisme et pour les restituer à la demande ou bien pour tirer des conclusions en rapprochant plusieurs faits élémentaires les uns aux autres.

Par exemple, une université a besoin pour sa gestion d'archiver des informations sur les enseignants, les étudiants, les enseignements, les conditions d'inscriptions à une formation... etc. Mais elle a aussi besoin de connaître les associations qui existent entre ces différentes informations : l'inscription qui associe l'étudiant avec la formation qu'il souhaite, les modules donnés par un enseignant...etc.

Vu sous l'angle des anciennes méthodologies de conception, chaque volet de ce cas d'exemple nécessiterait une nouvelle application avec ses propres fichiers et ses propres programmes. La création d'une base de données va à l'encontre de cette façon de faire : elle rend possible la centralisation, la coordination, l'intégration et la diffusion de l'information archivée.

D'une manière générale, on peut définir une BDD de cette manière :

Une base de données est un ensemble structuré de données enregistrées sur des supports de stockage pour satisfaire simultanément plusieurs utilisateurs de façon sélective en un temps opportun.

Une BDD est construite selon un modèle de données, il existe trois grandes catégories de modèles de données : le modèle réseau, le modèle hiérarchique et le modèle relationnel.

3. Notion de SGBD

Un SGBD est un ensemble de logiciels et matériels associés permettant aux utilisateurs d'interagir avec une BDD. Il permet principalement de créer, d'insérer, de modifier et de rechercher efficacement des données spécifiques dans une grande masse d'informations partagées par tous les usagers. Pour aboutir à ce résultat, l'utilisateur décrit en termes abstraits ce qu'il souhaite faire sur les données, laissant le soin au système d'effectuer les tâches de recherche en fonction de la représentation et de l'organisation des données sur les supports physiques.

Par exemple, vous arrivez dans une ville inconnue et vous prenez un taxi pour vous rendre chez X c'est le chauffeur de taxi qui, pour répondre à votre requête, prendra le chemin le plus court pour accéder chez X.

Un SGBD se compose de trois couches successives de fonctions, empilées depuis les mémoires secondaires vers les usagers :

1. Système de gestion de fichier (SGF) : permet de stocker les données sur les mémoires secondaires.
2. SGBD interne : permet la gestion des liens entre les données stockées dans les fichiers. Qui devient invisible aux programmes d'applications ?
3. SGBD externe : assure l'analyse et l'interprétation des requêtes des usagers, et la mise en forme des données échangées avec le monde extérieur.

La figure I.1 illustre ces trois couches de fonctions qui constituent un SGBD.

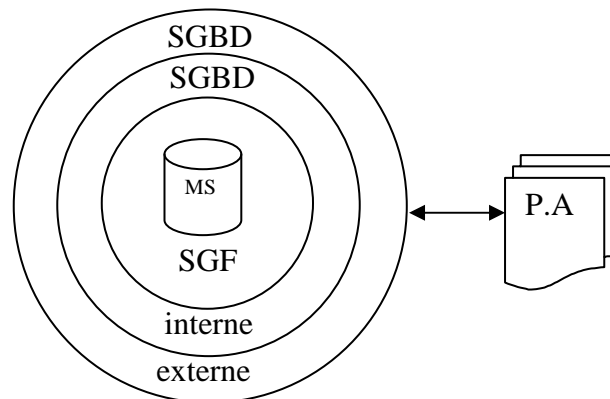


Fig.I.1-les trois couches de fonctions qui constituent un SGBD.

P.A :programme d'application

M.S :mémoires secondaires

4. Objectifs d'un SGBD

Les objectifs principaux des systèmes de gestion de base de données sont résumés dans les points suivants :

- Indépendance physique
- Indépendance logique
- Manipulation des données par des non-informaticiens
- Efficacité des accès aux données
- Administration cohérente des données
- Non redondance des données
- Cohérence des données
- Partageabilité des données
- Sécurité des données

a. Indépendance physique

C'est un objectif essentiel des SGBD. Il s'agit de permettre de réaliser l'indépendance des structures de stockage aux structures de données du monde réel. Cela consiste à pouvoir définir l'assemblage des données élémentaires entre elles dans le système informatique indépendamment de l'assemblage réalisé dans le monde réel, en tenant compte seulement des critères de performances et de flexibilité d'accès. Ce qui permettrait de 'transporter' plus ou moins facilement des logiciels et des données entre des équipements hétérogènes.

b. Indépendance logique

Il s'agit de permettre à chaque groupe de travail réalisant une application de pouvoir assembler différemment les données pour former des entités et des associations. De plus, chacun doit pouvoir se concentrer sur les éléments constituant son centre d'intérêt, c'est à dire que chacun doit pouvoir ne connaître et ne voir qu'une partie des données.

c. Manipulation des données par des non-informaticiens

Si les objectifs **a** et **b** sont atteints, les utilisateurs non-informaticiens verront les données indépendamment de leur implantation en machine, et peuvent alors les manipuler, les interroger et les mettre à jour, au moyen des langages non procéduraux, c'est à dire en décrivant les données qu'ils souhaitent retrouver sans décrire la manière de les retrouver qui est propre à la machine.

d. Efficacité des accès aux données

Il s'agit d'abord de fournir des langages de manipulation de données très efficaces permettant d'accéder rapidement aux données. D'autre part, pour les non-informaticiens, il doit être possible d'utiliser un langage non procédural dont l'implantation doit être très efficace, notamment au niveau des accès aux disques qui restent le goulot d'étranglement essentiel des SGBD.

e. Administration cohérente des données

L'administration des données est l'ensemble des fonctions privilégiées ayant trait à : la définition des structures de stockage, la création des espaces de travail pour les utilisateurs, l'attribution aux utilisateurs des droits d'accès, ...etc. L'extrême importance de ces fonctions, rend obligatoire la mise sous contrôle du SGBD par un ou plusieurs personnes hautement qualifiées (appelées administrateurs du système). L'administration des données est ainsi souvent centralisée. L'objectif est seulement de permettre une administration cohérente et efficace des données.

f. Non redondance des données

Dans les systèmes classiques à fichiers non intégrés, chaque application possède des données propres. Ceci conduit généralement à de nombreuses duplications de données avec, outre la perte de place en mémoire secondaire associée, un gâchis important en moyens humains pour saisir et maintenir à jour plusieurs fois les mêmes données.

Avec une approche 'base de données', les fichiers plus ou moins redondants sont intégrés en un seul fichier partagé par les diverses applications.

g. Cohérence des données

Les données peuvent être soumises à certaines règles. Par exemple, une quantité commandée Q d'un produit P ne doit pas être supérieure à la quantité S en stock du même produit. Le SGBD doit donc veiller à ce que les applications respectent ces règles lors des modifications des données et doit ainsi assurer la cohérence des données.

h. Partageabilité des données

Cet objectif consiste à permettre aux applications de partager les données de la base dans le temps mais aussi simultanément. Une application doit pouvoir accéder aux données comme si elle était seule à les utiliser, sans attendre mais aussi sans savoir qu'une autre application peut les modifier en même temps.

i. sécurité des données

Les données doivent être protégées contre les accès non autorisés ou mal intentionnés. Il doit exister des mécanismes pour autoriser, contrôler ou enlever les droits d'accès de n'importe quel usager à un ensemble de données.

Par exemple, un employé pourra connaître les salaires des personnes qu'il dirige, mais pas des autres employés de l'entreprise.

5. Concepts et méthodes de base

5.1. Notion de base

Type d'objet : ensemble d'objets possédant des caractéristiques similaires et manipulables par des opérations identiques.

Exemples :


- le type d'objet entier muni des opérations $\{+, -, \times, /\}$
- le type d'objet voiture composé des attributs suivants :
voiture(N°V, marque, type, couleur)

Peut être muni des opérations créer, détruire, modifier, consulter qui sont des opérations standards sur un objet.


Occurrence d'objet : élément d'un ensemble d'objets

Exemple :

- l'entier 10
- la voiture(139 RH 98, Renault, mégane, blanche)

 Toute description de données s'effectue donc au niveau du type à l'aide d'un ensemble d'éléments descriptifs permettant d'exprimer les propriétés d'ensembles d'objets et composant un modèle de description de données.

Modèle de description de données : ensemble de concepts et de règles de composition de ces concepts permettant de décrire des données.

 Un modèle de description de données est souvent représenté par un formalisme graphique. Il est mis en œuvre à l'aide d'un langage de description de données.

Langage de description de données : langage supportant un modèle et permettant de décrire les données d'une base de données d'une manière assimilable par une machine.


 La description d'un ensemble de données particulier, correspondant par exemple à une application, à l'aide d'un langage de description donne naissance à un schéma.

Schéma : description au moyen d'un langage déterminé d'un ensemble de données particulier.

5.2. Les niveaux de description des données

Pour assurer l'indépendance, physique et logique, on distingue trois niveaux de descriptions de données :

Le niveau conceptuel

Correspond à la structure canonique des données qui existent dans l'entreprise, c'est à dire leur structure sémantique inhérente sans souci d'implantation en machine, représentant la vue intégrée de tous les groupes de travail. Par exemple, le schéma conceptuel permettra de définir :

- les types de données élémentaires qui définissent les attributs des objets de l'entreprise (exemple : numéro de véhicule, puissance, couleur. ...)
- les types de données composés qui permettent de regrouper les attributs afin de décrire les entités du monde réel (exemple : personne, voiture...)
- les types de données composés qui permettent de regrouper les attributs afin de décrire les associations du monde réel (exemple : possède.....)
- les règles que devront suivre les données au cours de leur vie dans l'entreprise (exemple : tout personne possède une voiture ou plus)

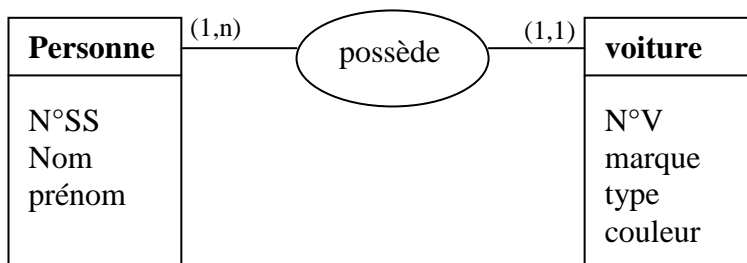


Fig.I.2-exemple de schéma conceptuel

Le niveau interne

Correspond à la structure de stockage supportant les données. Il permet donc de décrire les données telles qu'elles sont stockées dans la machine, par exemple :

- les fichiers qui contiennent ces données (nom, organisation, localisation...)
- les articles de ces fichiers (longueur, champs, modes de placement en fichier...)
- les chemins d'accès à ces articles (index, chaînages.....)

Le niveau externe

Chaque groupe de travail utilisant des données possède une description des données perçues effectuée de la manière dont il les voit dans ses programmes d'application. Alors qu'au niveau conceptuel et interne, les schémas décrivent toute une base de données, au niveau externe ils décrivent simplement la partie des données présentant un intérêt pour un utilisateur ou un groupe d'utilisateurs.

Il est souligné que pour une base de données, il existe un seul schéma interne et un seul schéma conceptuel. Il existe par contre, en générale, plusieurs schémas externes. Ainsi, certains schémas externes peuvent être déduits les uns des autres.

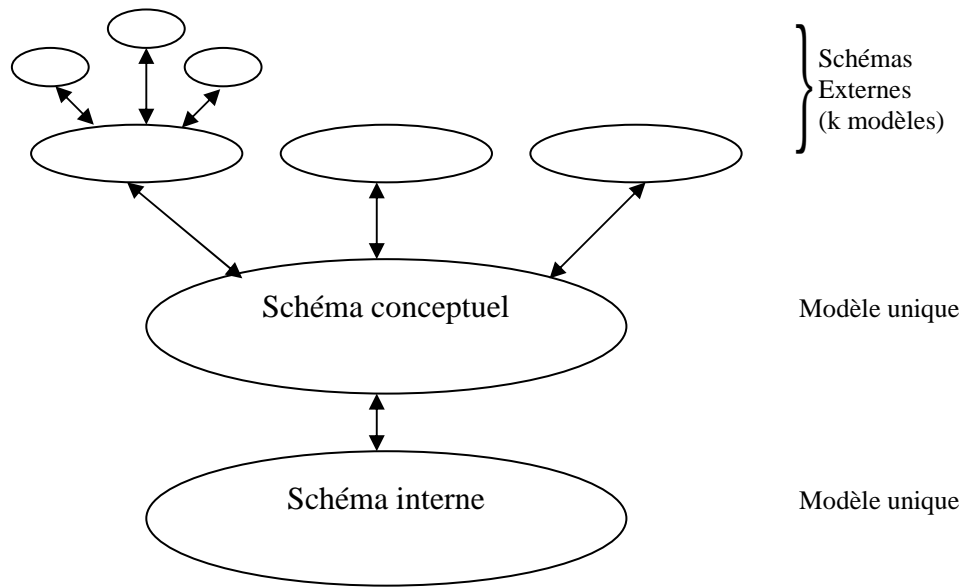


Fig.1.3-différents schémas d'une base de données