

Les modèles de données

Nous avons défini un modèle de données comme un ensemble de concepts et de règles de composition de ces concepts permettant de décrire des données. Il existe en fait trois modèles de données principales : le modèle hiérarchique, le modèle réseau et le modèle relationnel. Historiquement les deux premiers modèles ont été conçus dans la période 1965-1970, et ont donné naissance à plusieurs systèmes commercialisés : IMS (Information Management System) utilisant le modèle hiérarchique, IDS(Integrated Data Store) utilisant le modèle réseau...etc.

Nous verrons dans le chapitre suivant que le modèle relationnel a remplacé progressivement ces deux modèles cités.

1. Le modèle réseau

Le modèle réseau a été proposé par le groupe DBTG du comité CODASYL. Il a été amélioré à plusieurs reprises grâce à des études qui ont abouti surtout à sa normalisation.

Les éléments de base de ce modèle sont les notions d'ensembles d'entités (enregistrement logique) et d'associations entre ensembles d'entités (liens).

Enregistrement logique et lien

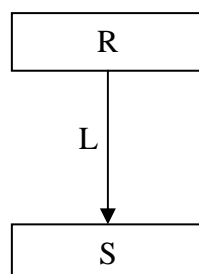
La notion d'enregistrement logique n'est pas spécifique aux SGBD, c'est celle que l'on rencontre dans la plupart des systèmes de gestion de fichiers et des langages de programmation. Elle permet de regrouper des données élémentaires en enregistrements logiques.

Exemple : etudiant = enregistrement

 Numéro_etudiant : entier, clé
 Nom : caractere(10)
 Adresse : caractere(40)
 Fin

Le deuxième élément fondamental dans le modèle réseau est le lien. Un lien peut se définir comme la représentation d'une association, l'association est une perception abstraite de la réalité alors que le lien va être sa matérialisation.

Un lien L est défini entre deux types d'enregistrements R et S dans une direction donnée. Il offre un mécanisme d'accès qui, à partir d'un enregistrement r de type R, permet d'accéder aux enregistrements s_1, s_2, \dots, s_n de type S qui lui sont reliés. Conventionnellement, un lien est représenté par un diagramme appelé *diagramme de bachman*, du nom de l'inventeur, où sont mentionnés les types d'enregistrement R et S et le lien L qui va de R vers S.



Par exemple, soit le diagramme suivant (fig.II.1) qui associe les entités 'étudiant' et 'formation' par le lien 'est inscrit à'

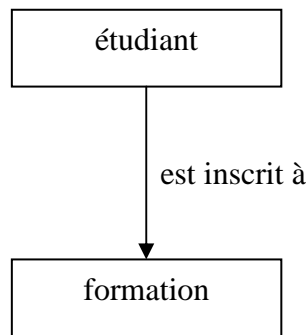


Fig.II.1- exemple d'entités et de lien

Si à chaque occurrence de l'entité R, il existe une et une seule occurrence de l'entité S, le lien est dit de type : 1-1, par exemple : un étudiant ne peut s'inscrire qu'à une seule formation.

Si on peut associer à chaque occurrence de l'entité R plusieurs occurrences de l'entité S, le lien est dit de type : 1-n, par exemple : un père peut avoir plusieurs fils.

Si à plusieurs occurrences de l'entité R, on peut associer une seule occurrence de l'entité S, le lien est dit de type : n-1, par exemple : plusieurs enfants sont les descendants d'un seul père.

Si à une occurrence de l'entité R, on peut associer n occurrences de l'entité S, et inversement, à une occurrence de l'entité S, on peut associer m occurrences de l'entité R, le lien est dit de type : m-n. par exemple, une formation peut faire appel à n matières, et inversement, une matière peut faire partie de m formations différentes. Nous verrons dans le paragraphe suivant que ce type de lien pose un problème d'implémentation. En effet, plusieurs techniques ont été proposées pour le représenter, ce qui a conduit à des variantes du modèle réseau : modèle réseau CODASYL, modèle réseau SOCRATE....etc.

Le modèle réseau CODASYL

Le lien appelé 'SET' dans la terminologie CODASYL, ne peut être utilisé que lorsque l'association qui existe entre les enregistrements de type R et S fait correspondre à un enregistrement r les enregistrements s_1, s_2, \dots, s_n alors qu'inversement à chaque élément s_i ne correspond au plus qu'un élément r. Dans ce cas, on pourra construire un lien directionnel de R vers S.

Cela veut dire, qu'un lien CODASYL pourra être construit si l'association est de type 1-1, 1-n ou n-1. Dans le cas d'une association de type m-n, il faudrait introduire un enregistrement de nouveau type (voir ci-dessous) ; avant cela, expliquons comment se fait le mécanisme d'accès.

Le mécanisme d'accès qui permet de passer de l'enregistrement r aux enregistrements s_1, s_2, \dots, s_n est celui d'une liste circulaire (voir fig.II.2) où la tête de liste est constituée par l'enregistrement r et les éléments de la liste sont constitués par les enregistrements s_1, s_2, \dots, s_n

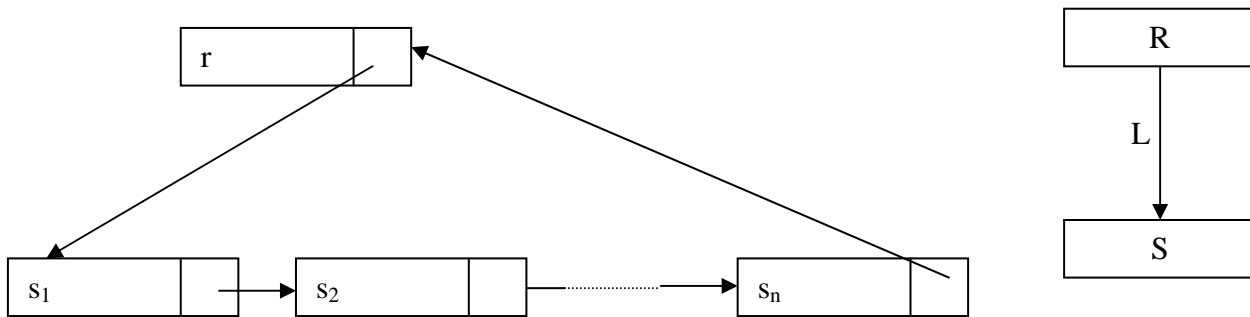


Fig.II.2-représentation d'un lien CODASYL à l'aide d'une liste circulaire

Cette représentation est possible car chaque enregistrement de type S n'étant associé qu'à un seul enregistrement de type R , et ne peut apparaître que dans une seule liste circulaire. En effet, si un enregistrement de type S était parcouru par plusieurs listes, il y aurait un nombre variable de pointeurs et c'est la raison pour laquelle une association de type $m-n$ ne peut être représentée par un lien CODASYL sans une transformation préalable.

Dans le cas d'une association de type $m-n$ entre deux types d'enregistrements R et S , la solution consiste à se ramener à deux associations de type $1-n$ en construisant un enregistrement de type noté $R \times S$ (fig.II.3) dont les éléments sont tous les couples de la forme (r_i, s_j) si r_i est de type R , s_j de type S et r_i et s_j étant reliés entre eux dans l'association. Si on examine maintenant les associations qui existent respectivement entre $R \times S$ et R et S , on remarque qu'elles sont toutes deux de type $1-n$

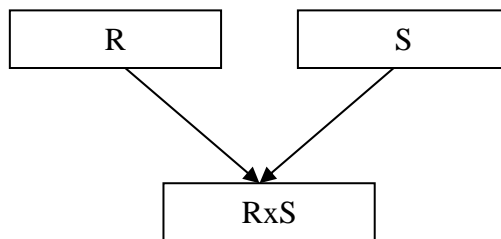


Fig.II.3-représentation de l'association $m-n$ en formalisme CODASYL

Schémas réseaux

La notion de lien CODASYL telle qu'elle vient d'être définie va permettre de construire des schémas réseaux (fig.II.4) dont la structure sera spécifique de chaque application, mais qui d'une façon générale auront les propriétés suivantes :

- d'un enregistrement de type R peuvent partir autant de liens différents que l'on veut
- à un enregistrement de type S peuvent arriver autant de liens que l'on veut
- entre deux enregistrements de type R et S, il peut y avoir plusieurs liens différents allant de R vers S, et inversement
- pour un enregistrement de type R, un lien partant de R ne peut boucler sur R. Cette limitation a en fait été levée dans certains systèmes commercialisés.

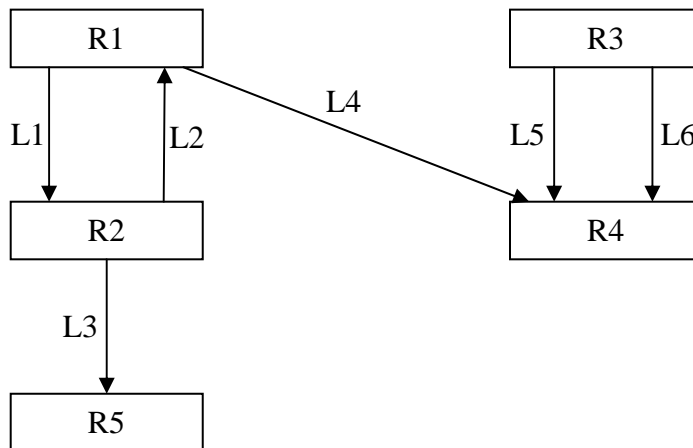


Fig.II.4-exemple de diagramme réseau CODASYL

A titre d'exemple, nous allons présenter le diagramme CODASYL d'une base de données 'toxicomane' (fig.II.5) composée des enregistrements suivants :

- TABAC composé des données élémentaires suivantes : Numéro-Tabac et Nom-Tabac
- FUMEUR composés des données : Numéro-Fumeur, Nom et Prénom
- ABUS décrivant pour chaque tabac, la quantité fumée par un fumeur
- PRODUCTEUR définissant pour chaque tabac, le nom du producteur
- COMMANDES spécifiant les commandes de tabac passées par les fumeurs aux producteurs

Les liens existants entre ces différents enregistrements sont :

- PRODUCTION qui associe un producteur aux tabacs produits
- VENTE qui associe un tabac aux commandes correspondantes
- ACHAT qui associe un fumeur à ses commandes
- FUMER qui associe un fumeur à une quantité de tabac fumé
- CONSOMMATION qui associe un tabac à toutes ses quantités fumées par les différents fumeurs.

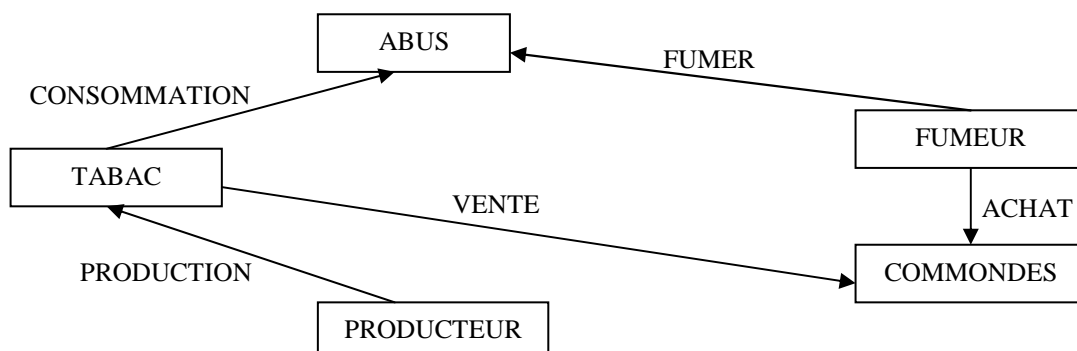


Fig.II.5-Diagramme de la BDD réseau 'toxicomane'

Les opérations de manipulation dans un modèle réseau

Il y a deux grandes catégories d'opérations : celles qui permettent de modifier le contenu de la base de données et celles qui permettent de sélectionner de l'information

Les opérations qui permettent de modifier le contenu de la BDD sont les opérations de création, de suppression et de mise à jour

Pour le modèle CODASYL, considérons un enregistrement r de type R et un enregistrement s de type S , et supposons que nous voulions ajouter le couple (r,s) comme appartenant au lien L qui va de R à S .

Si les enregistrements r et s n'existent pas dans la BDD, la première étape consiste à insérer ces deux enregistrements dans les fichiers relatifs aux enregistrements de type R et S , en respectant l'organisation des données relatives à ces deux fichiers.

La deuxième étape consistera à construire une liste circulaire ayant pour point de départ l'enregistrement r .

Pour ce qui est de la recherche d'information, le programmeur doit utiliser la connaissance qu'il a des chemins d'accès de la base de données.

2. Le modèle Hiérarchique

Les éléments de base du modèle hiérarchique sont les enregistrements logiques qui sont reliés entre eux pour constituer une arborescence. Une arborescence est un cas particulier d'un graphe réseau dans lequel un nœud, peut recevoir au plus une flèche, et d'où il peut partir un nombre quelconque de flèche, comme le montre la figure II.6.

Sur l'arborescence, il existe un nœud particulier : la racine, c'est un nœud qui ne reçoit aucune flèche. On appelle fils d'un nœud l'ensemble des nœuds qui se trouvent à une distance 1 de ce nœud. On appelle père d'un nœud, la notion inverse de celle de fils. Dans une arborescence, tout nœud n'a qu'un seul père et peut avoir plusieurs fils.

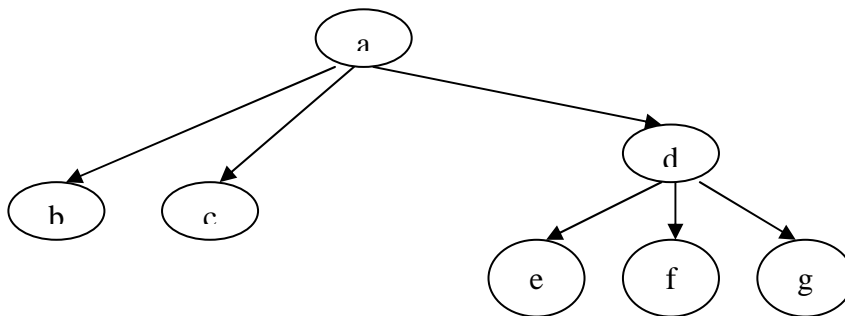


Fig.II.6-une arborescence

3. COCLUSION

Dans ce chapitre, nous avons introduit les deux principaux modèles de données, réseau et hiérarchique, qui ont existé bien avant le modèle relationnel. Compte tenu de l'accroissement des possibilités des ordinateurs, constaté à partir des années 70, ces modèles sont actuellement critiqués. Ils peuvent rester très compétitifs au niveau interne. Les critiques proviennent surtout de la complexité des langages de manipulations associés à ces modèles, basé sur la '*navigation*', donc peu accessibles aux utilisateurs néophytes.

Finalement, la question se pose de savoir si le modèle relationnel est un bon candidat pour constituer un modèle conceptuel. D'après Codd, l'inventeur du modèle relationnel, un bon modèle doit avoir quatre caractéristiques :

- 1- Permettre une représentation sous forme de tableaux des données.
- 2- Permettre une interprétation ensembliste afin d'autoriser des opérations similaires à ceux de l'algèbre relationnelle.
- 3- Permettre une interprétation des objets en termes de formules de la logique afin de faciliter les déductions de connaissances.
- 4- Supporter une représentation graphique afin de pouvoir visualiser les objets et leurs associations pour concevoir la base.

Ajoutons que la simplicité est une caractéristique essentielle d'un modèle. C'est là un des avantages déterminant du modèle relationnel qui, dans sa version entité-association, possède les quatre caractéristiques précédentes.