

## Chapitre 6 - SGBDR et L4G

Alors que le marché des bases de données semblait clairement réparti dans les années 80 avec les SGDB relationnels pour le décisionnel et les SGBD traditionnels pour le transactionnel, cette frontière s'effaça dans les années quatre-vingt-dix et les SGBDR se généralisèrent. Mais, avant cela, le chemin fut long et difficile. Nombreux doutaient que les SGBDR deviennent plus qu'une curiosité de laboratoire pour universitaires.

### Charles Bachman lance le mouvement

La préhistoire des bases de données repose presque entièrement sur un nom : Charles Bachman. C'est lui qui, en 1961 et chez General Electric, développa le premier IDS (pour Integrated Data Store), un système pionnier de base de données capable de prendre en compte les nouveaux supports de stockage comme les disques magnétiques qui commençaient à apparaître. Les tout premiers logiciels de bases de données datent donc des années soixante. La technologie d'alors était encore assez frustrée et les principes retenus ne permettaient pas l'indépendance des applications avec le système de base de données utilisé. C'est aussi Bachman qui au milieu des années soixante rassembla le DBTG (Database Task Group) qui allait publier une série d'articles et de spécifications sur les méthodes d'accès aux bases de données par des langages comme Cobol. Le DBTG fut intégré à CODASYL (déjà responsable de la normalisation du Cobol) et produisit une norme qui fut adoptée aussitôt par de nombreux constructeurs... Sauf IBM !

C'est qu'IBM, de son côté, avait lancé sur son marché son propre SGBD (IMS, en 1968) qui était dérivé d'un projet réalisé pour la NASA pendant le programme Apollo. Jusqu'à l'apparition du modèle relationnel, les principaux systèmes étaient soit hiérarchiques (comme IMS d'IBM ou Total de Cincom) soit navigationnels (ou réseau, conformes au modèle proposé par le DBTG issu de CODASYL en octobre 1969) comme IDMS (Computer Associates), Datacom DB (d'ADR, plus tard racheté par Computer Associates), ou Adabas (Software AG, Adabas -mais aussi Datacom DB- a été largement modifié avec le temps et a même été adapté au modèle relationnel...).

C'est ce modèle navigationnel promu par CODASYL qui provoqua une poussée de développement dans le monde des bases de données jusqu'alors pas très dynamique. Quand General Electric décida d'abandonner le secteur informatique, c'est la SSII de John Cullinane qui racheta leur base de données et s'occupa de la migration du logiciel pour la porter sur mainframe IBM (mais il eu aussi des versions pour DEC et ICL). IDMS (c'est le nom que donna Cullinane à ce produit) était conforme à la "norme" CODASYL alors que les autres bases de données qui dominaient jusqu'alors le marché (IMS et TOTAL) ne l'étaient pas... Du coup, IDMS connut une croissance foudroyante et ses ventes passèrent de \$2 millions en 1975 à \$20 millions en 1980.

Le développement d'applications sur ces SGBD n'était pas facile et c'est pour cela que certains éditeurs proposaient d'associer leur base de données avec un langage de programmation de haut niveau qu'on a vite baptisé L4G (langage de 4ème génération). Les deux "couples" (SGBD + L4G du même éditeur) les plus fameux du début des années 70

Cow-boys contre chemin de fer ou que savez-vous vraiment de l'histoire de l'informatique ?

étaient sans doute Datacom/Ideal (d'ADR) et Adabas/Natural de Software AG. Adabas (qui est une abréviation de "adaptable database system") vient de la société allemande Software AG. Adabas a été développé par Software AG avec l'aide de l'AVI Institut. Cet éditeur était le seul acteur européen un peu important (avant que SAP le dépasse et le fasse un peu oublier) dans un secteur entièrement dominé par les sociétés américaines.

On reviendra sur cet épisode des L4G car il le mérite... Charles Bachman reçu le prix Turing en 1973 et, à cette occasion, prononça le fameux discours "The programmer as navigator" (le programmeur comme navigateur). Charles Bachman est alors à son apogée mais intéressons-nous plutôt à celui qui va lui succéder... Un certain Codd.

### **Codd pose les bases du relationnel**

C'est Edgar F. Codd qui en 1970 posa les bases du modèle relationnel. Dans une série d'articles (Edgar F. Codd rédige en premier l'article fondateur "A Relational Model of Data for Large Shared Data Banks" dans la revue Communications of the ACM -Association for Computing Machinery), Codd proposa un modèle qui reposait sur deux principes : 1) l'indépendance des données par rapport à leur mode de stockage matériel, 2) une navigation automatique grâce à un langage de requêtes de haut niveau qui évite au programmeur de devoir "naviguer" d'un enregistrement à l'autre pour trouver la donnée qu'il cherche...

Associé à un langage d'interrogation "ad hoc", les bases de données relationnelles représentaient effectivement une vraie percée par rapport à la complexité et à la rigidité des systèmes de bases de données en usage au début des années 70. Avec une base de données classique (conforme ou non au modèle CODASYL), vous ne pouviez pas obtenir le résultat d'une requête sur votre écran simplement en la tapant depuis le clavier de votre terminal connecté au système central de votre organisation... Non, il fallait passer par un programmeur qui allait coder votre requête dans un programme qui, une fois mis au point, sera exécuté en batch et le résultat vous arrivera sur un listing... Avec un processus aussi lourd, on imagine facilement la frustration de l'utilisateur en s'apercevant que la requête était incomplète ou pire, erronée !

Il fallait recommencer tout le parcours en espérant avoir un accès rapide au programmeur capable de s'y retrouver dans la structure de la base de données... Le relationnel permettait de résoudre tout cela en une seule fois : on avait enfin une structure de base de données à peu près lisible, un langage d'interrogation assez facile à manipuler sinon à apprendre (en tout cas, bien moins complexe et rébarbatif qu'un langage de programmation, même le plus basique...) et, cerise sur le gâteau, un logiciel permettant de saisir la requête au clavier et d'avoir le résultat immédiatement (enfin, plus ou moins rapidement selon la complexité de la requête, la taille de la base de données et la qualité de son indexation...) à l'écran, un vrai paradis par rapport à la boucle programme-batch-listing ! Codd proposait donc cela à travers sa série d'article publiée en 1970 mais rien d'autre... Il fallu attendre un peu avant de voir les premières implémentations expérimentales.

### **System/R chez IBM, l'enfant non désiré !**

Codd travaillait chez IBM mais Big Blue n'avait pas l'intention de remettre en cause son SGBD traditionnel IMS qui fonctionnait bien pour une idée dont on ne savait pas encore si elle était vraiment réalisable (et si oui, amenait-elle effectivement les bénéfices espérés ?). Mais Codd sut mener une action de lobbying intelligente, y compris en acceptant un débat avec Charles Bachman qui représentait alors l'autorité suprême en matière de bases de données... Tant et si bien qu'IBM fut quasiment "forcé" de lancer un projet de recherche au laboratoire de San José pour vérifier la faisabilité des travaux de Codd.

Le projet s'appelait System/R et la première phase (en 1974/75) permit de produire rapidement un prototype qui démontrait la validité des principes énoncés par Codd. Le code de ce logiciel expérimental ne fut pas repris par la suite car il s'agissait vraiment d'un premier jet. La seconde phase (en 1978/79) produisit une version plus complète fonctionnel-

Cow-boys contre chemin de fer ou que savez-vous vraiment de l'histoire de l'informatique ?

lement, permettant l'accès concurrentiel (multi-utilisateurs) et surtout, dotée d'un langage d'interrogation qui fit du chemin depuis... SQL.

System/R était un projet de recherche et il ne fut donc jamais commercialisé en tant que tel. On sait que le premier SGDR d'IBM fut SQL/DS proposé en 1982 pour les mainframes tournant sous VM et VSE. Pour la version MVS, il fallu attendre 1983 et le fameux DB2. Ce qu'on sait moins c'est que le code de System/R fut tout de même réutilisé en partie pour la toute première implémentation d'un SGBDR chez IBM : celui qui était intégré au système d'exploitation du System/38 en 1980 (le 38 était le mini-ordinateur de Big Blue, successeur du 36 et avant l'AS/400). Cette implémentation était encore une retombée du projet "Future Systems" d'IBM, qui prévoyait de gérer des bases de données relationnelles en mode natif... L'équipe FS avait fait l'inventaire de tous les développements "pirates" à l'intérieur d'IBM (VM, APL, System/R, etc.) dans le but de les intégrer.

## La montée des SGBDR

À la suite de System/R, Larry Ellison créa Oracle en 1977 (mais, dans un premier temps, la société s'appelait Software Development Laboratories, puis en 1979 le nom changea pour Relational Software, Inc. -RSI- et, enfin, en 1982, RSI changea encore pour Oracle Systems). Mais avant Oracle, Michael Stonebraker, chercheur à Berkeley, développa un prototype universitaire de SGBDR à peu près à la même époque qu'IBM avec System/R et lui aussi en s'inspirant de l'article de Codd.

## Ingres et la vertu de l'essaimage

Codd posa les bases mais c'est bien Stonebraker qui sema la première graine ! Alors que Codd se débattait pour qu'IBM fasse enfin quelque chose avec ses idées, Michael Stonebraker agissait sans perdre de temps. Aidé d'Eugen Wong, il trouva un peu d'argent afin de lancer un projet de recherche basé sur le modèle relationnel et ce dès 1973. Initialement, les fonds sont venus de la branche études économiques de Berkeley afin de réaliser un système de modélisation géographique. C'est pour cela que le projet s'appelait Ingres (qui voulait dire Interactive Graphics and Retrieval System et non une référence au célèbre peintre français du XIXème siècle...).

Au bout du compte, c'est surtout la NSF (National Science Foundation) qui finança le projet Ingres qui dura jusqu'en 1982 (puis fut suivi de Postgres destiné à promouvoir les bases de données objets mais c'est une autre histoire et qui ne connut pas un dénouement aussi heureux que pour le relationnel !). Ingres était très semblable à System/R mais reposait sur des Minis de DEC plutôt que sur un mainframe IBM. Une première version était "montrable" dès 1974 mais fut suivie par plusieurs révisions afin de rendre le code plus maintenable. Très vite, Stonebraker commença à diffuser le code d'Ingres en dehors de Berkeley afin de recueillir des commentaires et des idées sur le prototype qui évoluait en permanence.

En 1976, Ingres fut doté d'un langage de requêtes (QUEL) très semblable à SQL (semblable mais différent tout de même...). On sait que dans l'affrontement entre QUEL et SQL, c'est ce dernier qui triompha mais le riche héritage d'Ingres va bien au-delà de cette anecdote : durant les années "Berkeley" du projet Ingres, pas moins de trente personnes ont travaillé dessus à tour de rôle (mais il n'y eut jamais plus de 6 programmeurs à la fois sur le code en évolution permanente). Si on ajoute les quinze programmeurs qui ont développé et corrigé le code de System/R, ça nous fait une petite cinquantaine d'individus qui se sont ensuite éparpillés autour de Berkeley et de San José pour créer effectivement les éditeurs qui ont donné force et vie au modèle relationnel. En effet, tous ceux qu'ont retrouvés impliqués dans les multiples projets des années 80 autour des SGDR sont passés à un moment ou à un autre soit à Berkeley sur Ingres, soit à San José sur System/R.

Michael Stonebraker lui-même créa Ingres Inc pour commercialiser le code de son projet. En 1980, Roger Sippl et Laura King fondent Relational Database Systems (RDS et qui de-