

# **ARCHITECTURE D'UN SYSTEME DE GESTION DES BASES DE DONNEES**

**SANE ARNAUD & Dr N'GUESSAN GERARD**

# Table des matières



<b>Objectifs</b>	3
<b>I - LES NIVEAUX DE DESCRIPTION DE DONNEES</b>	4
<b>II - EVALUATION 1</b>	7
<b>III - LES LANGAGES DE DONNEES</b>	8
<b>IV - EVALUATION 2</b>	9
<b>Ressources annexes</b>	10

# Objectifs

Décrire les différents niveaux de l'architecture d'un système de gestion de bases de données

Définir les types de langages de manipulation d'une base de données

# LES NIVEAUX DE DESCRIPTION DE DONNEES



## *1. Le niveau conceptuel*

Il s'agit ici de décrire les niveaux de présentation de données dans un Système de Gestion des Bases de données.

Il correspond à la description des données et des liens existant entre elles en tenant compte des contraintes de gestion (Dictionnaire des données), sans souci d'implémentation.

c'est le niveau conçu par l'analyse, il fait intervenir le modèle entité-association.

Ce niveau permettra de définir :

- Les types de données élémentaires qui définissent les propriétés élémentaires d'un objet( exemple : nom, prenom etc.).
- Les types de données composées qui font intervenir un ensemble de propriétés portant sur des objets de même nature appelés entités (Exemple PRODUIT, commande etc.)
- Les types de données composées qui permettent de décrire les associations.

## *2. Le niveau interne*

Il correspond à la description de la structure des données dans les moindres détails et de son implémentation sur un support de stockage.

Il permet de décrire les données telles qu'elles seront stockées à la machine dans la base de données.

Ce niveau fait intervenir le modèle relationnel aboutissant au modèle physique des données qui précède l'implémentation d'une base de données.

### *a. Illustration d'une table élaborée dans le modèle physique des données*

Exemple : Structure de la table ETUDIANT.

Cette table a pour attributs(propriétés) :

Le matricule étudiant (matetud)

le nom étudiant (nometud)

les prénoms étudiant (pnometud)

le contact étudiant (contetud)

<i>NOM TABLE</i> : ETUDIANT <i>CLE PRIMAIRE</i> : matetud <i>TYPE D'ORGANISATION</i> : direct <i>MODE D'ACCES</i> : direct <i>TAILLE</i> : 71			
<i>CODE</i>	<i>DESIGNATION</i>	<i>TYPE</i>	<i>LONGUEUR</i>
matetud	Matricule étudiant	AN	10
nometud	Nom étudiant	AN	20
pnometud	Prénoms étudiant	AN	30
contetud	Contact étudiant	AN	11

*b. Description des éléments constituant le modèle physique des données.*

*NOM TABLE* : Définit le nom de la table décrite.

*CLE PRIMAIRE* : Définit l'attribut permettant d'identifier une occurrence de la table décrite.

*TYPE D'ORGANISATION* : Définit l'organisation des données dans la table, c'est à dire la manière dont ces données seront enregistrées. Ainsi, nous distinguons trois types d'organisation :

- *Organisation séquentielle* : c'est une organisation qui n'a aucun critère de rangement des données (pas de clé primaire), les enregistrements sont effectués les uns à la suite des autres sans identifiant. L'exemple ici peut être l'enregistrement d'un étudiant sans matricule.
- *Organisation relative ou directe* : Contrairement à l'organisation séquentielle, elle a un critère de rangement (une clé primaire). chaque enregistrement est référencé. L'exemple dans notre cas est celui du matricule étudiant.
- *Organisation séquentielle indexée* : Elle est d'abord relative et la définition de sa clé primaire tient compte de celle d'un champ de la table appelé table d'index. Par exemple, si l'étudiant est de sexe masculin (M), la définition de son matricule peut être fonction de ce champ, et ceci donnerait par exemple (M0001).

*MODE D'ACCES* : Il détermine la manière dont les données pourront être accessibles ; ils sont fonction du type d'organisation. Ceci dit :

- Lorsque l'organisation est directe ou séquentiel indexé, nous pouvons directement accéder à une information à partir de sa clé primaire, on parle de *mode d'accès direct*.
- Lorsque l'organisation est séquentielle, le *mode d'accès est séquentiel* car pour accéder à une information, il faut passer par tous les enregistrements qui la précède.

*TAILLE* : Elle correspond à la longueur maximale d'un enregistrement, c'est à dire à la somme des longueurs maximales des différents champs. la taille est exprimée en octet car un caractère est égal à un octet.

### 3. Le niveau externe

Ce niveau présente la manière dont les données se présentent dans une application (bien entendu, après son implémentation et son déploiement) pour chaque utilisateur ou pour chaque groupe d'utilisateurs. Il intervient dans l'exploitation des données par les utilisateurs. ces différents écrans présentant des données différentes

issues de la même base de données sont appelées *vues externes*. Les vues présentent généralement une partie des données d'une même base en fonction du type d'accès. De même, la même information pourra présenter des vues différentes.

*Aperçu grand format (cf. p.10) (cf. p.10)*

# EVALUATION 1

  
II

## Exercice

---

Parmi ces niveaux de description de données, quel est celui qui ne s'intéresse pas à l'implémentation des données ?

- ☐ Le niveau interne
- ☐ Le niveau conceptuel
- ☐ Le niveau externe

## Exercice

---

Contrairement à l'organisation séquentielle, l'organisation séquentielle indexée

- ☐ tient compte d'un index (clé primaire).
- ☐ n'a pas besoin de clé primaire mais a plutôt besoin d'une table d'index.
- ☐ ne définit que le mode d'accès
- ☐ séquentiel

## Exercice

---

Dans l'architecture de données, la perception des informations d'une base de données est prise en compte au niveau

- ☐ conceptuel
- ☐ externe
- ☐ interne

# LES LANGAGES DE DONNEES

## III

### *1- LES LANGAGES RELATIONNELS*

Ce sont des langages qui permettent d'effectuer des traitements.

Au sens du relationnel, un langage d'interrogation de base de données est complet, lorsqu'il permet au moins l'utilisation d'un certain nombre d'opérateurs à savoir :

- Les opérateurs ensemblistes (l'union, la différence, le produit cartésien)
- Les opérateurs relationnels (la projection et la sélection)

De plus, ces langages doivent , en pratique s'étendre à d'autres fonctionnalités telles que :

- La possibilité de créer des tables, d'effectuer des insertions, des modifications et des suppression.
- La possibilité d'utiliser des fonctions d'agrégation telles que la somme, la moyenne, le maximum etc.
- La possibilité de gérer les accès pour assurer une protection des bases de données.

Comme exemples de langages relationnels, nous pouvons citer entre autres :

- L'algèbre relationnelle
- Le calcul relationnel par n-uplet
- Le calcul relationnel par domaine

### *2- LES LANGAGES PRATIQUES*

Ce sont des langages de requêtes qui jouent deux rôles essentiels à savoir :

- Le rôle de langage de définition des données : Il contient des commandes de création, de modification et de suppression de relations (tables) ainsi que les commandes de gestion des accès. Il est généralement destiné à l'administrateur de bases de données.
- Le rôle de langage de manipulation de données : il s'agit des commandes de recherches et de mises à jour des données telles que l'insertion, la modification, la sélection et la suppression des données.

Comme exemples de Langages pratiques, nous pouvons citer entre eux :

- SQL (Structured Query Language)
- QUEL (Query Language)
- SEQUEL (Structured English as a Query Language)
- QBE (Query By Example)



# EVALUATION 2

  
IV

## Exercice

---

Dans le langage relationnel, quel est l'opérateur qui ne fait pas partie des opérateurs ensemblistes ?

- ☐ La différence
- ☐ l'union
- ☐ la sélection

## Exercice

---

Identifiez les opérateurs relationnels parmi ces opérateurs

- ☐ La projection
- ☐ La différence
- ☐ La sélection
- ☐ L'union

## Exercice

---

Quel type de langage utilisé pour modifier la taille d'un champ dans une table ?

- ☐ Le langage de manipulation des données
- ☐ Le langage de contrôle des données
- ☐ Le langage de définition des données

