

Annexe 2

Les concepts du modèle relationnel

Les bases de données relationnelles¹

L'usage des bases de données, au sens où nous l'entendons aujourd'hui, a débuté dès les années 1960. Aujourd'hui elles sont présentes dans de divers domaines d'activités. Plusieurs modèles conceptuels d'agencement et de stockage des données ont existé au cours des années, impliquant également la publication de logiciels propriétaires en très grand nombre. En 1970, une évolution majeure allait cependant modifier durablement les habitudes en matière de structuration et de gestion des informations. Le modèle relationnel conçu par E.F Codd énonce clairement ces objectifs : proposer un modèle simple, rigoureux et complet. Il devint possible de tout manipuler, de tout décrire et de tout figurer avec ce nouveau modèle. En 1975, le concept « entité-association » est défini pour clarifier l'organisation des données dans les bases relationnelles. Il s'agit d'une représentation graphique sous forme de diagramme destinée à modéliser les données (entités) et à décrire toutes les relations (associations). Une entité correspond à un ensemble d'informations liées à un même objet. Les objets peuvent être attachés entre eux via les associations. Actuellement les bases de données relationnelles sont les plus utilisées même si d'autres modèles continuent d'être exploités².

Les éléments techniques fondamentaux d'une base de données relationnelle :

- Une base de données est constituée d'un ensemble de tables (ou entités).
- Chaque table contient des objets de même nature . Par exemple, dans une base de données de gestion de ventes, plusieurs tables pourraient symboliser les relations commerciales avec comme objets définis : les clients, les commandes et les produits.
- Une ligne d'une table (ou enregistrement) regroupe toutes les informations d'un même objet. La ligne comprend toutes les valeurs des colonnes d'une table.
- Une colonne de table représente une propriété commune des objets de la table. En suivant le même exemple que précédemment, une table « client » contiendrait des objets relatifs aux clients, leurs noms, leurs lieux d'habitation et leurs coordonnées.
- Chaque ligne est distincte. Les colonnes dont la valeur est unique dans chaque ligne sont appelées identifiants (techniquement il s'agit de la **clé primaire** ou de la **clé candidate**).
- Les lignes d'une table peuvent faire référence à d'autres lignes d'autres tables. C'est ce qu'on appelle une association (ou relation) qui se traduit par la définition d'une **clé étrangère**. Toujours dans le même exemple, une table « client » contenant un identifiant unique pourrait servir de référence à une autre table « commande » répertoriant toutes les commandes faites et indiquant le numéro unique du client qui l'a passée. L'identifiant de la table « client » est utilisé dans cet exemple comme référence à une colonne « numéroClient » sur laquelle est définie une clé étrangère.

Le déploiement d'une base de données relationnelle

Les bases de données sont employées pour présenter le plus finement possible les informations relatives à une activité humaine. Elles répondent à des contraintes de gestion précises. Des logiciels qui peuvent être assez complexes ont été développés pour contenir à la fois les structures (tables) et les données (lignes) sur un serveur et pour imposer les

¹ Cette brève introduction aux bases de données relationnelles provient notamment de l'ouvrage de Jean-Luc Hainaut, *Base de données, Concepts, utilisation et développement*, collection Sciences Sup, Dunod, Paris, 2009.

² A titre d'exemple, le modèle hiérarchique ou plus récemment, le modèle noSql : Not Only Sql.

contraintes inhérentes au déploiement du modèle entité-association. Ils sont appelés « systèmes de gestion des bases de données relationnelles ». Dans l'étude relative au format SIARD, les SGBD supportés par le programme d'archivage sont les produits Oracle et Microsoft. Les fonctionnalités offertes sont de plusieurs ordres et intègrent des notions techniques que le format SIARD est capable d'archiver.

Les contraintes basiques

L'organisation des données nécessite de créer au préalable un schéma conceptuel qui permet ensuite d'insérer les tables et leurs colonnes. Ces dernières doivent être typées et il est nécessaire de renseigner leur caractère obligatoire ou facultatif. L'administrateur ajoute les identifiants **primaires** (clés primaires) et **secondaires** (clés candidates) pour définir les contraintes d'intégrité et d'unicité. Les contraintes **référentielles** correspondent aux clés étrangères et permettent de figurer les associations entre les différentes tables. On appelle ces contraintes de base, les **déclarations natives**.

Les contraintes additionnelles

La gestion des données s'effectue par l'ajout, la modification ou le retrait des lignes, des colonnes ou des tables. Il est primordial de pouvoir effectuer des vérifications lors de ces opérations. Pour ce faire, les **prédicats** de colonne ou de tables qui permettent d'exprimer une propriété simple de la valeur d'une ligne, sont utilisés. Par exemple, une colonne contenant une catégorie « B1 » ou « B2 » ne pourrait pas contenir une valeur « B3 » grâce à la vérification d'un prédicat.

De même, les **déclencheurs** qui sont de petits morceaux de code sont employés pour définir des contraintes dynamiques. Ils se déclenchent avant ou après une modification de ligne.

Enfin les **procédures stockées** sont utilisées pour remplacer l'opérateur de modification natif défini dans le SGBD et pour définir des comportements complexes, notamment en matière d'intégrité des données. Elles sont codées dans des langages propres et propriétaires créés par les éditeurs de SGBD.

Les trois fonctionnalités décrites dans ce paragraphe sont présentes pour définir les **contraintes supplémentaires** (ou additionnelles) aux déclarations natives. Leur emploi conjugué permet en théorie d'appliquer toutes les contraintes que l'on souhaite sur une base de données relationnelle.

La manipulation des données : les vues et les droits d'accès

La personnalisation des données extraites est opérée grâce à l'usage des vues. Techniquement, il s'agit de tables virtuelles en cela qu'elles ne possèdent qu'une structure conceptuelle sans contenir d'enregistrements. Les **vues** peuvent être assimilées à des interfaces de consultation des lignes de la base.

Le modèle relationnel permet également de créer des privilèges et des droits d'accès. Il est en possible que plusieurs utilisateurs reconnus accèdent en même temps à une même base de données. Dans la plupart des cas, des **privilèges** leur sont alors accordés. Il s'agit d'autorisations affectées au déclenchement d'opérations. Parfois des **rôles** sont également définis. Ils sont des classes générales d'utilisateurs auxquelles l'administrateur peut accrédi-ter de nouveaux privilèges.

Un langage d'usage natif :

L'accès aux données se fait par l'usage du langage de requête normalisé SQL spécialement développé pour ce modèle. Les différents éditeurs de SGBD ont conçu leur propre langage de requête basé sur la norme SQL. Il s'agit d'adaptations propriétaires qui sont parfois très différentes dans la définition des contraintes. Il peut arriver également que d'autres langages de programmation soient implémentés et utilisés par les SGBD. SIARD peut en théorie les archiver tels quels même si cette démarche ne présente pas beaucoup d'intérêt, dans la mesure où ils ne sont pas décrits.

Les autres fonctionnalités offertes par les SGBD :

Les SGBD proposent beaucoup d'autres fonctionnalités qui ne trouvent cependant pas de fondements dans l'usage du modèle relationnel. On peut citer l'exemple de l'accès au serveur par l'intermédiaire de programmes distants et d'API propres. Le programme *SiardSuite* se connecte par exemple aux SGBD grâce à des pilotes ODBC-JDBC . Il est évident que le format SIARD ne prévoit pas d'archiver ces autres fonctionnalités qui n'ont pas un rapport direct avec la gestion et avec la structuration des informations. Il faut retenir que le format d'archivage SIARD a été conçu pour archiver toutes les contraintes relatives à l'usage au modèle relationnel.