

1. Introduction aux bases de données

1.1. Base de données

1.1.1. Définition

Une base de données est une banque d'informations interdépendantes mémorisées sur un support de stockage accessibles par une ou plusieurs applications, sans répétition inutile et structurées d'une façon indépendante de toute application pour répondre aux besoins de différents utilisateurs [3].

Exemple :

Dans une université les informations concernant : étudiants, enseignants, cours présentés et moyens généraux peuvent être regroupées et mises à la disposition de nombreux utilisateurs (service de pédagogie, service de stages, conseil scientifique, etc.).

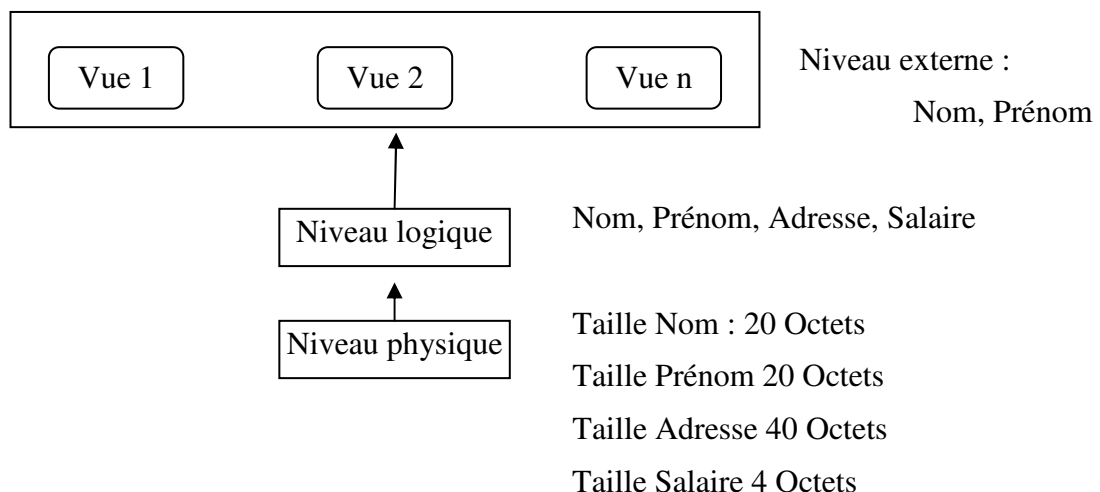
1.1.2. La description d'une base de données

1.1.2.1. Définitions

1. Un schéma est simplement la description des données contenues dans la base. Cette description est conforme à un modèle de données qui propose des outils de description (structures, contraintes et opérations).
2. Une instance est le contenu effectif de la base de données à un instant donné.

La description d'une BD s'effectue à trois niveaux [1]:

1. Le niveau Externe : ce niveau concerne la définition des types d'utilisateurs qui pourront chacun avoir une vue distincte d'une même base. Chacune de ces vues est déduite du schéma conceptuel.
2. Le niveau Interne ou physique : ce niveau est relatif aux développeurs. L'organisation physique des données et les fonctions d'accès sont alors définies (organisation des fichiers, index, structure...).
3. Le niveau conceptuel englobe toutes les vues externes sans chercher comment sauvegarder les données (description abstraite).



1.2. Système de Gestion de Base de Données

1.2.1. Définition

SGBD ou Data Base Management System est un logiciel système destiné à stocker et à partager des informations dans une base de données, en garantissant la qualité, la pérennité et la confidentialité des informations, tout en cachant la complexité des opérations (wikipedia).

1.2.2. Les buts et propriétés de ces systèmes sont multiples

- Offrir une description des informations sauvegardées dans la base (on doit assurer une indépendance entre les données et les programmes qui les traitent) [8].
- Offrir une interface pour assurer l'interaction avec la base de données (recherche, mise à jour, suppression).
- Offrir un langage de manipulation des données (LMD) pour permettre à l'utilisateur de formuler des requêtes pour manipuler les bases de données.
- Garantir l'indépendance physique et logique des données : l'utilisateur n'a pas à connaître l'organisation physique des données (accès séquentiel, séquentiel indexé...) et leur organisation logique. Un changement de l'implantation de la base ne doit pas avoir d'effet sur les requêtes de l'utilisateur. Cette indépendance permet en particulier d'offrir plusieurs vues partielles de la même base à différents types d'utilisateurs.
- Minimiser la présence de redondances inutiles.

On distingue deux types de redondance :

1. La même donnée est présente plusieurs fois dans des fichiers différents.
 2. Une donnée est présente physiquement dans un fichier mais peut se déduire d'autres données.
- Maintenir la cohérence des données par une administration centralisée prise en charge par le Data Base Administrator (DBA) (validité des données, dépendance des données...). On peut citer par exemple les cas :
 1. De la redondance obligatoire où la mise à jour devra être « *complète* » pour assurer la cohérence de la base.
 2. De la vérification des contraintes d'intégrité qui doivent être constamment vérifiées pour assurer la cohérence de la base. Elles sont vérifiées lors des mises à jour des n-uplets : valeur devant être nécessairement définie, intervalle de valeurs possibles, typage...
 - Offrir un accès multiple aux données et régler le problème de la concurrence d'accès (problème de l'exclusion mutuelle lors, par exemple, de la mise à jour « simultanée »).
 - Garantir la sécurité d'accès aux données (confidentialité, identification des utilisateurs).
 - Administrer les possibilités de pannes et les rendre transparentes à l'utilisateur (point de reprise).
 - Fournir un accès efficace aux données.

Parmi les SGBD existant, on peut citer : Oracle, Ingres, SQL Server, O2, Access, DB2, MySQL, MSQ, PostgreSQL...