

الجمهورية الجزائرية الديمقراطية الشعبية
RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
وزارة التعليم العالي والبحث العلمي
MINISTRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE
جامعة عمار ثلجي بالأغواط
UNIVERSITÉ AMMAR THÉLIDJI LAGHOUAT



كلية العلوم
FACULTÉ DES SCIENCES
قسم الرياضيات و الإعلام الآلي
DÉPARTEMENT DE MATHÉMATIQUE ET INFORMATIQUE

MÉMOIRE DE MASTER

DOMAINE : MATHÉMATIQUE ET INFORMATIQUE (MI)
FILIERE : INFORMATIQUE
OPTION : SYSTEME D'INFORMATION ET DE DECISION(SID)

PRÉSENTÉ PAR :
Zaza Faiza

Thème :

**LES LANGAGES D'INTERROGATION GRAPHIQUE DES
BASES DE DONNEES MULTIDIMENSIONNELLES**

Soutenu publiquement devant le jury composé de :

Mr. Y. GUELLOUMA	Université de Laghouat	(Président)
Mr. T. BEN DOUMA	Université de Laghouat	(Examineur)
Melle. Z. ABDELHAFIDI	Université de Laghouat	(Examineur)
Melle. N. HAMINI	Université de Laghouat	(Encadreur)

DEDICACES

*La femme la plus chère au monde, ma très chère mère, la source de tendresse qui a tout donné sans rien recevoir
(Je remercie du fond de mon cœur) ;*

*Le plus cher homme du monde, mon père la source de patience, la volonté, et le courage
(Je remercie du fond du mon cœur) ;*

*Mes Chères Sœurs, mes chers Frères et tous mes amis.
Vous tous qui lire cette page.*

Je dédie ce modeste travail.

Remerciements

Avant tous nous remercions le bon dieu qui nous a donné le courage et la force pour continuer. Merci de nous avoir éclairé le chemin de la réussite.

J'exprime toute la reconnaissance qui j'éprouve envers tous ceux qui ont contribué par leurs conseils et leurs encouragements à l'aboutissement de ce travail.

Je tenais à remercier tous les enseignants du Département Informatique

Je remercie sincèrement Mademoiselle Narejace Hamini mon encadreur, pour m'avoir encouragé à prendre ce sujet, et pour toute son aide.

Mes remerciements à toutes les personnes de l'université d'Amar TELIDJI Laghouat enseignants, administrateurs, et plus particulièrement, ceux du Département de Mathématiques et Informatique.

Sans oublier les membres de jury d'avoir accepter de se porter juge de ce travail.

À mes familles et mes amis, j'adresse un grand merci.

Résumé

Ce travail se situe dans le contexte des systèmes décisionnels et plus particulièrement, dans le domaine des bases de données multidimensionnelles.

Nous intéressons dans ce travail une étude comparative entre les langages d'interrogation de base de données multidimensionnelle (BDM), les principes de la manipulation OLAP pour nous proposons un nouvel opérateur OLAP.

Outre les opérateurs OLAP classiques - la rotation (rotate), la permutation d'une modalité dans une dimension (switch), le forage vers le haut ou vers le bas (roll up ou drill down), la sélection et sur un cube (select), la traduction des mesures en dimensions ou des dimensions en mesures (pull ou push) et l'opérateur cube - et les opérateurs inspirés du relationnel, peu d'opérateurs ont été proposés pour enrichir l'analyse en ligne en général.

Nous avons implémenté quelques d'opérations de manipulation OLAP et testé leurs performances sur exemple simple dans le but de comparer et interpréter les résultats obtenus.

Mots-clés : entrepôts de données, magasins de données, Base de données multidimensionnelles, modèle multidimensionnel, OLAP, MDX, OLAP-SQL.

Abstract

This work is in the context of decision support systems, and more particularly in the field of multidimensional databases.

We are interested in this work a comparative study between the query languages of the multidimensional database (MDB), the principles of handling OLAP we propose a new OLAP operator.

Besides the classical OLAP operators - rotation (rotate), switching from one modality in one dimension (switch), drilling up or down (roll up or drill down), the selection and projection on a cube (slice and dice), the translation of measures dimensions or dimensional measures (push or pull) and cube operator - and inspired by the relational operators, some operators have been proposed to enrich the analysis online General.

We have implemented some manipulation operations OLAP and tested their performance on real example in order to compare and interpret the results.

Key words: data warehouses, data marts, multidimensional data base, multidimensional model, OLAP, MDX, OLAP-SQL.

Table Des Matières

Remerciements	I
Résumé, Abstract	II
Introduction	1
<i>Chapitre I : Les bases de données multidimensionnelles</i>	
1. Introduction	3
2. Les systèmes d'aide à la décision	3
2.1 Architecture de systèmes d'aide à la décision	3
2.1.1 Les sources de données	4
2.1.2 L'entrepôt de données (data warehouse)	4
2.1.3 Les magasins de données (Data mart).....	4
2.1.4 Les outils d'analyse	3
2.1.5 Les outils ETL (Extract Transform Load)	4
3. Les entrepôts de données pour l'aide à la décision	4
4. Modélisation de l'entrepôt de données	6
5. La modélisation multidimensionnelle	6
5.1 Modélisation conceptuelle	7
5.1.1 Le fait	7
5.1.2 La dimension	8
5.1.3 Notion de hiérarchie	8
5.1.4 Modèles en étoile	9
5.1.5 Modèles en flocon	10
5.1.6 Modèles en constellation	11
5.1.7 La notion d'hypercube	12
6. Le processus d'analyse interactif (olap) Revue des concepts	13
6.1 OLTP	13
6.2 OLAP	13
6.3 OLTP vs. OLAP	14
6.4 Définition et Composantes D'OLAP	14
7. Modélisation logique	15
7.1 ROLAP	15
7.2 MOLAP	15
7.3 HOLAP	15

5.3 OLAP-SQL	38
6. Langage graphique	38
6.1 Manipulations OLAP Graphiques	40
6.2 GRAPHIC-OLAP	40
7. JPivot	41
7.1. Principales caractéristiques.....	42
7.2. Navigation multidimensionnelle avec JPivot/Mondrian	42
7.2.1 Gestion des dimensions.....	43
7.2.2 Affichage de la table	44
7.2.3 Navigation.....	45
7.2.4 Visualisation des données originales	45
7.2.5 Gestion des graphiques	46
7.2.6 Impression et export.....	46
8. Business Intelligence	47
8. I.1 Définition	47
8. I.2 Oracle Business Intelligence	47
8. I.3 Microsoft Business Intelligence	48
8. I.3.1 Microsoft Office	49
8. I.3.1.1 Outil client.....	49
8. I.3.1.2 Portail d'accès	49
8. I.3.1.3 Outil Fonctionnel.....	50
8. I.3.2 Socle Technique : SQL Server 2005	51
9. Conclusion	51
 <i>Chapitre III : Contribution</i>	
1. Existant	52
2. Le langage utilisé	53
3. Implémentation des opérateurs OLAP	53
4. Les étapes de l'exécution	53
5. L'opération proposée	58
 Conclusion	 59
 Bibliographie	

Table des figures

1.1 : Architecture d'un système d'aide à la décision	3
1.2 : L'entrepôt de données dans le système décisionnel	5
1.3 : Représentation multidimensionnelle	7
1.4 : Exemple de fait	7
1.5 : Exemple de dimensions	8
1.6 : type d'hiérarchies	8
1.7 : Modélisation en étoile	9
1.8 : Exemple de schéma en étoile	10
1.9 : Modélisation en flocon	11
1.10 : Modélisation en constellation	11
1.11 : Hypercube	12
1.12 : Multicubes	12
1.13 : Cube de données	13
1.14 : Architecture ROLAP.....	16
1.15 : Architecture MOLAP	16
1.16 : Architecture HOLAP	17
2.1 Notion de tranche du cube de données	20
2.2 : Exemple d'un tableau multidimensionnel	20
2.3 : Principe du forage	21
2.4 : Le forage vers le haut (« roll-up ») et le forage vers le bas (« drill-down »)	21
2.5 : Principe de rotation.	21
2.6 : Rotation de fait (« FRotate»)	22
2.7 : Rotation de Dimension (« DRotate»)	22
2.8 : Rotation de l' hiérarchie (« HRotate»)	23
2.9 : Fonctionnement de l'éditeur OLAP-SQL	38
2.10 : Interfaces de visualisation dans GRAPHIC-OLAP.....	39
2.11 : Construction graphique d'une table multidimensionnelle	39
2.12 : Construction d'une requête	40
2.13 : Architecture de GRAPHIC-OLAP	41
2.14 : Le tableau des données décisionnelles JPivot	42
2.15 : Oracle Business Intelligence Discoverer exemple de tableau de bord.....	47
2.16 : Exemple Oracle Reports	48
2.20 : Microsoft Business Intelligence	49

3.1 : Schéma d'une constellation de la BDM.	53
3.2 : Le choisir du fait	54
3.3 : Sélectionner les dimensions	54
3.4 : La table multidimensionnelle de fait VENTE.	55
3.5 : Les graphes de la TM.	55
3.6 : Le forage vers le bas (« drill-down »)	56
3.7 : Rotation de fait VENTES (« FRotate»)	56
3.8 : Rotation de Dimension Clients (« DRotate»)	57
3.9 : Opérateur Select	57
3.10 : Opérateur Cube	58
3.11 : Opérateur Proposer	59

Liste des tableaux

1.1 : Tableau bidimensionnel.	6
1.2 : OLTP vs OLAP	14
1.3 : Résumé des caractéristiques des architectures ROLAP, MOLAP et HOLAP.....	17
2.1 : Synthèse des travaux sur les langages de manipulation OLAP.....	24
2.2 : les instructions CREATE, ALTER et DROP	34
2.3 : les instructions SELECT, INSERT et DELETE	34
2.4 : les instructions GRANT et REVOKE	35
2.5 : MDX versus SQL	35
2.6 : Fonctions sur les "sets"	37
3.1 : tous les opérateurs d'OLAP	52

Introduction

Les entrepôts de données se sont développés afin de fournir aux décideurs des systèmes dédiés à l'analyse des données et de pallier les insuffisances des systèmes transactionnels (OLTP OnLine Transactionnel Processing). Pour faciliter l'accès aux données, plusieurs entrepôts ont adopté l'approche des bases de données multidimensionnelles (BDM).

Les BDM ont donc émergé pour répondre aux besoins spécifiques d'analyse multidimensionnelle (OLAP On-Line Analytical Processing) [Cod93]. Les BDM sont généralement décrites selon un schéma multidimensionnel comportant des faits (regroupant les mesures d'activité) et différents axes appelés dimensions [Kimball, 1996]. Les dimensions regroupent les paramètres d'analyse organisés selon des hiérarchies. Une hiérarchie décrit différents niveaux de granularité entre les paramètres. Cette structure hiérarchique permet d'analyser les données au travers du cube multidimensionnel [Gray et al., 1996].

Les données d'une BDM sont interrogées via la technologie OLAP à l'aide d'outils graphiques ou suivant un langage textuel [Ravat et al. 2007, 2008].

Une requête OLAP est une requête multidimensionnelle permettant d'agréger les données d'une ou de plusieurs mesures d'un fait suivant les attributs d'une ou plusieurs dimensions.

Multidimensionnelles OLAP analyses consistent à explorer les bases de données multidimensionnelles de manière interactive par le forage, la rotation, la sélection et l'affichage des données. Bien qu'il n'y ait pas de consensus sur un tronc commun d'un ensemble minimal d'opérations pour une algèbre multidimensionnelle.

L'objectif de ce mémoire est de faire une étude comparative entre les langages d'interrogation de base de données multidimensionnelle (BDM), les principes de la manipulation OLAP pour nous proposons un nouvel opérateur OLAP. Les opérations de manipulation OLAP font l'objet de nombreux travaux de recherche dont le but est la manipulation de données multidimensionnelles.

Dans le cadre de notre travail, nous avons organisé ce mémoire en trois chapitres.

Le premier vise à présenter et à définir les différents concepts servant de support à nos travaux. Nous étudions les composants d'un système d'aide à la décision (SAD), d'un entrepôt de données (ED), d'un magasin de données (MD), La modélisation multidimensionnelle et le système OLAP (On Line Analytical Processing). Nous analysons également les différents travaux à chacun de ces composants

Dans le deuxième chapitre, nous nous intéressons à l'état de l'art et aux travaux de recherche sur les langages d'interrogation de base de données multidimensionnelle (BDM). Nous discutons sur ces travaux et expliquons les principes de la manipulation de données multidimensionnelles. Nous allons présenter les principaux composants décisionnels disponibles en open source et quelques des langages textuels et graphiques.

Dans le troisième chapitre, Nous avons expliqué les principes d'opérations de manipulation OLAP et présenter l'implémentation de quelques d'opérations de manipulation OLAP. Nous avons essayé de proposer nouvel opérateur OLAP. À la fin nous avons interprété et comparé les résultats trouvés après l'exécution de ces opérateurs OLAP et cité quelques inconvénients ce nouvel opérateur OLAP.

La conclusion de ce mémoire synthétise les études faites durant la réalisation de ce mémoire. Ainsi que les domaines d'application d'opérations de manipulation OLAP.

CHAPITRE I

Les bases de données
multidimensionnelles

1. Introduction

Les bases de données et les systèmes de gestion de bases de données jouent un rôle vital dans le développement et le succès d'entreprises. Le modèle le plus utilisé au sein d'entreprise est le modèle relationnel, il permet de stocker et de manipuler des importants volumes de données. L'exploitation de ces données pour l'analyse et la prise de décision est réalisée souvent par des outils de reporting (requêtes), graphiques, tableaux...

Le décideur dans le monde transactionnel, pose des questions relativement simples par exemple <<quel est le chiffre d'affaires du mois derniers >>. Les besoins du décideur sont totalement différents. IL ne pose pas la question précédente, mais <<quel est le chiffre d'affaires par produit et par zone géographique pour les trois derniers mois >>. Cette requête est complexe, lente et pénalise le système de production. Face à ce problème les entreprises sont recourus à des systèmes d'aide à la décision spécifiques, basés sur l'approche multidimensionnel (OLAP).

2. Les systèmes d'aide à la décision

Un système d'information permet de faciliter la mise en œuvre des stratégies de l'entreprise. Le rôle d'un système d'aide à la décision est plutôt d'aider à déterminer les bonnes stratégies. Plus précisément, un système d'aide à la décision permet de développer la capacité de réflexion et d'action de l'entreprise en aidant à l'apprentissage (analyse et suivi des activités précédentes) et au pilotage des plans d'actions (prévision et planification des activités futures).

Un système d'aide à la décision est l'ensemble des outils matériels et logiciels qui permettent de collecter, de stocker et d'analyser des données issues du système d'information des entreprises dans le but de faciliter la prise de décision par les décideurs. Les décideurs, utilisateurs de ces systèmes, sont des experts d'un métier chargés d'analyser les données décisionnelles pour le pilotage de l'organisation. Ils sont généralement non informaticiens. Dans la suite, nous les désignerons simplement par le terme usagers.

2.1 Architecture de systèmes d'aide à la décision

La conception d'un système d'aide à la décision doit être basée sur la séparation entre deux espaces de stockage : l'entrepôt qui regroupe toute l'information décisionnelle et les magasins qui contiennent une partie de cette information, dédiée à un thème, un métier, ou une analyse. Nous définissons l'architecture des systèmes d'aide à la décision décrite à la Figure 1.1. Cette architecture est composée de deux niveaux de stockage (entrepôt et magasin de données) et de deux niveaux fonctionnels (outils ETL et outils de restitution et d'analyse).

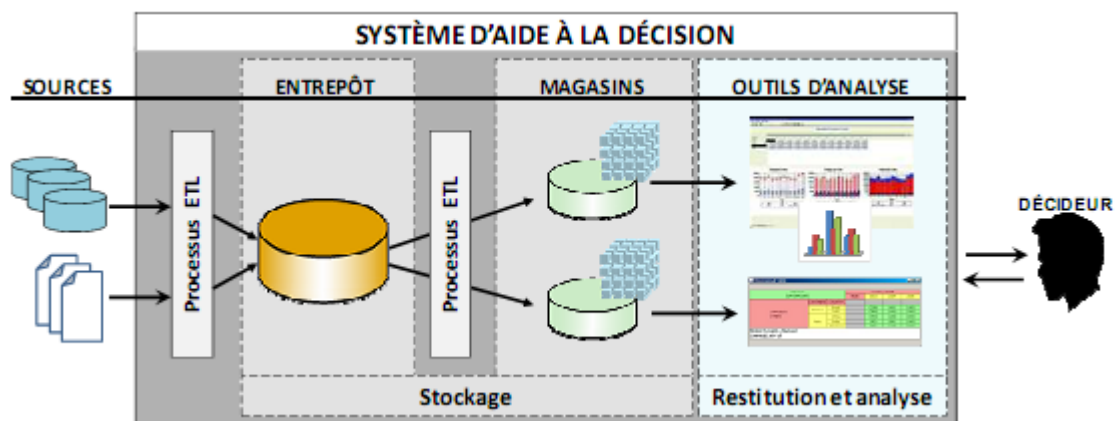


Figure 1.1 : Architecture d'un système d'aide à la décision

2.1.1 Les sources de données

Les sources de données sont nombreuses et variées :

- les bases de production (différents format, réparties...). Ces bases peuvent enregistrer l'ensemble des informations formelles.
- les informations externes à l'entreprise, mais nécessaires aux décideurs telles celles obtenues au cours de forums, salons, séminaires, auprès d'un expert, internet. Ces informations entrant plutôt dans la catégorie informelle.

2.1.2 L'entrepôt de données (data warehouse)

L'entrepôt de données est le lieu centralisé pour stocker les informations décisionnelles. C'est une base de données conçue spécifiquement pour l'analyse et l'interrogation (requêtes complexes). Il collecte les données provenant des différentes sources et garde leur historique (conserve l'évolution de données).

2.1.3 Les magasins de données (Data mart)

Les magasins de données sont des extraits de l'entrepôt orientés sujet. Les données sont organisées de manière adéquate pour permettre des analyses rapides à des fins de prise de décision.

2.1.4 Les outils d'analyse

Les outils d'analyse sont des outils d'accès aux données des magasins. Ils permettent de manipuler les données suivant des axes d'analyses (Modèle multidimensionnel). L'information est visualisée au travers d'interfaces interactives et fonctionnelles dédiées à des décideurs souvent non informaticiens (directeurs, chefs de services,...) pour leur aider de prendre les décisions. Ces outils représentent la partie visible dans le système décisionnel pour les utilisateurs (décideurs). Ils offrent de différentes représentations de l'information: tableaux, graphiques, courbe de prévision (statistique).

2.1.5 Les outils ETL (Extract Transform Load)

Prendent en charge l'une des fonctions les plus essentielles du système globale décisionnel. Il s'agit en et de gérer toutes les étapes préalables au chargement effectif des données dans le Data Warehouse. **Extraire** : Accéder à la majorité des systèmes de stockage de données (SGBD, ERP, fichiers à plat...) et récupérer les données identifiées.

Transformer : Transformer, reformater et nettoyer les données an d'éliminer les valeurs non conforme au modèle de destination et d'éviter les doublons et autres incohérences.

Charger : Insérer les données dans le Data Warehouse. Où elles sont mises à disposition des outils d'analyse Data Mining, l'analyse multidimensionnelle OLAP, etc.

3. Les entrepôts de données pour l'aide à la décision

La définition classique d'un entrepôt donnée par [Widom,95 ; Inmon,96]

"Un **Entrepôt de Données** (ED) est une collection de données orientées sujet, intégrées, non volatiles, historisées, résumées et disponibles pour l'interrogation et l'analyse". Cette définition met l'accent sur les caractéristiques suivantes :

-Orientées sujet : Les données des entrepôts sont organisées par sujet plutôt que par application. Par exemple, une chaîne de magasins d'alimentation organise les données de son entrepôt par rapport aux ventes qui ont été réalisées par produit et par magasin, au cours d'un certain temps.

-Intégrées : Les données provenant des différentes sources doivent être intégrées, avant leur stockage dans l'entrepôt de données. L'intégration (mise en correspondance des formats, par exemple), permet d'avoir une cohérence de l'information.

-Non volatiles : A la différence des données opérationnelles, celles de l'entrepôt sont permanentes et ne peuvent pas être modifiées. Le rafraîchissement de l'entrepôt, consiste à ajouter de nouvelles données, sans modifier ou perdre celles qui existent.

-historisées : La prise en compte de l'évolution des données est essentielle pour la prise de décision qui, par exemple, utilise des techniques de prédiction en s'appuyant sur les évolutions passées pour prévoir les évolutions futures.

-Résumées: les informations issues des sources de données doivent être agrégées et réorganisées afin de faciliter le processus de prise de décision.

-Disponibles pour l'interrogation et l'analyse: les utilisateurs doivent pouvoir consulter les données réorganisées de l'entrepôt en fonction de leurs droits d'accès au travers d'outils interactifs d'aide à la manipulation et l'analyse.

La figure suivante illustre l'emplacement d'un entrepôt de données dans l'architecture de système décisionnel.

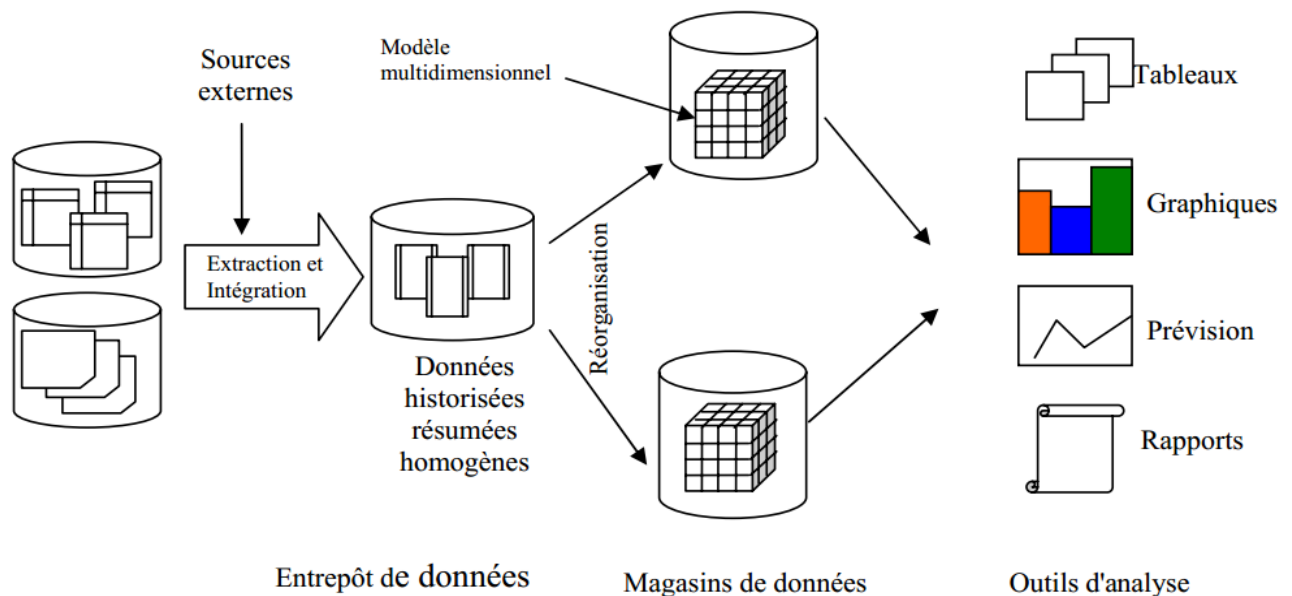


Figure 1.2: L'entrepôt de données dans le système décisionnel

De plus, l'entrepôt de données offre à l'entreprise les avantages suivants :

- Il constitue une collection de données centralisée disponible pour l'aide à la décision (OLAP, datamining,....).
- Les évolutions des données de l'entrepôt sont conservées (historisation des données).
- Il contient un ensemble de données consolidées (données homogènes et fiables).
- Il contient des données agrégées permettant une analyse à différent niveaux de détails.
- Il permet de développer différents thèmes d'analyse (réorganisation en fonction des sujets à analyser).

4. Modélisation de l'entrepôt de données

Les modèles utilisés pour la conception, des entrepôts de données reposant sur une conception multidimensionnelle des données. C'est à dire que les objets à modéliser sont décrits par plusieurs attributs différents; indépendants à priori; qui permettent chacun en particulier de pouvoir classifier et/ou ordonner ces objets. Prenons par exemple un entrepôt de données d'une entreprise commerciale multi-sites. Les informations pertinentes concernant les informations sur les sites. Sur les produits, sur le temps calendaire. Le vocabulaire utilisé est alors la dimension: dimension produit ; dimension temporelle , dimension site chaque dimension possède sa propre table. Les tables de dimension sont regroupées autour d'une table contenant ces informations (mesures). On parle de schéma en étoile.

5. La modélisation multidimensionnelle

Il existe deux façons d'analyser, soit par reporting, soit par une analyse on-line (c'est ce qui nous intéresse ici). Une société désire construire une base de données pour suivre l'évolution de ses ventes par mois et par catégories. Elle dispose d'une table qui contient les ventes.

Mois	Catégorie	Quantité	Montant
Janvier 2000	Alimentation	15	150.00
Janvier 2000	Nettoyage	10	200.00
...

Pour analyser ces données, on peut par exemple placer les mois en colonne et les catégories en ligne.

Mois/Catégorie	Nettoyage	Alimentation	Vêtement	Papeterie
janvier	200.00	150.00	180.00	50.00
février	180.00	140.00	250.00	55.00
Mars	220.00	155.00	100.00	48.00

Tableau 1.1 : Tableau bidimensionnel.

Il s'agit ici d'un tableau à deux dimensions et non d'une table !

Si l'on veut comparer les ventes par catégories, par région et pour plusieurs années, nous considérons plusieurs tables, relatives aux ventes de chaque année par exemple entre 2004 et 2006, On peut alors observer les données dans un espace à trois dimensions : la dimension géographie, la dimension produits et la dimension temps. Chaque intersection de ces dimensions représente une cellule comportant le total des ventes. On parle alors de cube de données .

La figure 1.3 montre la représentation multidimensionnelle (cube de données) permettant l'analyse des ventes pour les derniers trois années, par région et par catégorie.

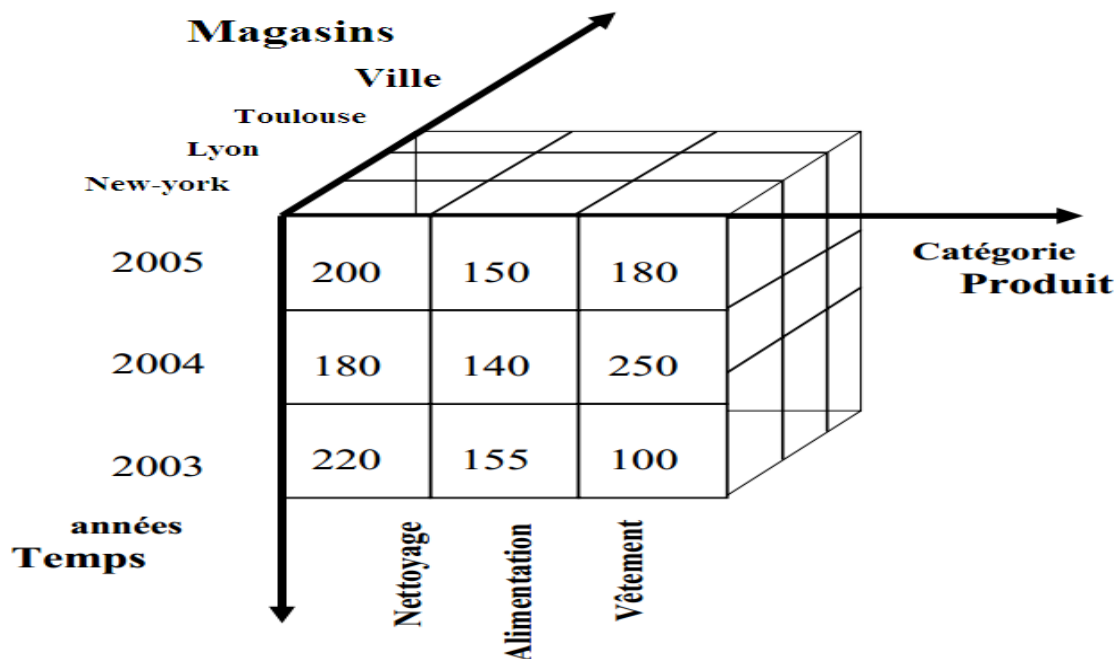


Figure 1.3 : Représentation multidimensionnelle

Les analyses décisionnelles sont basées sur des traitements OLAP directement liés à la modélisation de l'information sous une forme conceptuelle proche de la perception qu'en a l'analyste. Cette perception de l'information est basée sur une vision multidimensionnelle des données. Cette modélisation multidimensionnelle consiste à considérer un sujet analysé comme un point dans un espace à plusieurs dimensions (axes d'analyses). Les données sont organisées de manière à mettre en évidence le sujet analysé et les différentes perspectives de l'analyse.

Dans notre exemple le sujet analysé est le vente qui prend comme mesure le montant des ventes et les dimensions (axes d'analyses) sont : la dimension Temps, la dimension Géographie et la dimension Produit.

5.1 Modélisation conceptuelle

La modélisation multidimensionnelle repose sur le concept de fait et de dimension.

5.1.1 Le fait: Le fait représente le sujet analysé qui est formé de mesures d'activité analysée, les mesures sont généralement numériques pour permettre de résumer un grand nombre d'enregistrements (les additionner, calculer le minimum, le maximum,...). Ils sont aussi additifs ou semi-additifs pour pouvoir les combiner au moyen d'opérateurs arithmétiques.

Exemple : le fait de Vente de notre exemple peut être constitué des mesures d'activités suivantes : quantité de produits vendus et montant des ventes. On représente le fait par un rectangle qui contient le nom du fait et les mesures.

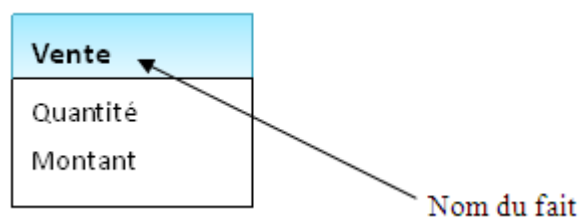


Figure 1.4 : Exemple de fait

5.1.2 La dimension : Le fait est analysé suivant différentes perspectives. Ces perspectives correspondent à une catégorie utilisée pour caractériser les mesures d'activité analysées ; c'est-à-dire les dimensions. Une dimension se compose de paramètres (attributs). Chaque paramètre peut prendre différentes valeurs faisant varier les mesures de l'activité.

Exemple: Poursuivons l'exemple précédent, les dimensions sont : Temps, Produit et Magasin. La dimension temps peut prendre plusieurs paramètres par exemple jour, mois, trimestre et année.

Dans le modèle multidimensionnel les paramètres sont essentiellement discrets et leurs valeurs sont bien déterminées.

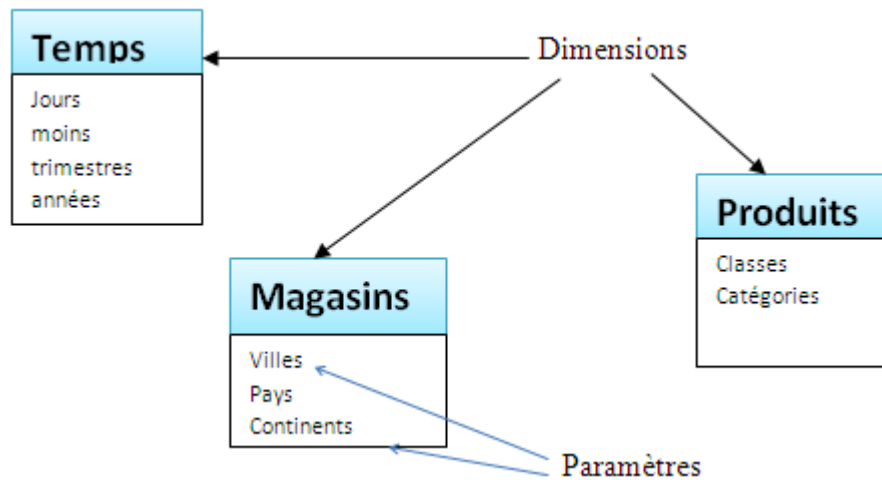


Figure 1.5 : Exemple de dimensions

5.1.3 Notion de hiérarchie

Lors du processus OLAP, les paramètres d'un dimension sont organisés en une ou plusieurs hiérarchie(s) pour permettre d'analyser les données en partant d'un faible niveau de détail vers des données plus détaillées "forage vers le bas", ou réciproquement d'un niveau détaillé vers un niveau supérieur " forage vers le haut ". La hiérarchie spécifie une relation d'ordre entre les paramètres.

Exemple: pour la dimension Magasin Les paramètres des dimensions sont organisés suivant une hiérarchie :

villes → Pays → Continents

Ainsi chaque ville appartient à une Paye qui est situé dans une Continent.

- Hiérarchie simple ou multiple

Dans le cas d'une hiérarchie simple; chaque paramètre n'a qu'un seul paramètre supérieur. Dans le cas d'hiérarchie multiple, on a différents chemins pour créer une hiérarchie.

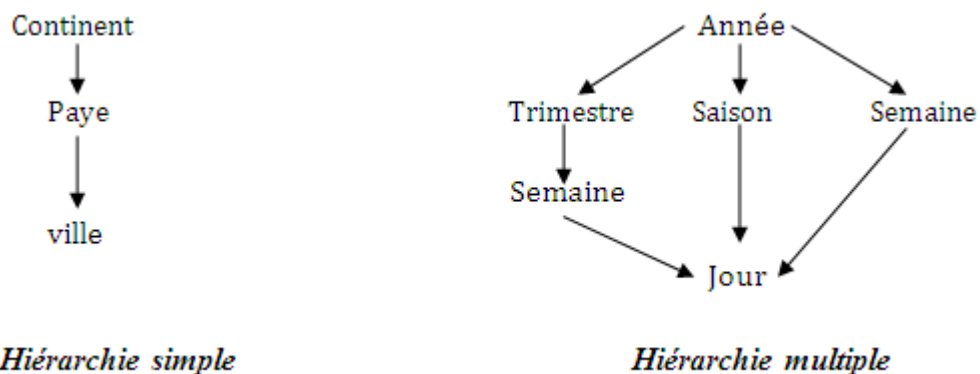


Figure 1.6 : type d'hiérarchies.

5.1.4 Modèles en étoile

Il est possible d'établir un modèle multidimensionnel simple à partir du fait et des dimensions. Ce modèle est constitué du fait central et des dimensions représentant visuellement une étoile, on parle de modèle en étoile.

a. Avantages:

- Facilité de navigation
- Nombre de jointures limité

b. Inconvénients:

- Redondance dans les dimensions
- Toutes les dimensions ne concernent pas les mesures

Exemple: On peut modéliser les analyses des quantités et des montants des ventes selon trois dimensions : le Temps, le Produit et le Magasin par le schéma en étoile décrit dans La Figure 1.7.

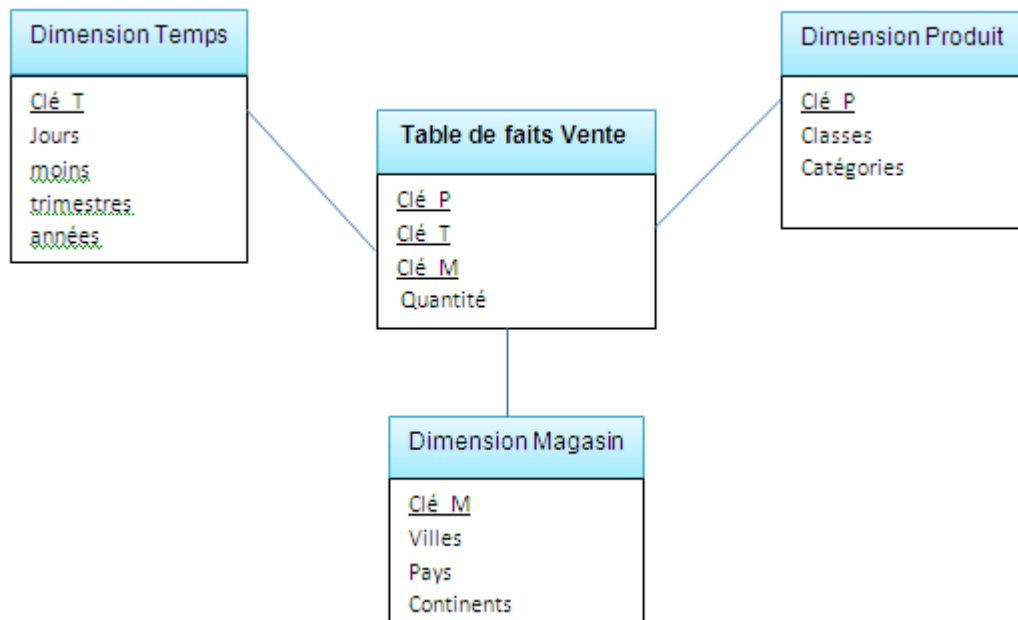


Figure 1.7: Modélisation en étoile.

Exemple. L'exemple de la figure suivante correspond à la modélisation d'une analyse de ventes en fonction des produits et des magasins au cours du temps. Le sujet est modélisé par le fait composé des mesures quantité et montant. Les axes de l'analyse sont représentés par les dimensions produits, magasins et temps. La dimension magasins est caractérisée par trois paramètres villes, pays et continents organisés hiérarchiquement : le paramètre villes représente une graduation plus fine (permettant d'observer le sujet plus finement) que la graduation pays, elle-même étant une graduation plus fine de continents.

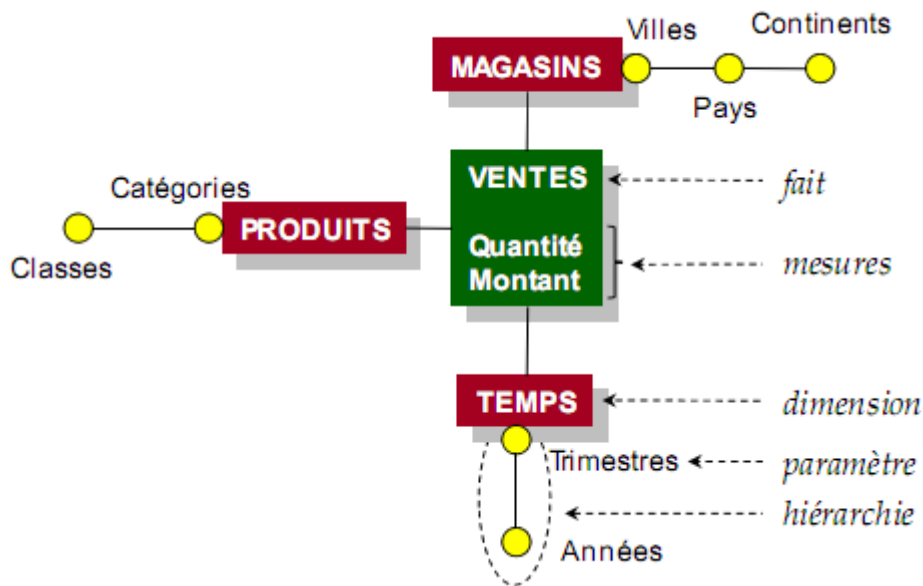


Figure 1.8 : Exemple de schéma en étoile.

5.1.5 Modèles en flocon

Il existe d'autres types de modélisation multidimensionnelle, la modélisation en flocon consiste à décomposer les dimensions du modèle en étoile en sous hiérarchies. La modélisation en flocon est donc une extension de la modélisation en étoile ; le fait est conservé et les dimensions sont éclatées selon leurs hiérarchies des paramètres. Cette modélisation permet de formaliser une hiérarchie au sein d'une dimension.

a. Avantages:

- Normalisation des dimensions
- Économie d'espace disque

b. Inconvénients:

- Modèle plus complexe (jointure)
- Requêtes moins performantes

Exemple: La Figure 1.8 illustre la modélisation en flocon en normalisant les dimensions du modèle en étoile de la Figure 1.7.

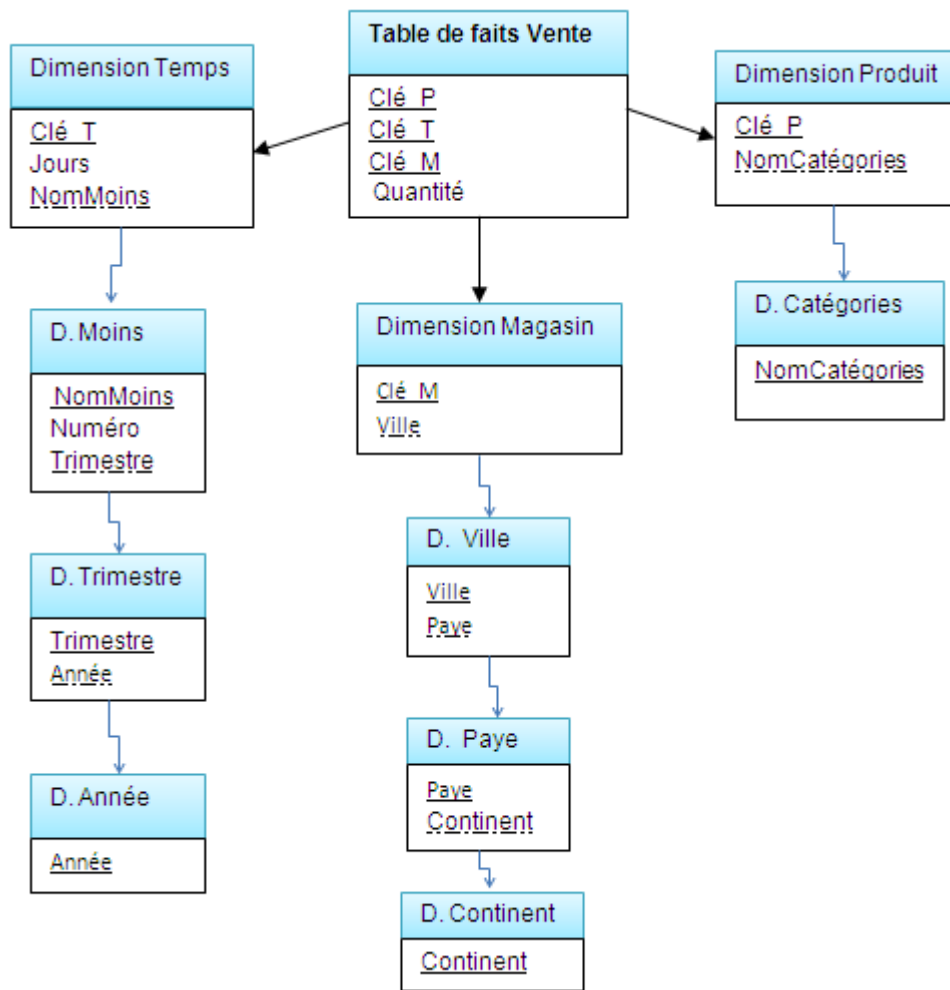


Figure 1.9 : Modélisation en flocon.

5.1.6 Modèles en constellation

Une autre technique de modélisation, issue du modèle en étoile, est la modélisation en constellation. Il s'agit de fusionner plusieurs modèles en étoile qui utilisent des dimensions communes. Un modèle en constellation comprend donc plusieurs faits et des dimensions communes ou non.

Exemple: La Figure 1.10 illustre la modélisation en constellation de deux schémas en étoile : l'un analyse les ventes et l'autre correspond aux fournisseurs.

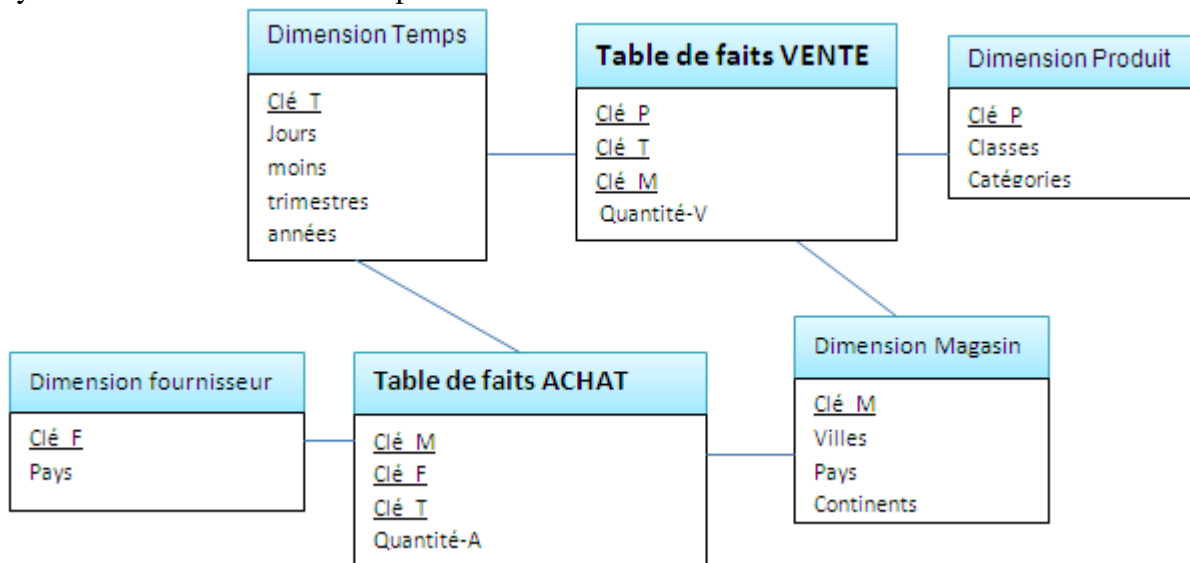


Figure 1.10: Modélisation en constellation.

5.1.7 La notion d'hypercube

- Hypercubes et multicubes

Bases de données multidimensionnelles peuvent présenter leurs données à une application utilisant deux types de cubes: hypercubes et multicubes. Dans le modèle hypercube, comme le montre l'illustration ci-dessous, toutes les données apparaît logiquement comme un seul cube. Toutes les parties des collecteurs représentés par ce hypercube ont dimensionnalité identique.

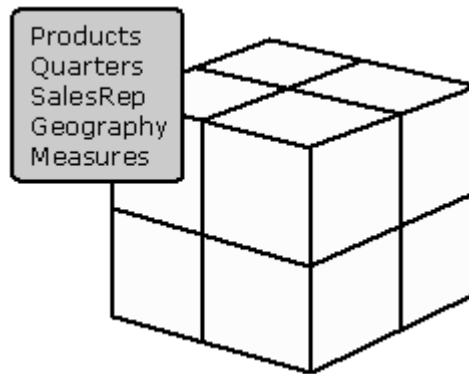


Figure 1.11. Hypercube

Dans le modèle multicube, de données est segmentée en une série de petits cubes, dont chacun est composé d'un sous-ensemble des dimensions disponibles, comme indiqué dans le schéma suivant:

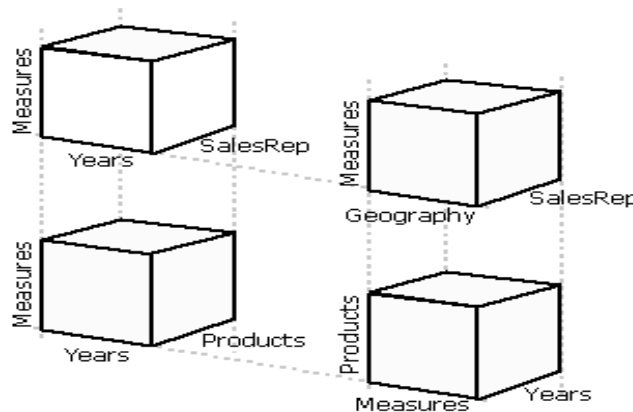


Figure 1.12. Multicubes

Hypercubes et multicubes diffèrent en termes de métadonnées disponibles. Dans un hypercube, chaque dimension appartient à un cube seulement. Une dimension est "possédée" par l'hypercube. Dans un multicube, une dimension peut faire partie de plusieurs cubes. Autrement dit, les dimensions ne sont pas possédées par tout un cube, mais plutôt, ils sont disponibles pour tous les cubes. En outre, il peut y avoir des dimensions qui n'appartiennent pas à un cube.

La figure suivante présente un cube de données formé de montants de vente en cellules et de trois arêtes graduées respectivement par des catégories de produits, des villes de magasins et des trimestres. La notion de cube de données ne se limite pas à trois axes mais se généralise en hyper-cube où le nombre d'axes est quelconque pouvant aller jusqu'à plusieurs dizaines.

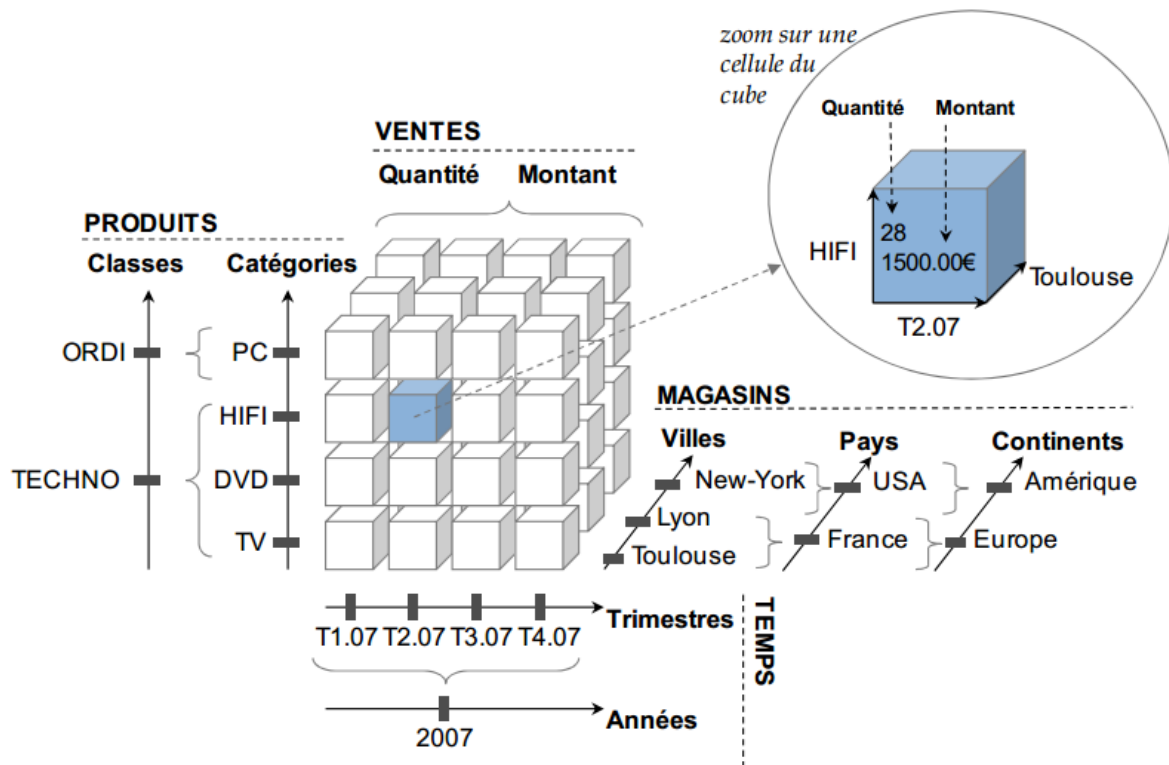


Figure 1.13 : Cube de données.

6. Le processus d'analyse interactif (olap) Revue des concepts

On pourrait se demander pourquoi ne pas utiliser un SGBD pour réaliser cette structure d'informatique décisionnelle. En réalité SGBD et datawarehouse ont des objectifs différents. Ils stockent les données de manière différentes et font l'objet de requêtes différentes. Ils sont ainsi basés sur deux systèmes différents : OLTP et OLAP

6.1 OLTP

OLTP (On Line Transaction Processing) est le modèle utilisé par les SGBD. Le mode de travail est transactionnel. L'objectif est de pouvoir insérer, modifier et interroger rapidement et en sécurité la base. Ces actions doivent pouvoir être effectuées très rapidement par de nombreux utilisateurs simultanément. Chaque transaction travail sur de faibles quantités d'informations, et toujours sur les versions les plus récentes des données.

6.2 OLAP :

Les datawarehouses eux reposent sur le système OLAP (On Line Analytical Processing). Ce système travail en lecture seulement. Les programmes consultent d'importantes quantités de données pour procéder à des analyses. Les objectifs principaux sont regrouper, organiser des informations provenant de sources diverses, les intégrer et les stocker pour donner l'utilisateur une vue orientée métier, retrouver et analyser l'information facilement et rapidement. Cela nécessite de consulter des versions historiques de la base et peut se permettre d'ignorer temporairement les dernières mises à jour. Ces bases sont souvent d'un ordre de grandeur nettement supérieur à celle des bases OLTP, du fait de la conservation de l'historique.

6.3 OLTP vs. OLAP :

	Caractéristiques	OLTP	OLAP
Conception	Orientation	Transaction	Analyse
	Conception	Entité-Relation	Etoile/flocon
Données	Granularité	Détail	Résumées, agrégées
	Nature	Relationnelle	Multidimensionnelle
	Actualisation	Actualisées, mises à jour	Historisées , Recalculées
	Taille	100 Mo/Go	100 Go/To
Traitements	Unité de travail	Transaction simple	Requête complexe
	Accès	Lecture/écriture	Lecture
	Nb de tuples accédés	Dizaines	Million
	Métrique	Débit de transactions	Temps de réponse
Utilisateurs	Utilisateur	Agent opérationnel	Analyste/décideur
	Nombre d'utilisateurs	Milliers	Centaines

Table 1.2 : OLTP vs OLAP

6.4 Définition et Composantes D'OLAP

OLAP est défini comme « une catégorie de logiciels axés sur l'exploration et l'analyse rapides des données selon une approche multidimensionnelle à plusieurs niveaux d'agrégation » [Caron, 1998]. L'architecture d'un système OLAP est constituée de trois services : la base de données, le serveur et le module client [Bédard et al. 1997].

La base de données d'une architecture OLAP doit posséder une structure multidimensionnelle. Celle-ci peut être implantée à l'intérieur d'un système de gestion de base de données (SGBD) relationnel (et objet-relationnel) ou multidimensionnel. Lorsqu'un SGBD relationnel est utilisé, les données doivent être structurées selon une approche multidimensionnelle en utilisant des schémas particuliers. Il s'agit principalement des schémas en étoile et en flocon, Lorsqu'un SGBD multidimensionnel est utilisé, les données doivent être chargées dans la structure multidimensionnelle propriétaire à partir d'une base de données relationnelle ou d'un fichier texte. La base de données d'une architecture OLAP doit pouvoir supporter des données agrégées.

Le serveur OLAP est le logiciel qui gère la base de données OLAP et l'accès des utilisateurs à celle-ci. Il comprend un engin de calcul et permet habituellement à plusieurs utilisateurs concurrents d'effectuer des requêtes de type agrégatif sur la base de données. Le serveur OLAP est conçu spécifiquement pour manipuler des données structurées de façon multidimensionnelle.

Le module client est un logiciel d'accès, de manipulation et d'exploration des données de nature multidimensionnelle. Il permet d'effectuer des requêtes, d'obtenir des données du serveur OLAP et de les visualiser en deux ou trois dimensions. Le module client propose habituellement plusieurs modes de visualisation, allant du simple tableau aux différents types de diagrammes tels que les diagrammes à barres ou les diagrammes circulaires, pour ne nommer que les plus courants.

7. Modélisation logique

Au niveau logique plusieurs possibilités sont envisageables pour la modélisation multidimensionnelle. Il est possible d'utiliser :

- un système de gestion de bases de données (SGBD) relationnelles (ROLAP)
- un système de gestion de bases de données multidimensionnelles (MOLAP)
- un système de gestion de bases de données hybrides (HOLAP).

7.1 ROLAP

L'approche la plus couramment utilisée consiste à utiliser un système de gestion de bases de données relationnelles, on parle de l'approche ROLAP ("Relational On-Line Analytical Processing"). Le modèle multidimensionnel est alors traduit de la manière suivante :

- chaque fait correspond à une table, appelée table de fait,
- chaque dimension correspond à une table, appelée table de dimension.

Ainsi, la table de fait est constituée d'attributs représentant les mesures d'activité et les attributs clés étrangères de chacune des tables de dimension. Les tables de dimension contiennent les paramètres et une clé primaire permettant de réaliser des jointures avec la table de fait.

7.2 MOLAP

Le terme MOLAP ("Multidimensional On-Line Analytical Processing") désigne plus spécifiquement une technologie de stockage cartésien. MOLAP s'oppose à ROLAP. Pour le premier, les jointures sont déjà faites, ce qui explique les performances. Dans le second, les jointures entre les tables de dimension et de fait sont effectuées au moment de la requête.

Les structures multidimensionnelles utilisées sont des tableaux à n dimensions. Cette approche permet de stocker les données de manière multidimensionnelle. L'intérêt est que les temps d'accès sont optimisés, mais cette approche nécessite de redéfinir des opérations pour manipuler ces structures multidimensionnelles.

7.3 HOLAP

Ce terme signifie "Hybrid On-Line Analytical Processing" et désigne les outils d'analyse multidimensionnelle qui récupèrent les données dans des bases relationnelles ou multidimensionnelles, de manière transparente pour l'utilisateur. C'est en quelque sorte une association des principes MOLAP et ROLAP.

8. Architectures OLAP

Le type de base de données gérée par le serveur OLAP définit le type de serveur et ainsi le type d'architecture OLAP. Il existe trois principales architectures : ROLAP, MOLAP et HOLAP. La présente section contient un survol des caractéristiques importantes de ces architectures.

8.1 ROLAP

Une telle architecture se compose d'une base de données relationnelle qui est structurée selon un schéma multidimensionnel. Cette base de données se compose des données détaillées et des données agrégées de l'entrepôt qui sont stockées sous forme de tables dans une base de données relationnelle. Le serveur ROLAP extrait les données de la base de données à l'aide de requêtes SQL [Bédard et al. 1997], les interprète selon une vue multidimensionnelle, et les transfère au module client.

La figure 1.14 présente le fonctionnement de l'architecture ROLAP.

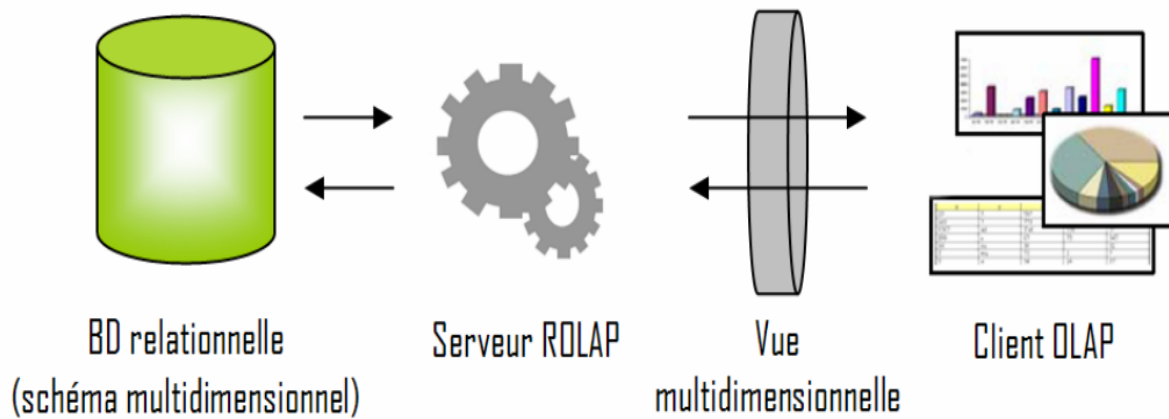


Figure 1.14 : Architecture ROLAP

8.2 MOLAP :

Les données détaillées ainsi que les données agrégées sont stockées dans une base de données multidimensionnelle (souvent appelée cube ou hypercube). Une base de données multidimensionnelle utilise une structure propriétaire au logiciel utilisé. Le serveur MOLAP extrait les données de l'hypercube et les présente correctement au module client.

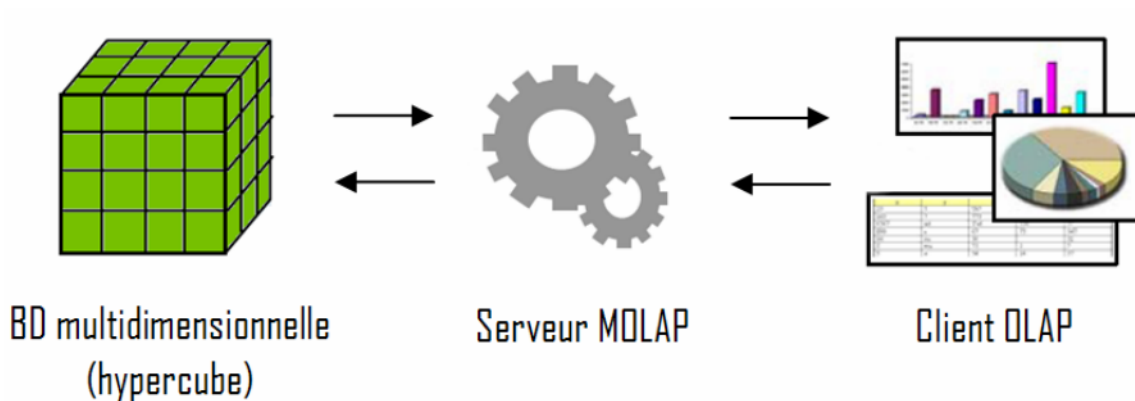


Figure 1.15 : Architecture MOLAP

8.3 HOLAP :

Cette architecture est un croisement des deux architectures précédentes. Les données détaillées sont stockées dans une base de données relationnelle et les données agrégées sont stockées dans une base de données multidimensionnelle. Le serveur HOLAP accède à ces deux bases de données et les présente au module client, selon une vue multidimensionnelle dans le cas des données de la BD relationnelle.

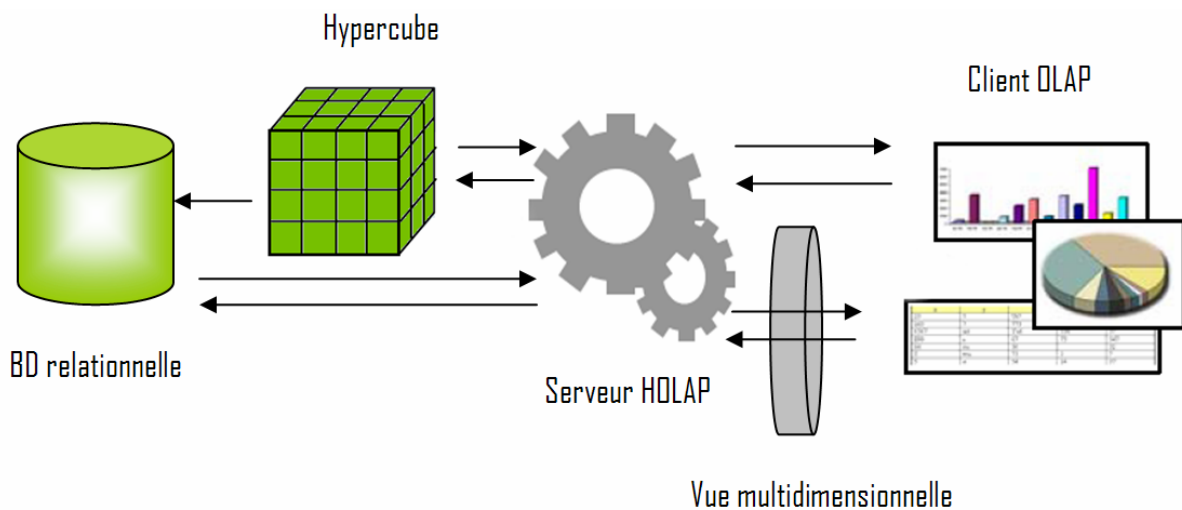


Figure 1.16 : Architecture HOLAP

8.4 Comparaison des architectures ROLAP, MOLAP et HOLAP

Le tableau 1.3 résume les caractéristiques de chacune des architectures OLAP.

Critère de comparaison	ROLAP	MOLAP	HOLAP
Stockage des données de base (détaillées)	BD relationnelle	BD multidimensionnelle	BD relationnelle
Stockage des agrégations	BD relationnelle	BD multidimensionnelle	BD multidimensionnelle
Performance des requêtes (en ordre croissant de performance, 3 correspondant à la meilleure performance)	1	3	2
Espace de stockage nécessaire (en ordre croissant d'espace supplémentaire nécessaire, 3 correspondant au plus grand volume)	3	2 (si des technologies de compression sont utilisées)	1 (si des technologies de compression sont utilisées)
Maintenance de la structure (en ordre croissant du niveau de maintenance nécessaire)	1	2	1

Tableau 1.3: Résumé des caractéristiques des architectures ROLAP, MOLAP et HOLAP

9. Conclusion

Les entrepôts de données forment ainsi le socle des systèmes d'information décisionnels et sont le support de l'analyse multidimensionnelle en ligne (On-Line Analytical Processing - OLAP) [Cod93].

Les entrepôts de données répondent donc à un réel besoin en matière d'accès à l'information résumée. Les offres commerciales actuelles proposent des logiciels d'extraction d'information et d'analyse de données

Les systèmes OLTP assurent une gestion optimisée des données (notamment en limitant la redondance des données). Dans les systèmes OLAP, une base de données multidimensionnelle est organisée pour faciliter l'analyse des données par des utilisateurs finaux, les décideurs.

Dans le chapitre suivant on va présenter les travaux réalisés dans les langages d'interrogation de base de données multidimensionnelle (BDM) et la manipulation OLAP.

CHAPITRE II

Les langages d'interrogation
de bases de données
multidimensionnelles

1. Introduction :

Nous présentons dans ce chapitre un état de l'art qui porte sur quelques travaux qui ont été proposés pour les langages d'interrogation de base de données multidimensionnelle (BDM), nous discutons et expliquons sur la manipulation de données multidimensionnelles, nous allons présenter les principaux composants décisionnels disponibles en open source et quelques des langages textuels et graphiques

Les outils d'analyse permettant, à partir de bases multidimensionnelles (MOLAP), relationnelles (ROLAP) ou hybrides (HOLAP), d'agencer les données selon plusieurs dimensions (cubes de données). L'apport de la navigation dans des cubes de données est déterminant dans la révélation de corrélations cachées.

Les outils d'aide à la décision dont la particularité est la facilité d'utilisation. L'ergonomie de la présentation des informations est optimisée de manière à ce que l'utilisateur (généralement, un dirigeant) interprète le plus rapidement possible ce qu'il voit.

2. Les langages pour bases de données multidimensionnelles

La représentation multidimensionnelle des données induit de nouveaux langages constitués de nouvelles opérations liées à l'aspect "analyse". Ces opérations peuvent être classées selon le type de langage.

2.1 Types de langages

Les langages dédiés aux bases de données multidimensionnelles sont classés en trois types [Ravat, Teste, Zurfluh, 02] à savoir :

2.1.1 Le langage de Définition des Données (LDD) : ce langage définit les opérateurs nécessaires à la définition d'un schéma multidimensionnel (définition des faits, dimensions et hiérarchies) ainsi qu'à la modification de ses éléments ou encore leur suppression.

2.1.2 Le langage de Manipulation des Données (LMD): ce langage définit les opérations nécessaires à la mise à jour ainsi qu'à la consultation des données.

2.1.3 Le langage de Contrôle des Données (LCD) : Ce langage définit les opérations nécessaires à l'affectation et à la suppression des droits des utilisateurs sur les éléments du schéma multidimensionnel.

3. Principes de la manipulation OLAP

3.1 Structure de visualisation

La représentation sous forme de tableau est la structure de visualisation qui est le plus souvent retenue [Agrawal, et al., 1997], [Gyssens, et al., 1997], [Lehner 1998]. Il s'agit d'une vision synthétique et précise des données que les décideurs appréhendent facilement. Elle dérive directement de la métaphore du cube de données puisqu'elle peut être considérée comme la tranche du cube de données comme l'illustre la figure suivante :

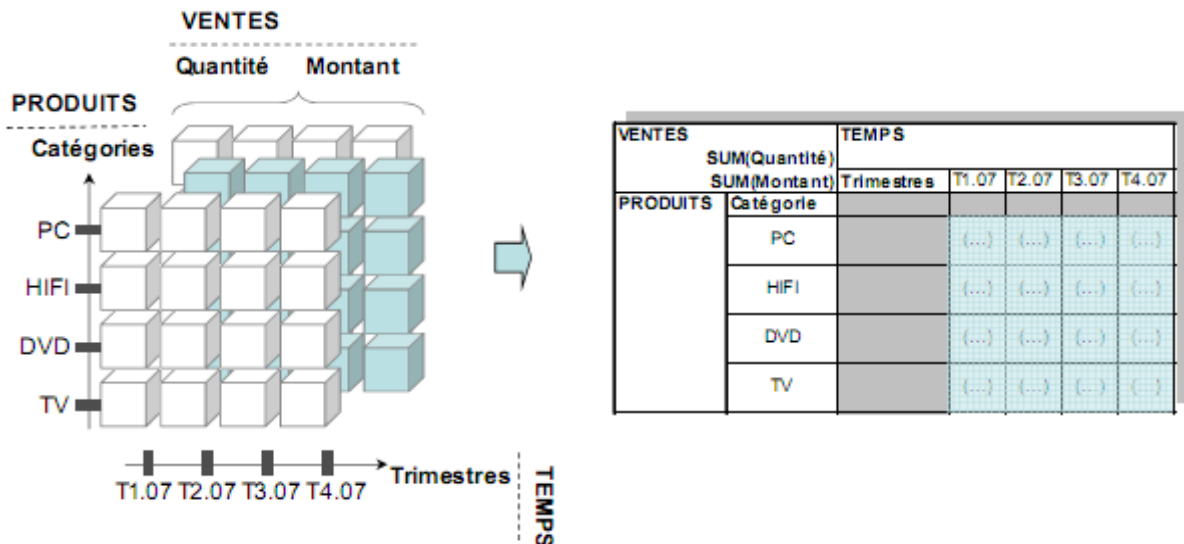


Figure 2.1 : Notion de tranche du cube de données.

Donc la construction d'une table multidimensionnelle se fait par la fonction DISPLAY (NCS, F, [{f1 (m1), f2 (m2),...}], DL, [HL], DC, [HC]) = TR, avec les arguments suivants :

- NCS : nom de la constellation,
- F : fait analysé (sujet de l'analyse),
- {f1 (m1), f2 (m2),...} : mesures et fonctions d'agrégations de F
- DL : dimension en ligne avec HL comme hiérarchie courante,
- DC : dimension en colonne avec HC comme hiérarchie courante.

Exemple 1: DISPLAY (C, VENTES, {sum(Montant), Sum(Bénéfices)}, TEMPS, MAGASINS)

VENTES		MAGASINS			
(Montant, Bénéfices)		Villes	Toulouse	Lyon	Dallas
TEMPS	Années				
	2002		(20, 2)	(30, 4)	(20, 3)
	2001		(10, 2)	(20, 4)	(30, 5)
	2000		(20, 5)	(20, 3)	(20, 1)
	1999		(30, 6)	(10, 2)	(10, 1)

Figure 2.2: Exemple d'un tableau multidimensionnel

3.2 Opérations de manipulation OLAP

De nombreuses propositions [Rafanelli, 2003] concernent la définition d'opérations de manipulation : cube de données, tranche du cube de données... Il n'existe pas de consensus sur la définition d'un ensemble minimum d'opérateurs assurant l'intégralité des opérations de manipulation OLAP, mais la plupart des propositions offrent un support partiel des différentes catégories d'opérations. Parmi ces opérations, les plus emblématiques sont les opérations de forage et les opérations de rotation qui reposent directement sur la métaphore du cube des données. Au-delà de ces opérations, la littérature scientifique et les nombreux logiciels offrent une grande variété d'opérations.

3.2.1 Opérations de forage

Les opérations de forage font reposer la navigation sur la structure hiérarchique des axes d'analyses, afin de permettre l'analyse d'un indicateur avec plus ou moins de précision. Le forage vers le haut (« roll-up ») consiste à analyser les données en fonction d'un niveau de granularité moins détaillé tandis que le forage vers le bas (« drill-down »), à l'inverse, permet d'analyser les données avec un niveau plus fin.

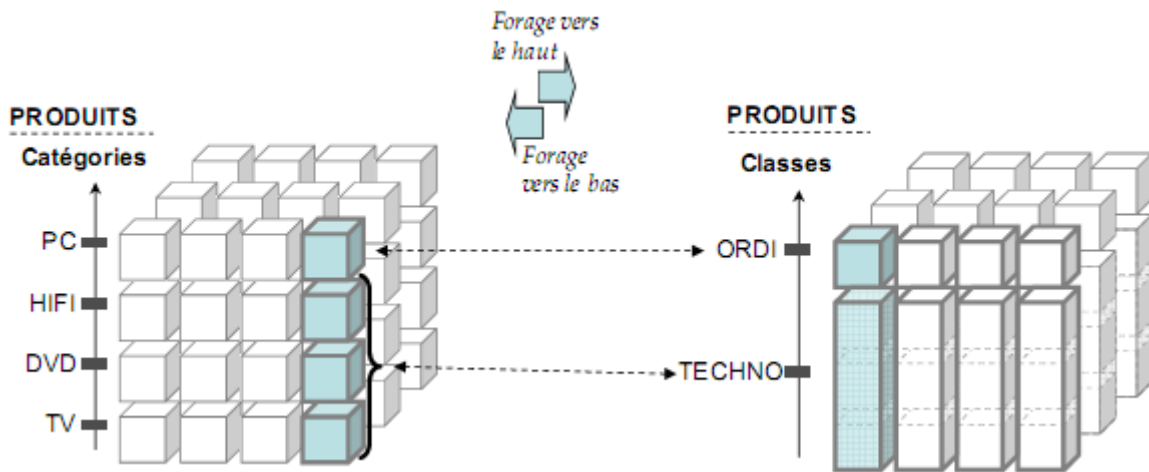
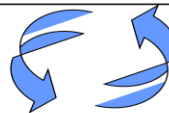


Figure 2.3 : Principe du forage.

Exemple 2:

VENTES		CLIENTS H_Cli			
Sum(montant)		pays	Allemagne	France	Espagne
TEMPS H_Tps	annee				
	2001		200	150	300
	2000		250	240	260
	1999		200	210	220

DrillDown (T1, CLIENTS, ville) = T2



RollUp (T2, CLIENTS, pays) = T1

VENTES		CLIENTS H_Cli					
Sum(montant)		pays	Allemagne		France	Espagne	
TEMPS H_Tps	annee	ville	Berlin	Hambourg	Paris	Toulouse	Madrid
	2001		150	50	100	50	300
	2000		160	90	100	140	260
	1999		100	100	110	100	220

Figure 2.4 : Le forage vers le haut (« roll-up ») et le forage vers le bas (« drill-down »)

3.2.2 Opérations de rotation

Les opérations de rotation réorientent une analyse. L'opération la plus courante consiste à changer l'axe d'analyse en cours d'utilisation (rotation de dimension).

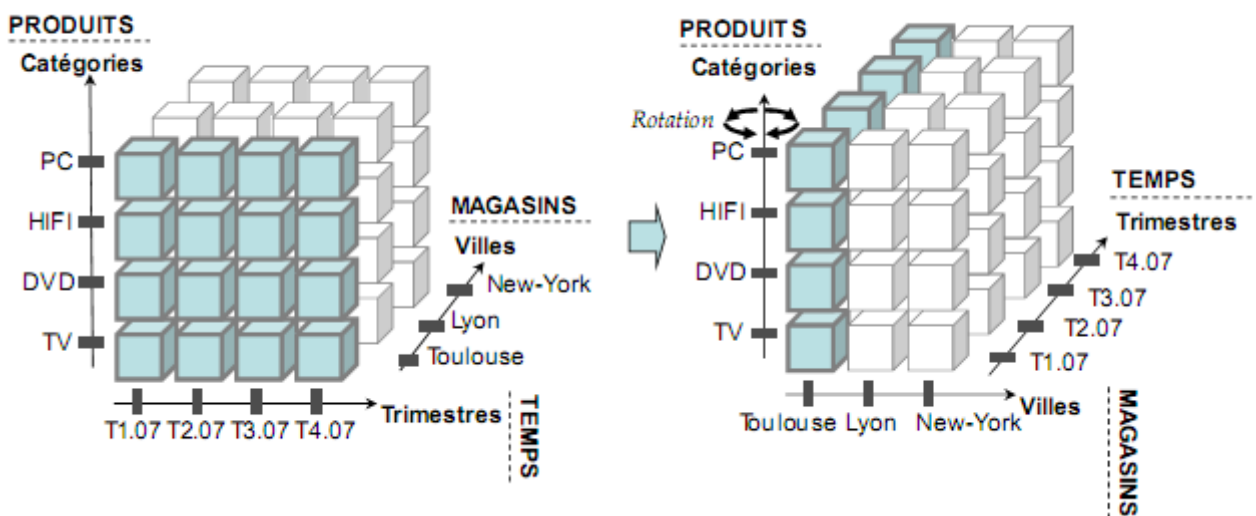


Figure 2.5 : Principe de rotation.

Un cas particulier de la rotation de dimension consiste à changer de perspective d'analyse (rotation de hiérarchie). Dans ce cas, l'axe en cours d'utilisation est maintenu, mais la manière de le graduer est changée. Une autre opération apparentée à cette opération de rotation consiste à changer le sujet de l'analyse (rotation de fait ou « FRotate») dans le contexte d'une constellation. Cette opération nécessite une forte compatibilité entre les dimensions des faits sur lesquels s'applique l'opération.

Exemple 3 :

FRotate (TS , F2, [$f_1(m_1), f_2(m_2), \dots$]) = TR : Rotation du fait courant pour visualiser les mesures du fait F₂

VENTES		PRODUITS H_Pdt				
Sum(montant)		categorie	C1	C2	C3	C4
TEMPS H_Tps	annee					
	2001		140	250	130	240
	2000		240	150	350	240
	1999		170	200	100	220

FRotate (TS, ACHAT,) = TR



ACHAT		PRODUITS H_Pdt				
Sum(PrixA)		categorie	C1	C2	C3	C4
TEMPS H_Tps	annee					
	2001		11	12	10	10
	2000		7	7	8	9
	1999		4	5	9	6

Figure 2.6 : Rotation de fait (« FRotate »)

DRotate (TS, D1, D2 [, Hi]) = TR : Rotation de la dimension D₁ avec la dimension D₂

VENTES		CLIENTS H_Zo				
Sum(montant)		zone	Nord	Sud	Est	Ouest
TEMPS H_Tps	annee					
	2001		100	120	200	180
	2000		230	210	200	140
	1999		180	150	190	160

DRotate (TS, CLIENTS, PRODUITS) = TR



VENTES		PRODUITS H_Pdt				
Sum(montant)		categorie	C1	C2	C3	C4
TEMPS H_Tps	annee					
	2001		140	250	130	240
	2000		240	150	350	240
	1999		170	200	100	220

Figure 2.7 : Rotation de Dimension (« DRotate »)

HRotate (TS, D, H1, H2) = TR : Rotation de la hiérarchie H₁ avec la hiérarchie H₂

VENTES		CLIENTS H_Cli			
Sum(montant)		pays	Allemagne	France	Espagne
TEMPS H_Tps	annee				
	2001		200	150	300
	2000		250	240	260
	1999		200	210	220

HRotate (TS, CLIENTS, H_Cli, H_Zo) = TR



VENTES		CLIENTS H_Zo				
Sum(montant)		zone	Nord	Sud	Est	Ouest
TEMPS H_Tps	annee					
	2001		100	120	200	180
	2000		230	210	200	140
	1999		180	150	190	160

Figure 2.8 : Rotation de l' hiérarchie (« HRotate »)

3.2.3 Autres opérations

Il est possible de classer les nombreuses autres opérations proposées dans le cadre des manipulations OLAP en différentes catégories.

- Les opérations de restriction permettent à un utilisateur de restreindre l'ensemble des données analysées. La spécification d'une tranche de cube (« slice ») consiste à exprimer une restriction sur une des données de l'un des axes d'analyse. La spécification d'un sous-cube (« dice ») consiste à exprimer une restriction sur les données d'un indicateur d'analyse.
- Des opérations de transformation permettent l'ajout d'attributs de dimension en tant qu'indicateur d'analyse (« push ») ou de convertir un indicateur d'analyse en paramètre (« pull »).
- Les opérations d'ordonnement permettent de changer l'ordre des valeurs (positions) des paramètres des dimensions (« switch ») ou de réordonner les paramètres d'une hiérarchie (« nest »). Par généralisation, cette dernière permet d'imbriquer un attribut dans une autre hiérarchie.
- Certains auteurs proposent aussi l'emploi des opérations binaires ensemblistes (union, différence et intersection) qui nécessitent une très forte compatibilité entre les deux structures multidimensionnelles manipulées. Certains travaux ont aussi proposé la notion de jointure inspirée de la jointure relationnelle, mais d'un intérêt limité dans un environnement multidimensionnel.

3.3 Travaux sur la manipulation de données multidimensionnelles

Les premiers travaux sur les manipulations OLAP ont étendu les opérateurs de l'algèbre relationnelle pour le modèle en cube [Gray et al., 1996], [Agrawal et al., 1997], [Li & Wang, 1996], [Gyssens & Lakshmanan, 1997], [Datta & Thomas, 1999], [Rafanelli, 2003]. Pour mieux prendre en compte les structures multidimensionnelles, d'autres travaux ont proposé des opérateurs pour spécifier et manipuler un cube [Cabibbo, et al., 1997] [Abelló et al., 2003] [Pedersen, et al., 2001] [Franconi, et al., 2004]. La majorité des travaux repose sur une structure de visualisation simplifiée dans laquelle le concept de hiérarchie n'est pas exploité. Certaines des propositions [Gyssen, et al., 1997] ne supportent qu'un niveau de paramètre en entête des lignes et des colonnes de la structure de visualisation. Ni la structure de visualisation des données, ni l'ensemble des opérateurs OLAP ne font encore aujourd'hui l'objet d'un consensus [Ravat, Teste, Tournier, Zurfluh, 2007] à l'image des opérateurs de l'algèbre relationnelle. De part l'absence d'un

standard de description des opérations OLAP, les outils de manipulation souffrent d'une grande hétérogénéité dans les langages proposés.

Ces propositions visent à répondre aux besoins d'analyse OLAP des décideurs en définissant des opérateurs interactifs facilitant la navigation au sein des données multidimensionnelles [Abelló, et al., 2003]. Différentes études comparatives ont été réalisées dans [Rafanelli, 2003] [Torlone, 2003], [Abelló, et al., 2006] et [Ravat, Teste, Tournier, Zurfluh, 2008]. Le tableau suivant dresse de manière chronologique un panorama des propositions sur la manipulation OLAP.

		[Gray, et al, 1996]	[Li, et al, 1996]	[Agrawal, et al, 1997]	[Gyssens, et al, 1997]	[Cabibbo, et al, 1997, 1998]	[Lehner, 1998]	[Datta, et al., 1999]	[Pedersen, et al, 2001]	[Abelló, et al., 2002, 2006]	[Franconi, et al, 2004]	[Ravat, Teste, Zurfluh, 2006] [Ravat, Teste, Tournier, Zurfluh, 2008]
Forage	Haut		X	X	X	X	X		X	X		X
	Bas		X	X			X	X		X		X
Rotation	Hiérarchie											X
	Dimension									X		X
	Fait							X	X	X		X
Restriction	Dimension			X	X	X		X			X	X
	Fait				X	X				X		X
Transformation	Paramètre en Mesure			X	X							X
	Mesure en Paramètre			X	X							X
Ordonnancement	Position				X							X
	Imbrication Paramètre		X									X
Calcul	Agrégation	X										X
Binaire	Union			X	X				X	X	X	X
	Intersection			X	X						X	X
	Différence			X	X						X	X
	Jointure		X						X			

Tableau 2.1 : Synthèse des travaux sur les langages de manipulation OLAP. [O.Teste,2010]

4. Les composantes d'une interface décisionnelles

4.1 Le Tableau Croisé Dynamique (TCD)

Un TCD est *croisé* : toute donnée dépend des en-têtes de sa ligne et de sa colonne.

Un TCD est *dynamique* : un TCD est évolutif, facilement modifiable. Il permet d'examiner les données sous des angles différents.

-Les Tableaux Croisés Dynamiques (TCD) sont des outils d'analyse très performants. Ils permettent de comparer et de calculer par des simples clics toute sorte d'éléments. Par un simple cliquer glisser l'utilisateur peut ensuite modifier les composants du tableau. Nous allons essayer de vous décrire l'utilité des tableaux croisés dynamiques par un exemple pratique avec une base de données simple.

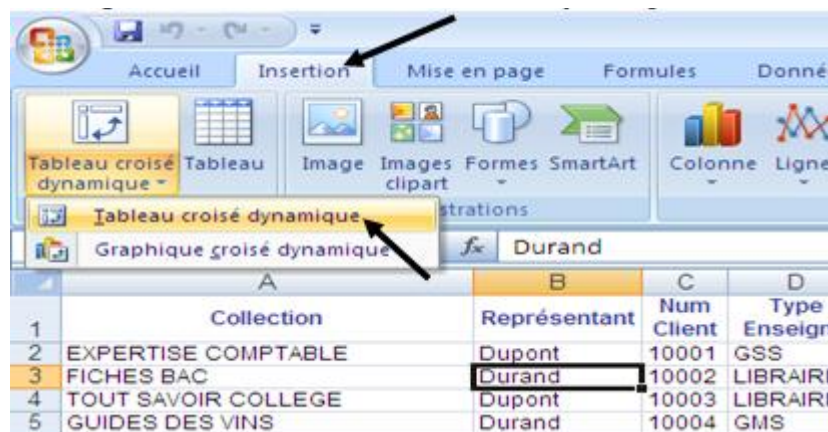
Dans notre exemple, nous utiliserons une base de données comme ci-dessous regroupant les ventes en quantité et en chiffre d'affaires sur deux ans, par produits/clients; et nous verrons comment synthétiser cette base en quelques clics.

	A	B	C	D	E	F	G	H	I	J
	Collection	Représentant	Num Client	Type Enseigne	Raison Sociale	Ville	Qté Vtes An	CA An	Qté Vtes An-1	CA An-1
1	EXPERTISE COMPTABLE	Dupont	10001	GSS	SA FNAC MONTPARNASSE	PARIS	8	87.94	7	79.14
2	FICHES BAC	Durand	10002	LIBRAIRIE	SARL BOITE A LIRE	NICE	18	49.46	16	44.52
3	TOUT SAVOIR COLLEGE	Dupont	10003	LIBRAIRIE	EURL BELLE ENCRE	NOGENT	14	48.09	13	43.28
4	GUIDES DES VINS	Durand	10004	GMS	CARREFOUR TRIFOUILLY	TRIFOUILLY	30	247.32	27	222.59
5	CAHIERS VACANCES MATERNELLE	Dupont	10005	GMS	LECLERC BECON	BECON	56	115.42	50	103.87
6	CONCOURS ADMINISTRATIFS	Durand	10006	GMS	LECLERC LES BUYERES	LES BRUYERES	100	618.30	90	556.47
7	ANGLAIS DE A A Z	Dupont	10007	GSS	SA FNAC FORUM	VINCENNES	139	668.45	125	601.61
8	EXPERTISE COMPTABLE	Durand	10008	LIBRAIRIE	LIVRE D'ART SARL	PARIS	31	340.75	37	408.90
9	FICHES BAC	Dupont	10009	LIBRAIRIE	EURL LIVRE EN FETE	PARIS	4	10.99	5	13.19
10	TOUT SAVOIR COLLEGE	Durand	10010	GMS	CARREFOUR ZI	NICE	42	144.27	50	173.12
11	GUIDES DES VINS	Dupont	10011	GSS	VIRGIN 456	NOGENT	1	8.24	1	9.89
12	CAHIERS VACANCES MATERNELLE	Durand	10012	GSS	FORUM BELLE	TRIFOUILLY	111	228.77	133	274.53
13	CONCOURS ADMINISTRATIFS	Dupont	10013	LIBRAIRIE	SALA GR	BECON	9	55.65	11	65.78
14	EXPERTISE COMPTABLE	Dupont	10001	GSS	SA FNAC MONTPARNASSE	PARIS	12	131.90	14	158.28
15	FICHES BAC	Durand	10002	LIBRAIRIE	SARL BOITE A LIRE	NICE	176	483.65	211	580.38
16	TOUT SAVOIR COLLEGE	Dupont	10003	LIBRAIRIE	EURL BELLE ENCRE					
17	GUIDES DES VINS	Durand	10004	GMS	CARREFOUR TRIFOUILLY					
18	CAHIERS VACANCES MATERNELLE	Dupont	10005	GMS	LECLERC BECON					
19	CONCOURS ADMINISTRATIFS	Durand	10006	GMS	LECLERC LES BUYERES					
20	ANGLAIS DE A A Z	Dupont	10007	GSS	SA FNAC FORUM					
21	EXPERTISE COMPTABLE	Durand	10008	LIBRAIRIE	LIVRE D'ART SARL					
22	FICHES BAC	Dupont	10009	LIBRAIRIE	EURL LIVRE EN FETE					
23	TOUT SAVOIR COLLEGE	Durand	10010	GMS	CARREFOUR					
24	GUIDES DES VINS	Dupont	10011	GSS	VIRGIN 456					
25	CAHIERS VACANCES MATERNELLE	Durand	10012	GSS	FORUM BELLE					
26										

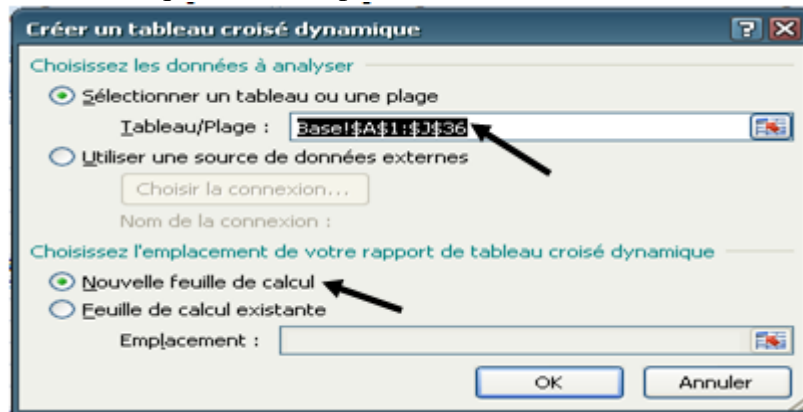
	CA An-1	CA An
Dupont	2 382.1	2 332.7
GMS	413.7	390.9
GSS	1 129.0	1 175.9
LIBRAIRIE	839.4	765.9
Durand	3 985.5	3 754.3
GMS	2 090.6	2 147.0
GSS	655.6	491.6
LIBRAIRIE	1 239.3	1 115.7
Total général	6 367.6	6 087.0

Etape 1: Sélection de la plage de données

Sélectionnez une cellule quelconque de la base de données, puis cliquez sur l'onglet "Insertion"; dans le groupe "Tableaux" à gauche choisissez "Tableau croisé dynamique":

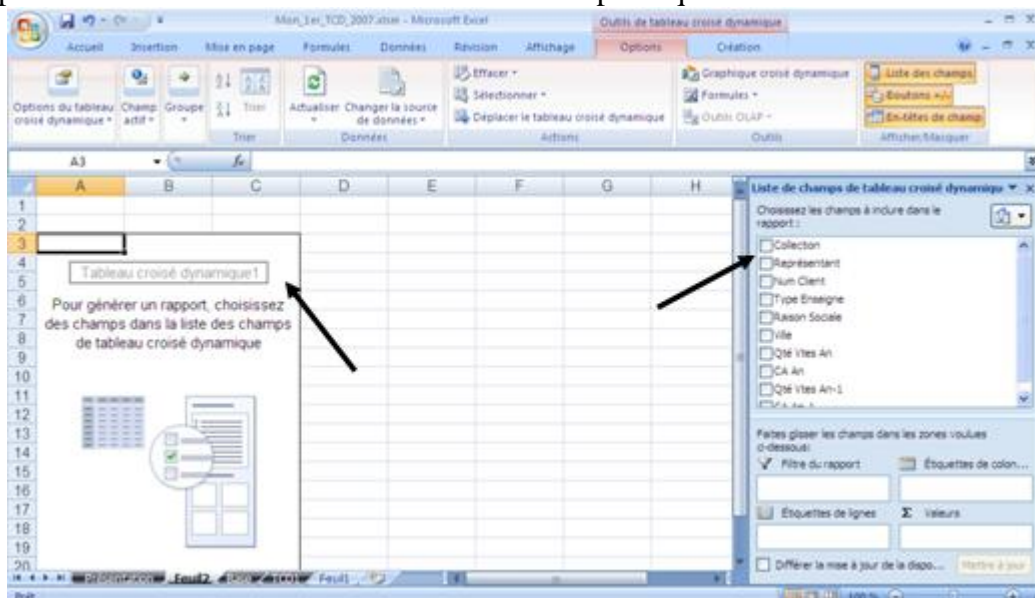


La fenêtre suivante apparaît, Excel a déterminé tout seul la plage de données en sélectionnant l'ensemble de notre base qui s'étend de la cellule A1 à la cellule J36, et propose par défaut de placer le TCD dans une nouvelle feuille qui sera créée pour l'occasion:

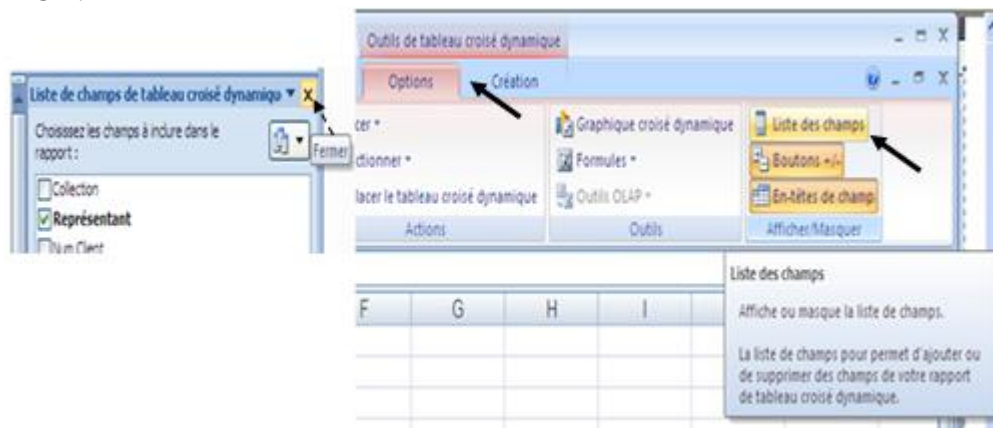


Etape 2: construction du TCD

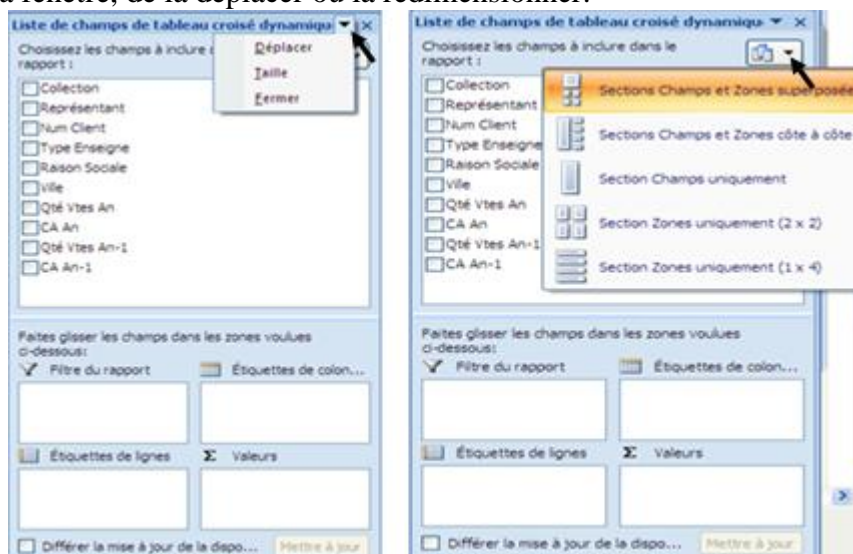
Vous êtes maintenant sur le nouvel onglet qu'Excel vient de créer, on remarque à gauche le TCD vide, tel qu'il apparaît avant que les champs soient définis; à droite la liste des champs, en haut deux onglets Options et Création offrant de nombreux outils spécifiques aux TCD:



Si vous avez fermé la fenêtre "Liste des champs" pour avoir plus de place à l'écran, elle ne réapparaît pas en cliquant sur le TCD, il faut cliquer sur le bouton Liste de champs de l'onglet Options du menu Outils de TCD:



C'est à partir de la fenêtre Liste des champs que nous allons construire le TCD, en sélectionnant et déplaçant les champs qui nous intéressent. La liste des champs contient deux menus permettant de personnaliser la fenêtre, de la déplacer ou la redimensionner:



Nous conservons volontairement ici la présentation standard d'Excel.

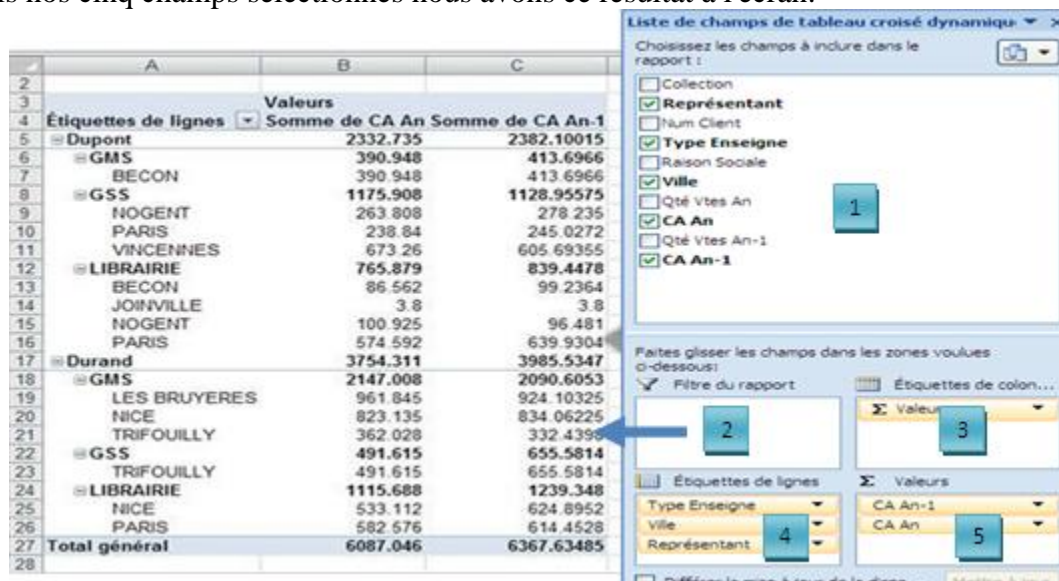
Commençons par un problème simple, nous voulons comparer le chiffre d'affaires An et An-1 par Représentant et Type d'enseigne, et souhaitons pouvoir afficher les chiffres pour l'ensemble de l'activité ou pour certaines villes. Nous allons construire un tableau comme celui-ci:

	A	B	C
1	Ville	(Tous)	
2			
3		CA An-1	CA An
4	Dupont	2 382.1	2 332.7
5	GMS	413.7	390.9
6	GSS	1 129.0	1 175.9
7	LIBRAIRIE	839.4	765.9
8	Durand	3 985.5	3 754.3
9	GMS	2 090.6	2 147.0
10	GSS	655.6	491.6
11	LIBRAIRIE	1 239.3	1 115.7
12	Total général	6 367.6	6 087.0
13			

Une première méthode consiste à cocher dans la liste des champs, ceux qui nous intéressent (**zone 1** ci-dessous)

Au fur et à mesure de nos clics, Excel renseigne les zones inférieures de la fenêtre "Liste de champs" et construit le TCD que nous voyons évoluer à chaque étape sur la gauche.

Une fois nos cinq champs sélectionnés nous avons ce résultat à l'écran:



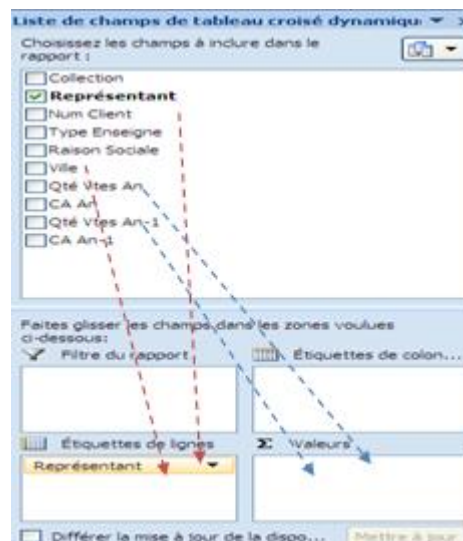
Excel a su placer tout seul certaines données au bon endroit, on retrouve le type d'enseigne et le représentant en étiquette de ligne (**zone 4**ci-dessus et en en-tête de ligne dans le TCD), nous n'avons pas d'étiquette de colonne et retrouvons bien les noms des champs de Chiffre d'Affaires dont nous voulons faire la somme.

En revanche, Excel ne pouvait pas deviner que nous souhaitions la ville non pas en étiquette de ligne, mais en filtre au dessus du tableau.

Nous pouvons alors cliquer sur le champ Ville, et tout en maintenant le bouton gauche de la souris enfoncé, le faire glisser des Etiquettes de ligne vers l'emplacement Filtre du rapport, le TCD se modifie immédiatement en plaçant le champ ville en première ligne du TCD:

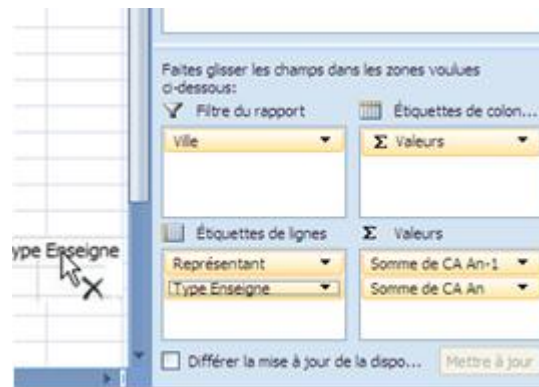
Étiquettes de ligne	Somme de CA An-1	Somme de CA An
Dupont	2382.10015	2332.735
GMS	413.6966	390.948
GSS	1128.95575	1175.908
LIBRAIRIE	839.4478	765.879
Durand	3985.5347	3754.311
GMS	2090.6053	2147.008
GSS	655.5814	491.615
LIBRAIRIE	1239.348	1115.688
Total général	6367.63485	6087.046

Cette méthode "automatique", en ne faisant que cliquer sur les champs qui nous intéressent, ne convient généralement que pour les cas très simples, il est globalement préférable d'utiliser la seconde méthode, qui consiste à sélectionner les champs et à les faire glisser en maintenant le bouton gauche de la souris enfoncé dans les zones appropriées:

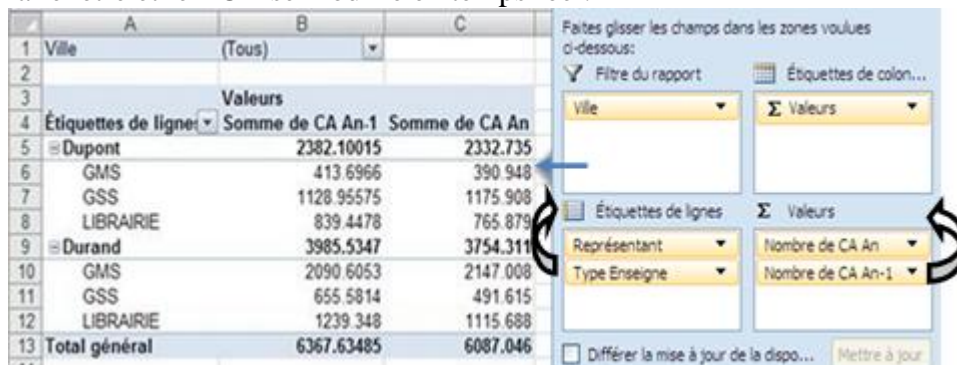


Les étiquettes de lignes correspondent aux en-têtes de ligne à gauche du TCD, les étiquettes de colonnes aux en-têtes de colonnes du TCD (nous n'en avons pas ici), les valeurs sont les données numériques que nous voulons synthétiser (ici les chiffres d'affaires dont nous voulons la somme), les filtres sont les anciens "Champs de page" des versions antérieures d'Excel, et permettent d'afficher ou filtrer certaines données de la base.

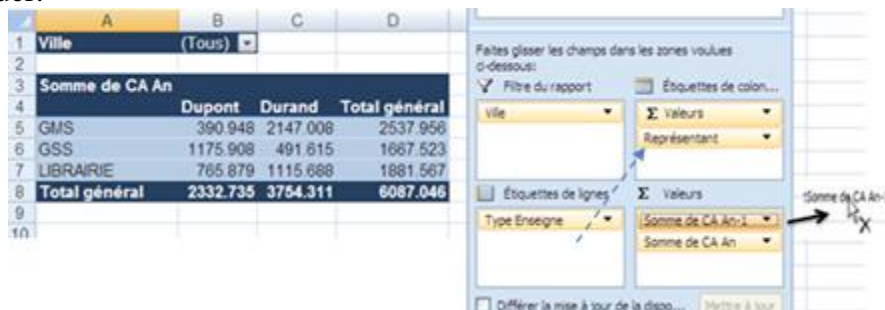
Pour supprimer un champ, il suffit de le sélectionner et de la faire glisser à l'extérieur de la fenêtre:



Pour modifier l'ordre des champs, il suffit de les déplacer en les faisant glisser dans la fenêtre. Par exemple nous souhaitons voir le champ An-1 avant le champ An, et préférons un sous-total par type d'enseigne puis représentant au lieu de représentant / type d'enseigne; nous modifions l'ordre des champs dans la fenêtre et le TCD se modifie en temps réel:



Si nous voulons voir le représentant en étiquette de colonne et ne voir que le chiffre d'affaire de l'année N, il suffit de faire glisser le champ représentant de la zone "Etiquette de ligne" vers la zone "Etiquette de colonne", puis de faire glisser le champ CA An-1 hors de la fenêtre, nous voyons immédiatement le TCD évoluer:



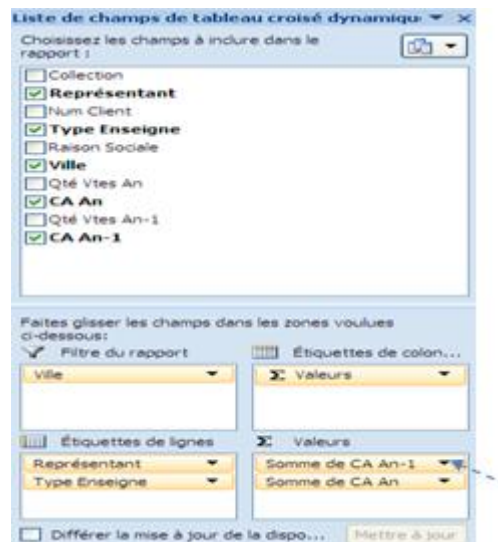
Etape 3: Première mise en forme: Notre tableau est construit, nous avons obtenu la synthèse ci-dessous, mais ce tableau manque un peu de mise en forme:

Ville	(Tous)		
		Somme de CA An-1	Somme de CA An
Dupont		2382.10015	2332.735
GMS		413.6966	390.948
GSS		1128.95575	1175.908
LIBRAIRIE		839.4478	765.879
Durand		3985.5347	3754.311
GMS		2090.6053	2147.008
GSS		655.5814	491.615
LIBRAIRIE		1239.348	1115.688
Total général		6367.63485	6087.046

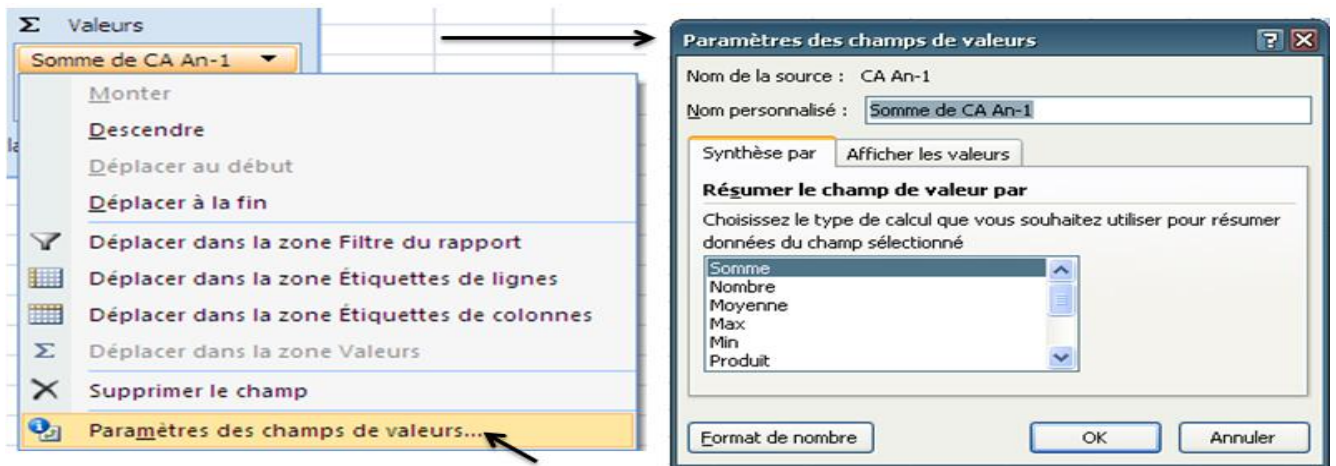
Dans la fenêtre Liste de champs, chaque champ comporte une petite flèche noire à côté de son nom, cette flèche permet d'ouvrir la fenêtre de gestion des paramètres.

Mise en forme des champs de valeur:

Commençons par les nombres en cliquant sur la petite flèche à côté du nom "Somme CA An-1":



La fenêtre suivante de gauche s'affiche, cliquez sur la dernière ligne "Paramètres des champs de valeur", la fenêtre de paramètres s'ouvre:



Ces calculs prédéfinis dans Excel sont très puissants et méritent un chapitre spécial à eux seuls. Votre premier tableau croisé dynamique est maintenant terminé:

	CA An-1	CA An
Dupont	2 382.1	2 332.7
GMS	413.7	390.9
GSS	1 129.0	1 175.9
LIBRAIRIE	839.4	765.9
Durand	3 985.5	3 754.3
GMS	2 090.6	2 147.0
GSS	855.8	491.8
LIBRAIRIE	1 239.3	1 115.7
Total général	6 367.6	6 087.0

Donc Quelle est l'utilité d'un tableau croisé dynamique ?

- Compiler, analyser, regrouper les informations brutes recueillies.
- L'avantage du tableau croisé dynamique est que vous pouvez mettre les données à n'importe quel endroit. De plus, en modifiant le tableau initial, vous pouvez actualiser les données dans votre tableau croisé dynamique d'un simple clic.

4.2 Les graphiques statistiques

4.2.1 Définitions

- Le **graphique statistique** est un Outil de communication visuelle, permettant la représentation de données statistiques sous forme de surfaces ou de traits.

-Un **graphe** permet une représentation visuelle des données complexes d'une feuille de calcul ou d'une base de données. Cette représentation peut être incorporée dans la feuille de données ou dans une feuille graphique.

-**Les données statistiques** sont des données numériques relatives à l'observation d'un fait, d'un phénomène.

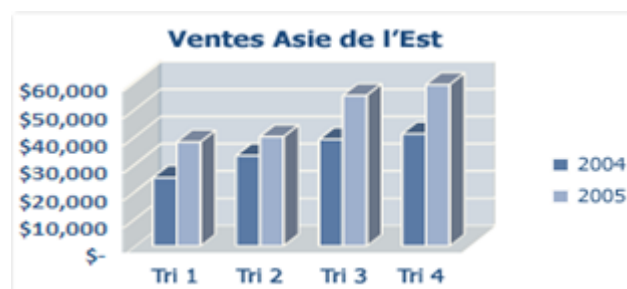
- Chaque type de graphe peut faciliter l'analyse globale de données et surtout leur interprétation. L'utilisation d'un type particulier de graphe dépend des données à représenter et des événements à mettre en évidence.

4.2.2 Types de graphiques : On prend quelques types de graphe que vous pouvez utiliser :

4.2.2.1 Histogrammes

Les données qui sont agencées dans des colonnes ou des lignes sur une feuille de calcul peuvent être représentées dans un histogramme. Les histogrammes permettent d'illustrer les variations des données sur une période donnée ou de comparer des éléments.

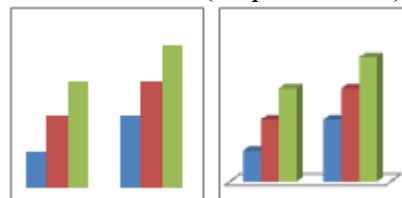
Dans les histogrammes, les catégories sont en général indiquées sur l'axe horizontal et les valeurs sur l'axe vertical.



Les sous-types d'histogrammes sont les suivants :

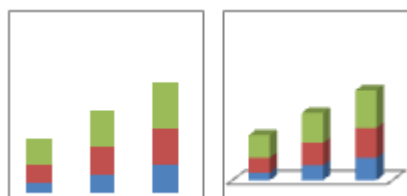
- Histogramme et histogramme groupé en 3D

Les histogrammes groupés comparent les valeurs de différentes catégories et affichent les valeurs dans des rectangles verticaux 2D. Les histogrammes groupés en 3D affichent les données en utilisant uniquement une perspective 3D. Un troisième axe (de profondeur) n'est pas utilisé.



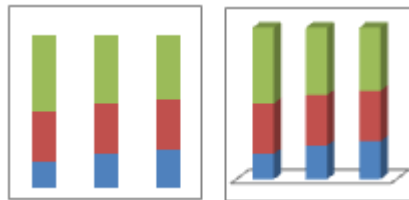
- Histogramme empilé et histogramme 3D empilé

Les graphiques en histogrammes empilés montrent les relations entre des éléments individuels et un tout, en comparant la contribution de chaque élément à un total, catégorie par catégorie. Un graphique en histogrammes empilés affiche les valeurs dans des rectangles verticaux en 2D. Un graphique en histogrammes 3D empilés affiche les données dans une perspective 3D uniquement. Un troisième axe (de profondeur) n'est pas utilisé.



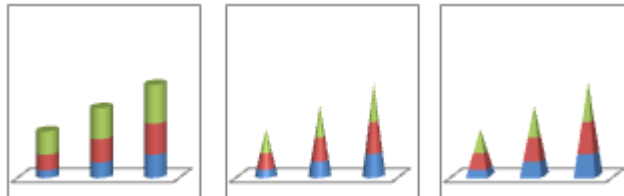
- Histogramme empilé 100 % et histogramme empilé 100 % en 3D

Les graphiques en histogrammes empilés 100 % et en histogrammes empilés 100 % en 3D comparent la proportion dans laquelle chaque valeur contribue à un total dans chaque catégorie. Un graphique en histogrammes empilés 100 % affiche les valeurs sous forme de rectangles verticaux empilés 100 % en 2D. Un graphique en histogrammes empilés 100 % 3D affiche les données dans une perspective 3D uniquement. Un troisième axe (de profondeur) n'est pas utilisé.



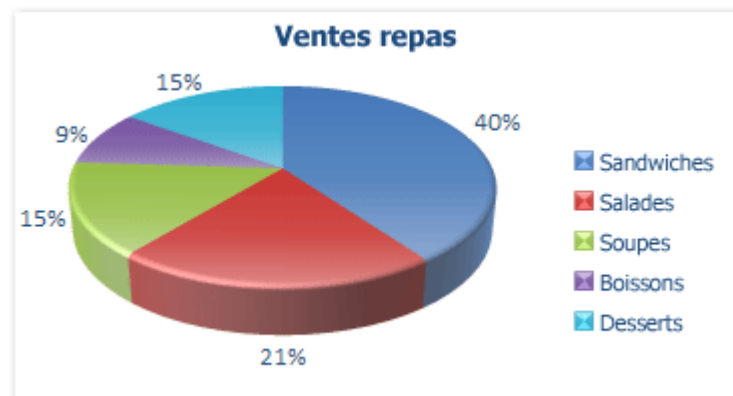
- Cylindre, cône et pyramide

Les graphiques en cylindres, cônes et pyramides existent également dans les mêmes types groupés, empilés, empilés 100 % et 3D que les graphiques en histogrammes rectangulaires. Ils affichent et comparent les données de la même manière. La seule différence réside dans le fait que ces types de graphiques utilisent des cylindres, des cônes et des pyramides au lieu de rectangles.



4.2.2.2 Graphiques en secteurs

Les données qui ne sont agencées qu'en une seule colonne ou une seule ligne sur une feuille de calcul peuvent être représentées sur un graphique en secteurs. Les graphiques en secteurs représentent la taille des éléments d'une série de données par rapport à leur somme. Les points de données dans un graphique en secteurs sont représentés sous forme de pourcentage du graphique entier.

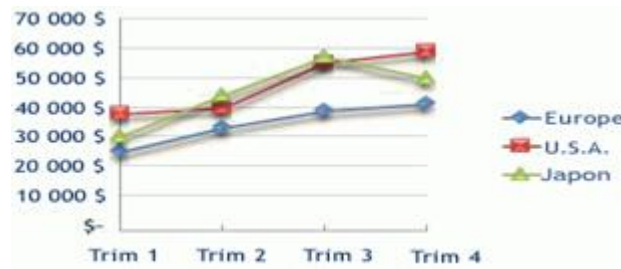


Il est conseillé d'utiliser un graphique en secteurs quand :

- une seule série de données doit être représentée ;
- aucune des valeurs à représenter n'est négative ;
- presque aucune des valeurs à représenter n'est nulle ;
- le nombre de catégories ne dépasse pas sept ;
- les catégories représentent des parties du graphique entier.

4.2.2.3 Graphiques en courbes

Les données qui sont agencées dans des colonnes ou des lignes sur une feuille de calcul peuvent être représentées dans un graphique en courbes. Les graphiques en courbes peuvent afficher des données continues sur une période donnée, ils sont définis par rapport à une échelle courante et conviennent parfaitement pour représenter les tendances dans les données à des intervalles réguliers. Dans un graphique en courbes, les données de catégories sont réparties régulièrement sur l'axe horizontal et les données de valeurs sur l'axe vertical.



4.2.2.4 Graphiques en barres

Les données qui sont agencées dans des colonnes ou des lignes sur une feuille de calcul peuvent être représentées dans un graphique en barres. Ce type de graphique permet de représenter les comparaisons entre différents éléments.



Il est conseillé d'utiliser un graphique en barres lorsque :

- les étiquettes des axes sont longues ;
- les valeurs affichées sont des durées.

4.2.2.5 objectifs d'un graphique

- Permettre une lecture rapide des données (avec une perte de précision par rapport au tableau statistique)
- Mettre en évidence les tendances
- Faciliter la comparaison grâce à la disposition visuelle

5. Langage textuels

5.1 SQL

SQL ("Structured Query Language" ou "Langage de requêtes structuré") est un langage informatique destiné à interroger ou piloter une base de données. Il est utilisé par toutes les bases de données (Oracle, Access...). C'est un langage est proche d'un langage naturel.

5.1.1 Historique

Le langage SQL est né à la suite des travaux du mathématicien *codd*. Historiquement, les années suivantes sont importantes :

1970 : IBM (International Business Machine) développe SQL

1986 : Normalisation SQL/86

1989 : Normalisation SQL/89

1992 : Normalisation ANSI SQL/92

1996: Microsoft adopte TSQL (Transact Structured Query Language)

5.1.2 Les différents types d'instructions : le langage SQL possède trois types d'instructions :

5.1.2.1 Une partie sur la définition des données

-le Langage de Définition des Données (LDD ou DDL en anglais) utilisé pour la création, la modification et la suppression des objets de base de données (table, vues, etc.). Il utilise principalement les instructions CREATE, ALTER et DROP.

Nom	Commentaires
CREATE TABLE	<p>Fonction : Crée une relation.</p> <p>Syntaxe : CREATE TABLE <nom_de_relation> (<nom_d'attribut> <type_de_données> [NOT NULL] [, ...] [, PRIMARY KEY (<nom_d'attribut> [, ...])] [, FOREIGN KEY (<nom_d'attribut>) REFERENCES <nom_de_relation> [, ...]]) [CHECK (<condition>)]</p>
ALTER TABLE	<p>Fonction : Modifie le schéma de relation.</p> <p>Syntaxe : ALTER TABLE <nom_de_relation> ADD (<nom_d'attribut> <type_de_données>)</p>
DROP TABLE	<p>Fonction : Permet de supprimer une relation d'un schéma de base de données.</p> <p>Syntaxe: DROP TABLE <nom_de_relation> [CASCADE CONSTRAINTS]</p>

Tableau 2.2 : les instructions CREATE, ALTER et DROP

5.1.2.2 Une partie sur les requêtes

-le Langage de Manipulation des Données (LMD ou DML en anglais) qui permet de sélectionner, d'ajouter, de modifier et de supprimer des données dans les objets de base de données (tables, vues, etc.). Il utilise principalement les instructions SELECT, INSERT et DELETE.

Nom	Commentaires
SELECT	<p>Fonction : SELECT exprime le résultat attendu sous la forme d'une liste d'attributs auxquels il est possible d'appliquer différents opérateurs et fonctions.</p> <p>Syntaxe : SELECT nom_colonne,..., nom_colonne * FROM nom_table,..., nom_table; [WHERE <condition>]</p>
INSERT	<p>Fonction : La commande INSERT permet d'ajouter des enregistrements à une relation dont le schéma a été préalablement définie soit un enregistrement à la fois soit en utilisant une expression de sélection. La syntaxe de cette commande est :</p> <p>Syntaxe : INSERT INTO <nom_de_relation > [(<liste_d'attributs>)] VALUES (<liste_de_valeurs >) ou INSERT INTO <nom_de_relation> [(<liste_d'attributs>)] <expression_de_sélection></p>
DELETE	<p>Fonction : La commande DELETE permet de supprimer des enregistrements d'une relation. Cette opération n'affecte pas le schéma de la relation et ceci même dans le cas où la commande a pour effet de supprimer tous les enregistrements de la relation.</p> <p>Syntaxe : DELETE FROM < nom_de_relation > [WHERE < expression_de_selection >]</p>

Tableau 2.3 : les instructions SELECT, INSERT et DELETE

5.1.2.3 Une partie sur le contrôle des données

-le Langage de Contrôle des Données (LCD ou DCL en anglais) utilisé pour la gestion des protections d'accès aux données. Il utilise principalement les instructions GRANT et REVOKE.

Nom	Commentaires
GRANT	<p>Fonction : La commande GRANT permet d'attribuer des privilèges. Un privilège est une autorisation d'exécuter un acte. (INSERT, DELETE et SELECT).</p> <p>Syntaxe : GRANT <privilège > [, ...] ON [TABLE] <nom_de_relation> TO {<nom_d'utilisateur > [, ...]}</p>
REVOKE	<p>Fonction : La commande REVOKE permet de supprimer des privilèges.</p> <p>Syntaxe : REVOKE <privilège > [, ...] ON [TABLE] <nom_de_relation> FROM {<nom_d'utilisateur > [, ...]}</p>

Tableau 2.4 : les instructions GRANT et REVOKE

5.2 MDX

5.2.1 Introduction à MDX

MDX (Multi Dimensional eXpression) est un langage de requêtes OLAP pour les bases de données multidimensionnelles, de la même manière que SQL est utilisé pour les requêtes sur les bases de données relationnelles. Dans son approche, MDX est proche du SQL sur son aspect **select** et **where** même si la similarité ne va pas plus loin. Le but des expressions multidimensionnelles MDX est de rendre aisé et intuitif l'accès aux données de différentes dimensions.

5.2.2 MDX versus SQL

MDX est fait pour naviguer dans les bases multidimensionnelles et pour définir des requêtes sur tous leurs objets (dimensions, hiérarchies, niveaux, membres et cellules) afin d'obtenir (simplement) une représentation sous forme de tableaux croisés

- MDX ressemble à SQL par ses mots clé SELECT, FROM, WHERE, mais :
 - SQL construit des vues relationnelles
 - MDX construits des vues multidimensionnelles des données
- Analogies entre termes multidimensionnels (MDX) et relationnels (SQL) :

Multidimensionnel (MDX)	Relationnel(SQL)
cube	table
Niveau (Level)	Colonne (chaîne de caractère ou valeur numérique)
Dimension	Plusieurs colonnes liées ou une table de dimension
Mesure (Measure)	Colonne (discrète ou numérique)
Membre de dimension (Dimension member)	Valeur dans une colonne et une ligne particulière de la table

Tableau 2.5 : MDX versus SQL

- Structure générale d'une requête **SQL** :
SELECT column1, column2, ... columnn **FROM** table
- Structure générale d'une requête **MDX** :
SELECT axis1 **ON COLUMNS**, axis2 **ON ROWS** **FROM** cube
- **FROM** spécifie la **source** de données :
 - en **SQL** : une ou plusieurs tables
 - en **MDX** : un cube

- **SELECT** indique les **résultats** que l'on souhaite récupérer par la requête :
 - **en SQL** :
 - une vue des données en 2 dimensions (lignes (rows) et colonnes (columns))
 - les lignes ont la même structure définie par les colonnes
 - **en MDX** :
 - nb quelconque de dimensions pour former les résultats de la requête.
 - terme d'axe pour éviter confusion avec les dimensions du cube.
 - pas de signification particulière pour les rows et les columns, mais il faut définir chaque axe : axe1 définit l'axe horizontal et axe2 définit l'axe vertical

5.2.3 Syntaxe de base de MDX :

5.2.3.1 Structure générale d'une requête MDX :

Un prototype de requête MDX est donné par la syntaxe suivante :

```
SELECT [<axis_specification>
      [, <spécification_des_axes>...]]
FROM [<spécification_d_un_cube>]
[WHERE [<spécification_de_filtres>]]
```

5.2.3.2 Spécification de Membres, Tuples et Sets dans MDX

- Membres

- Un **membre** est une **instance** d'un **niveau** d'une **dimension**, spécifié entre **crochets** [...]
Les membres [1997] et [Time]. [1997] renvoient le même résultat car l'année 1997 est l'année par défaut de la dimension "Time".

```
SELECT [Time]. [1997] ON COLUMNS
FROM [Sales]
```

-Tuples

-Un **tuple** est une suite de plusieurs **membres** entre **parenthèses** séparés par une **virgule** :

```
SELECT ([Time]. [1997], [Product]. [Food]) ON COLUMNS
FROM [Sales]
```

-Un **tuple** donne la liste des membres qui identifient une ou plusieurs **cellules** dans un cube.

- Une **cellule** est l'intersection des dimensions d'un cube de données.

-Dans un tuple, les **mesures** sont traitées comme une dimension particulière, nommée [Measures] :

```
SELECT ([Measures]. [Unit Sales], [Product]. [All Products]. [Food]. [Baked Goods]) ON COLUMNS
FROM [Sales]
```

- Sets

-Un **set** est un **ensemble ordonné** de **tuples**

- Un **set** peut être vu comme une **plage de valeurs**

- Un **set commence** par une accolade "{", dans laquelle sont énumérés les **tuples** séparés par des **virgules**, et se termine par une accolade appariée "}"

Ex1 : un exemple d'un set qui contient deux mesures différentes et le même membre (Baked Goods) :

```
SELECT {
    ([Measures]. [Unit Sales], [Product]. [All Products]. [Food]. [Baked Goods]),
    ([Measures]. [Store Sales], [Product]. [All Products]. [Food]. [Baked Goods]) } ON COLUMNS
FROM [Sales]
```

5.2.3.3 Spécification d'un axe dans MDX

La spécification d'un axe doit être un set suivi du mot clef **on** qui est suivi à son tour d'un nom d'axe. Une façon simple est de définir un axe est de présenter sur l'axe tous les membres de d'une dimension :

```
<dimension name>.MEMBERS
```

Par exemple la requête :

```
SELECT
    Years. MEMBERS ON COLUMNS,
    Régions. Continent. MEMBERS ON ROWS
FROM Sales
```

Donne le résultat :

	1994	1995	1996	1997
N. America	120,000	200,000	400,000	600,000
S. America	-	10,000	30,000	70,000
Europe	55,000	95,000	160,000	310,000
Asia	30,000	80,000	220,000	200,000

5.2.3.4 Fonctions sur les membres et les dimensions

Les fonctions **Members**, **CurrentMember** et **Children** permettent de rechercher les membres d'une dimension, La fonction **Members** s'applique sur une dimension et la fonction **Children** sur un membre. La fonction **CurrentMember** s'applique à la fois sur une dimension ou un niveau d'une dimension:

< Dimension >.CurrentMember ou < niveau >.CurrentMember.

Il existe d'autres fonctions qui s'appliquent sur les membres comme **Parent**, **NextMember**, **PrevMember**, **FirstSibling**, **LastSibling**, **FirstChild**, **LastChild**, etc.

5.2.3.5 Fonctions sur les "sets"

Le tableau ci-dessous donne quelques fonctions qui opèrent sur un set et qui retournent un set.

Fonction	Syntaxe	Description
Head	Head (<< Set >> [, << Numeric Expression >>])	Eléments de tête d'un set
Tail	Tail (<< Set >> [, << Numeric Expression >>])	Derniers éléments d'un set
Subset	Subset (<< Set >>, << Start >> [, << Count >>])	Sous-ensemble d'éléments d'un set
TopCount	TopCount (<< Set >>, << Count >> [, << Numeric Expression >>])	Les premiers éléments ayant la plus grande valeur de la mesure
Order	Order (<< Set >>, {<<String Expression>> <<Numeric Expression >>} [, ASC DESC BASC BDESC])	Tri des éléments d'un set
Filter	Filter (<< Set >>, << conditions >>)	Les éléments d'un set qui satisfont le filtre

Tableau 2.6 : Fonctions sur les "sets"

5.3 OLAP-SQL

Le langage OLAP-SQL est un langage de commandes assertionnelles qu'il saisit textuellement au travers de l'éditeur de commande. L'analyseur lexical et syntaxique valide la commande, puis retourne un message à l'utilisateur. Dans le cas d'une interrogation, le logiciel affiche en plus une table multidimensionnelle graphique.

Le langage assertionnel OLAP-SQL sont définies dans [Ravat, Teste, Zurfluh, 2002]. Ils envisagent de poursuivre ces travaux dans deux directions. Dans un premier temps, ils ont étendu le langage OLAP-SQL afin d'intégrer de nouvelles opérations multidimensionnelles plus complexes (Push, Pull, Fold, UnFold, Split,...). La complexité d'opérations telles que Push ou Pull résident dans la nécessité de transformer des mesures en paramètres et inversement. Au vu des expériences de l'équipe dans le domaine des langages graphiques, la seconde perspective qu'ils envisagent est le développement d'un module d'interface graphique pour la saisie de requêtes OLAP-SQL.

La Figure 2.9 illustre le fonctionnement de l'éditeur de commande :

- Une commande invalide provoquant un message d'erreur indiquant que la dimension est inconnue,
- Une commande valide aboutissant à l'affichage d'une table multidimensionnelle.

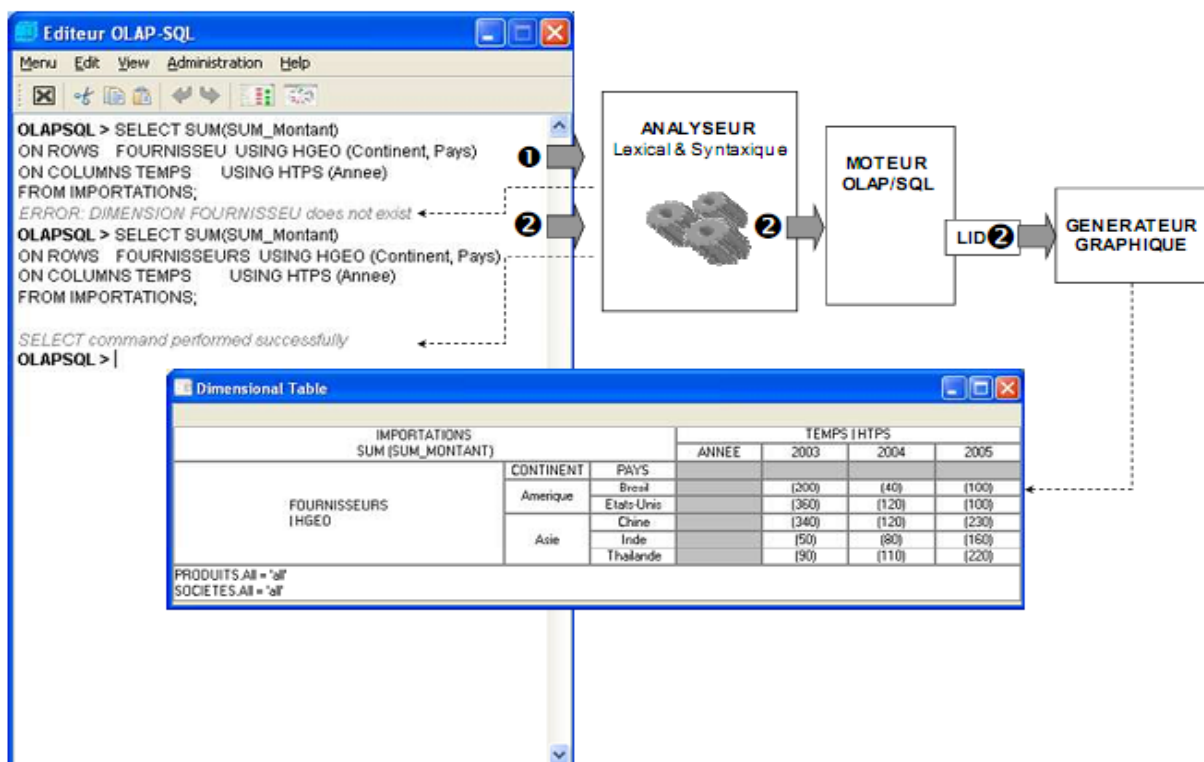


Figure 2.9 : Fonctionnement de l'éditeur OLAP-SQL.

6. Langage graphique

L'outil GRAPHIC-OLAP offre une vision conceptuelle de la BDM au travers de diverses interfaces :

- Arborescente (représentation classique),
- Graphique (représentation conforme aux formalismes de notre modèle en constellation), et
- Hyperbolique (représentation dédiée aux grandes constellations).

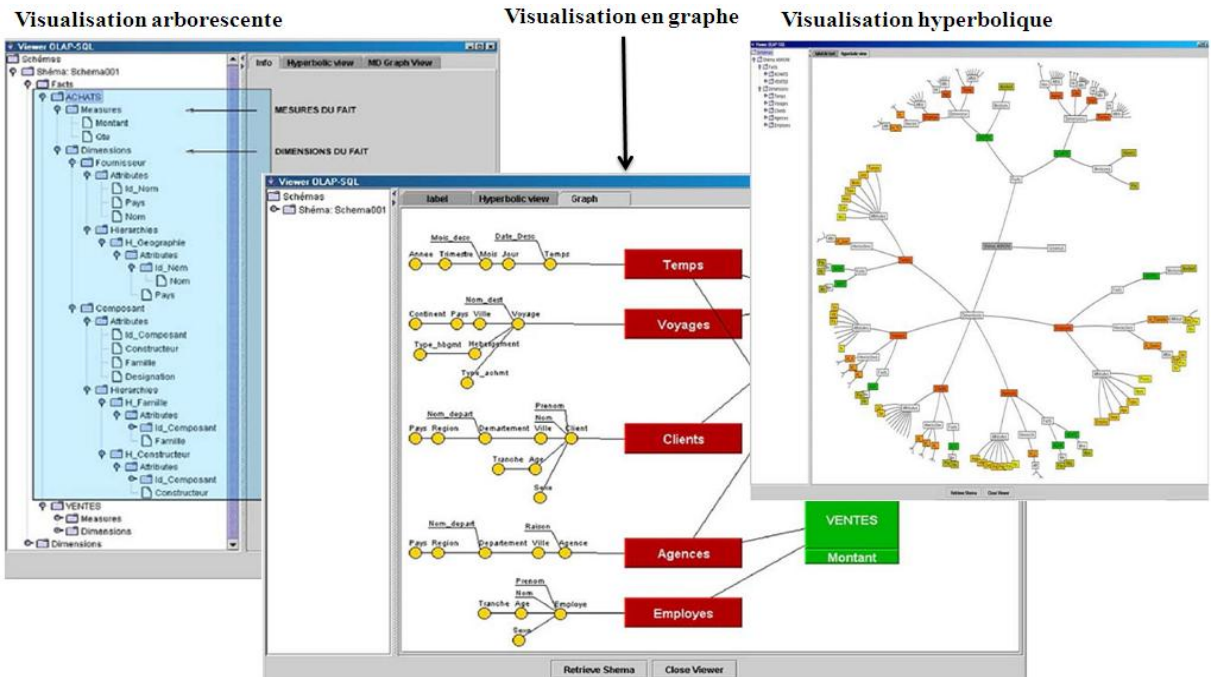


Figure 2.10 : Interfaces de visualisation dans GRAPHIC-OLAP.

L'interface de visualisation en graphe de la constellation sert de support à un langage d'interrogation graphique permettant à l'utilisateur d'élaborer par manipulation directe du graphe.

La figure suivante montre les manipulations graphiques permettant de construire une table multidimensionnelle (TM). Ces manipulations correspondent à l'opération DISPLAY.

La construction d'une TM s'effectue par manipulation directe et incrémentale du graphe. L'utilisateur sélectionne trois nœuds devant être un fait et deux dimensions, le système assure la cohérence des sélections en rendant inaccessibles les nœuds invalides au fur et à mesure des sélections opérées par l'utilisateur.

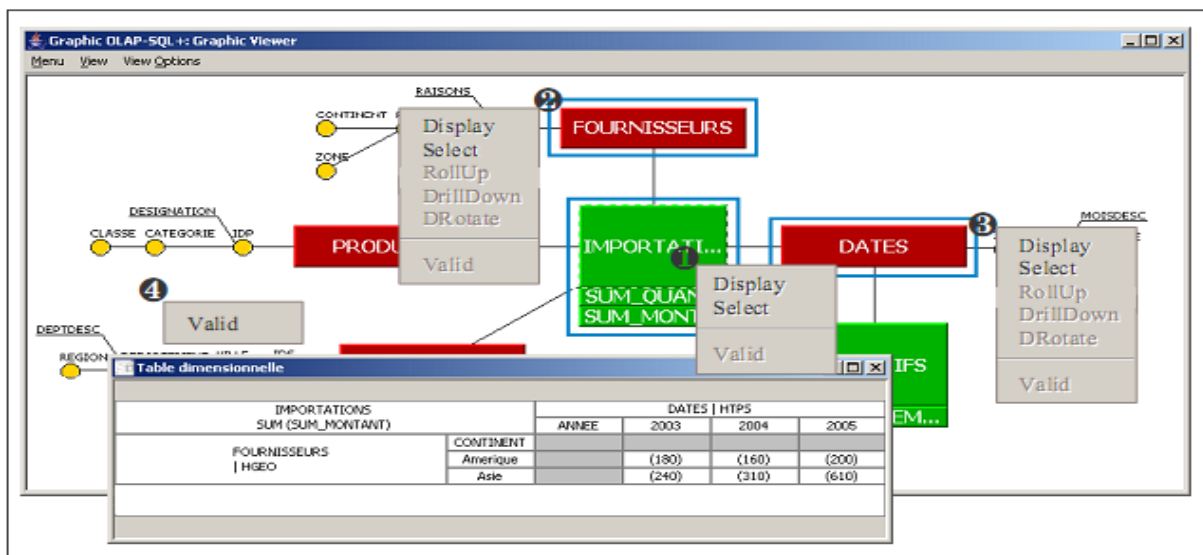


Figure 2.11 : Construction graphique d'une table multidimensionnelle.

6.1 Manipulations OLAP Graphiques

Le système offre à un décideur deux manières d'effectuer des analyses décisionnelles. Un décideur peut agir graphiquement sur la constellation, mais il peut également appliquer certaines opérations directement sur les TM.

Exemple : Après construction de la TM précédente, le décideur poursuit son analyse. Plus précisément, (1) il affine son analyse en affichant les fournisseurs par pays d'origine, et (2) il focalise son analyse en sélectionnant la catégorie de produit intitulée 'Electronique'.

- (1) L'opération de forage permettant d'affiner la graduation sur l'axe d'analyse est exprimée en manipulant la dimension Fournisseurs soit sur le graphe, soit directement sur la TM obtenue précédemment (figure 2.12)
- (2) L'opération de sélection permettant de focaliser l'analyse uniquement sur les produits électroniques s'exprime en manipulant la dimension Produits sur le graphe. Cette opération ne peut être définie qu'à partir du graphe puisque la dimension impliquée n'est pas disponible dans la TM.

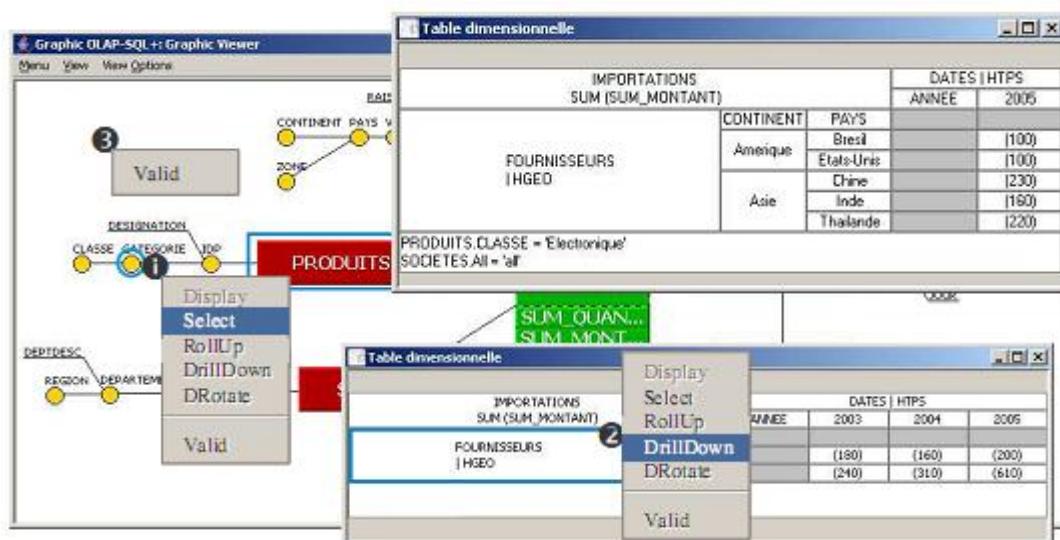


Figure 2.12 : Construction d'une requête

6.2 GRAPHIC-OLAP

l'outil GRAPHIC-OLAP permettant de définir, de manipuler et d'interroger des bases de données multidimensionnelles. Il repose sur un langage assertional OLAP-SQL et sur un langage graphique.

L'architecture de GRAPHIC-OLAP décrite à la figure suivante comporte trois niveaux.

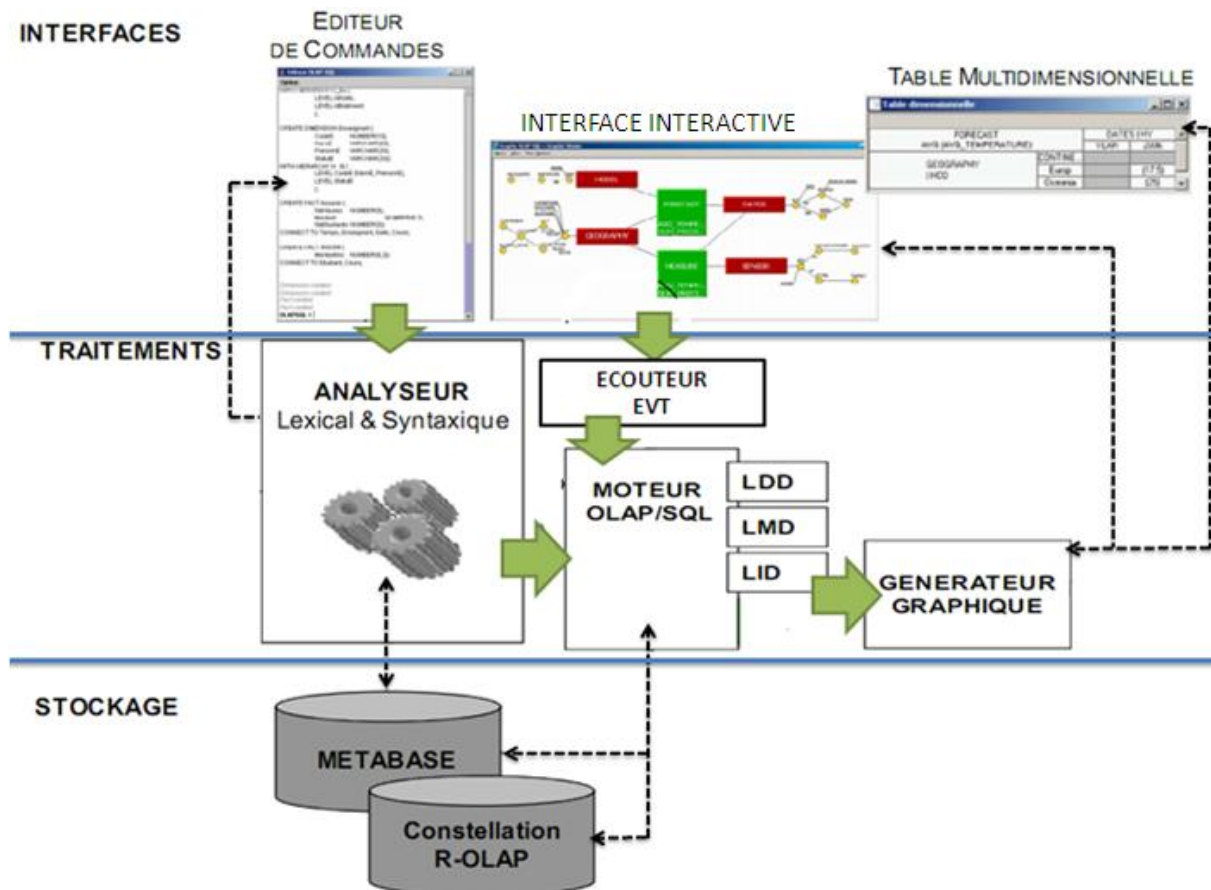


Figure 2.13 : Architecture de GRAPHIC-OLAP.

-Le niveau « **INTERFACES** » offre un ensemble d'interfaces permettant à l'utilisateur d'interagir avec la base de données multidimensionnelle. Le « Interface interactive » affiche le schéma en constellation de la base de données multidimensionnelle. L'éditeur de commandes OLAP-SQL permet de saisir textuellement des ordres assertionnels et d'obtenir des réponses (messages d'erreurs,...). Le résultat d'une commande d'interrogation permet d'obtenir une table multidimensionnelle contenant les données de l'analyse souhaitée.

-Le niveau « **TRAITEMENTS** » regroupe quatre composants. Chaque commande OLAP-SQL de définition (LDD), de manipulation (LMD) et d'interrogation (LID) est analysée lexicalement et syntaxiquement afin d'être validée. La commande valide (décomposée dans une représentation interne pseudo-algébrique) est prise en charge par le moteur OLAP/SQL chargé des transcriptions vers le SGBD relationnel de stockage. Les interactions graphiques de l'utilisateur sont également captées par l'écouteur d'événements qui peut communiquer au moteur OLAP/SQL la traduction interne des manipulations graphiques effectuées. Les ordres d'interrogations sont calculés et leurs résultats sont mis en forme par le générateur graphique dans une table multidimensionnelle.

-Le niveau « **STOCKAGE** » regroupe deux bases de données. La métabase décrit les structures multidimensionnelles de la constellation tandis que les données des faits et des dimensions sont stockées dans la base R-OLAP.

7. JPivot

JPivot est un client OLAP disposant d'une interface Web permettant de représenter un cube OLAP sous forme de tableau croisé multidimensionnel

7.1. Principales caractéristiques

La solution permet de réaliser plusieurs opérations d'analyse complémentaires telles que le drill down/up, les rotations...

Les plus : possibilité d'exporter tableaux et graphiques sous forme de fichier PDF ou de document Excel.

Les moins : lisibilité de l'interface Web amoindrie lorsque des dizaines de pages de données sont chargées.

			Mesures			
Region	Department	Positions	Actual	Budget	Variance	Variance Percent
-All Regions	-All Departments	-All Positions	\$143,530,082.00	\$143,100,389.00	\$440,593.00	0.31%
Central	-All Departments	-All Positions	\$37,323,162.00	\$38,307,608.00	-\$984,436.00	-1.21%
Eastern	-All Departments	-All Positions	\$35,248,840.00	\$35,487,801.00	-\$238,921.00	-0.67%
Southern	-All Departments	-All Positions	\$35,248,840.00	\$34,803,881.00	\$444,959.00	1.28%
Western	-All Departments	-All Positions	\$35,248,840.00	\$34,510,097.00	\$738,743.00	2.14%

Drill Through Table for Actual			
Region	Department	Positions	Actual
Southern	Sales	District Manager	700 000.00
Southern	Sales	Senior Sales Rep	421 200.00
Southern	Sales	Sales Rep	890 000.00
Southern	Sales	Account Executive	290 000.00
Southern	Sales	Pre-Sales	450 000.00
Southern	Executive Management	CEO	500 000.00
Southern	Executive Management	SVP WW Operations	249 800.00
Southern	Executive Management	SVP Strategic Development	229 000.00
Southern	Executive Management	SVP Partnerships	531 780.00
Southern	Finance	CEO	631 800.00

Figure 2.14: Le tableau des données décisionnelles JPIVOT

7.2. Navigation multidimensionnelle avec JPIVOT/Mondrian

Mondrian est un serveur OLAP (On Line Analytical Processing) ou « moteur OLAP » disponible sous licence open source EPL (Eclipse Public Licence).

Mondrian exécute des requêtes utilisant le langage MDX, également utilisé par d'autres moteurs OLAP comme celui de Microsoft SQL Server. Ce langage permet de créer des requêtes dont l'équivalent en langage SQL nécessiterait un grand nombre de requêtes et des temps d'exécution beaucoup plus longs.

Lorsque vous vous connectez à Mondrian à partir de l'adresse <http://localhost:8080/mondrian/>, vous obtenez l'interface suivante. Le premier lien vous donne accès à l'interface de JPIVOT table.

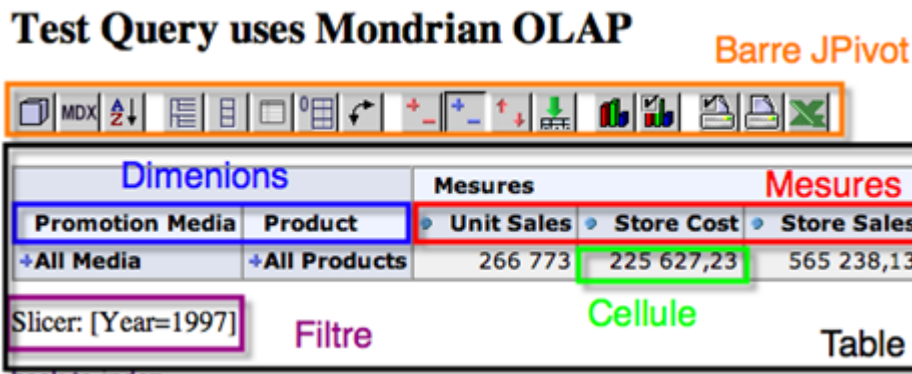
Mondrian examples:

- [JPIVOT pivot table](#) ← Cliquez ici
- [JPIVOT pivot table by XMLA](#)
- [JPIVOT with 4 hierarchies](#)
- [JPIVOT with role 'California Manager' set](#)
- [JPIVOT with arrows](#)
- [JPIVOT with colors](#)
- [Various queries formatted using the Mondrian tag-library](#)
- [Basic interface for ad hoc queries](#)
- [XML for Analysis tester](#)

Other links:

- [Mondrian home page](#)
- [Mondrian project page](#)
- [JPIVOT home page](#)
- [JPIVOT project page](#)

L'interface de JPIVOT table est composée d'une barre d'outil et d'une grille de données (table). La figure suivante en donne un aperçu de l'interface.

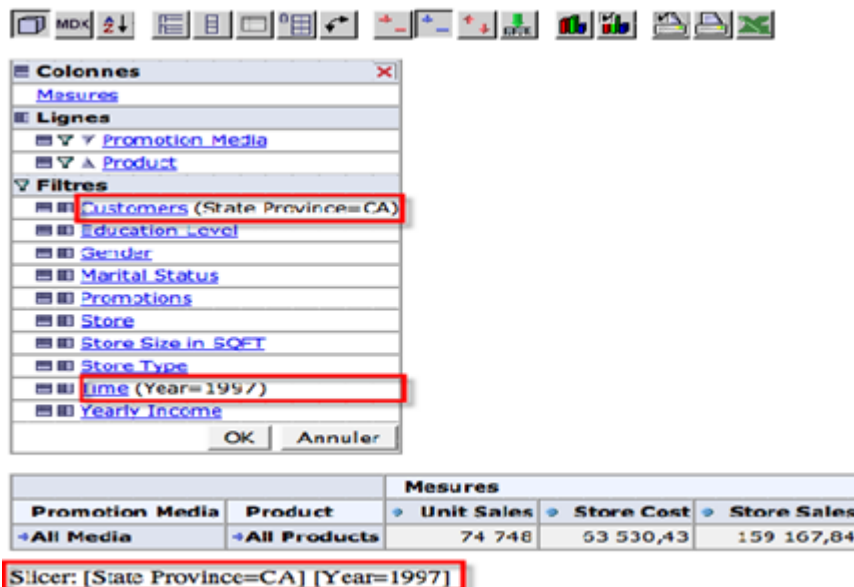


L'ensemble des boutons de la barre d'outils permet de manipuler les données avec JPivot. On peut les catégoriser selon leur fonction comme suit : gestion des dimensions, affichage de la table JPivot, navigation, gestion des graphiques et impression/export.



7.2.1 Gestion des dimensions

Le premier bouton permet d'ajouter ou d'enlever des champs (colonnes ou lignes) ou des filtres. Par exemple, les données sont filtrées suivant les clients californiens et l'année 1997. La dimension "Gender" (civilité) est ajoutée aux colonnes (en plus de la table de faits "mesures" qui contient les données "Store Cost", "Store Sales" et "Unit Sales").




Il y a trois sections : une pour les dimensions affichées sur les colonnes, une pour celles affichées sur les lignes, et une pour les autres, qui peuvent éventuellement servir comme filtre. Pour déplacer les dimensions d'une section à une autre, il faut utiliser les icônes à gauche du nom de la dimension (■ ■ ▾). Essayez, par exemple, de remplacer "State Province" par "Gender".

En cliquant sur une dimension, vous pouvez sélectionner les valeurs qui seront affichées ou pas. Les dimensions qui ne sont pas sélectionnées en tant que colonnes ou lignes sont utilisées comme filtre. Essayez, par exemple, de vous limiter aux magasins présents aux Etats-Unis.

Finalement, vous pouvez changer l'ordre des dimensions en utilisant les flèches (▲ ▼). Il est à noter qu'après chaque modification du cube, il faut la valider à l'aide du bouton "OK" en bas du formulaire pour que les modifications prennent effet.

Vous pouvez aussi trier les valeurs d'une dimension donnée comme le montre la figure suivante.





Un clic sur le bouton "OK" envoie une requête MDX à Mondrian ; qui à son tour réalise les calculs demandés et renvoie le résultat qui sera affiché sous forme d'un tableau. Il est possible à tout moment de visualiser ou de modifier une requête MDX via le bouton  de la barre de JPivot. (Nous verrons MDX plus en détail dans une section lui étant dédié.)



Le bouton  permet de trier les données selon sa convenance.





7.2.2 Affichage de la table

Les boutons   sont dédiés à des changements de présentation, plus particulièrement, la fusion des lignes ou des colonnes.








Promotion Media	Product			Mesures		
(All)	(All)	Product Family	Product Department	Unit Sales	Store Cost	Store Sales
→All Media	-All Products			74 748	63 530,43	159 167,84
→All Media	All Products	-Drink		7 102	5 662,27	14 203,24
→All Media	All Products	Drink	→Alcoholic Beverages	1 936	1 601,72	4 011,37
→All Media	All Products	Drink	→Beverages	3 886	3 190,40	8 006,53
→All Media	All Products	Drink	→Dairy	1 280	870,15	2 185,34
→All Media	All Products	+Food		53 656	45 980,35	115 193,17
→All Media	All Products	+Non-Consumable		13 990	11 887,80	29 771,43


Slicer: [State Province=CA] [Year=1997]

Le bouton  permet de supprimer les cellules dont la valeur est nulle. Le bouton  quant à lui permet d'interchanger les colonnes et les lignes.

7.2.3 Navigation

Les boutons     sont dédiés à la navigation. L'appui sur le "+" permet de faire un "drill-down" (zoomer sur les données pour découvrir de nouvelles sous catégories), l'appui sur le "-" permet de faire un "roll-up" (agréger les données sur une catégorie commune). Il est possible lors du "drill-down" d'être en mode remplacement afin de supprimer les lignes de plus haut niveau. Le bouton  permet de faire du "drill-through". Ce qui permet d'afficher des flèches vertes sur les données et ainsi d'avoir une répartition détaillée de la donnée suivant les différentes dimensions.

7.2.4 Visualisation des données originales

Activez le bouton de navigation . Vous remarquez des flèches vertes qui apparaissent dans le tableau. Déroulez les deux premières colonnes et choisissez une case avec un nombre de ventes inférieur à 100 (pour éviter de faire surcharger le serveur). Clic sur la flèche verte, les données originales correspondant à cette case s'affichent.



Promotion Media	Product	Mesures		
		Unit Sales	Store Cost	Store Sales
→All Media	-All Products	↓74 748	↓63 530,43	↓159 167,84
	-Drink	↓7 102	↓5 662,27	↓14 203,24
	-Alcoholic Beverages	↓1 936	↓1 601,72	↓4 011,37
	-Beer and Wine	↓1 936	↓1 601,72	↓4 011,37
	-Beer	↓429	↓337,11	↓833,87
	+Good	↓68	↓52,20	↓125,84
	+Pearl	↓80	↓48,78	↓119,35
	+Portsmouth	↓82	↓88,72	↓226,02
	+Top Measure	↓105	↓46,52	↓117,94
	+Walrus	↓94	↓100,89	↓244,72
	+Wine	↓1 507	↓1 264,61	↓3 177,50
	+Beverages	↓3 886	↓3 190,40	↓8 006,53
	+Dairy	↓1 280	↓870,15	↓2 185,34
	+Food	↓53 656	↓45 980,35	↓115 193,17
	+Non-Consumable	↓13 990	↓11 887,80	↓29 771,43

Slicer: [State Province=CA] [Year=1997]

Modifier les données à détailler

City	Name	Gender	Unit Sales
Los Angeles	Diane Yah	F	2.0000
Pomona	Bruce Peters	F	3.0000
Downey	Zenin Sussman	M	3.0000
West Covina	Georgeane Jennings	F	4.0000
West Covina	Coy Chamberlain	M	3.0000

Page 1/5 Aller à la page 1 Lignes/page 5

Détail de la cellule

Parcourir

7.2.5 Gestion des graphiques



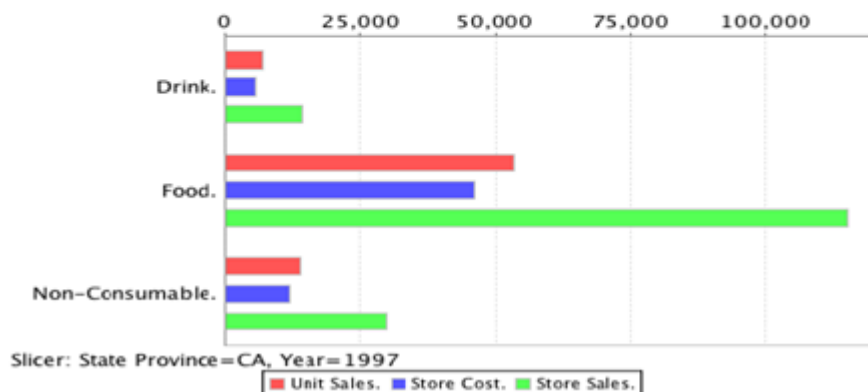
Le bouton  permet de représenter à l'aide d'un graphique les données que vous venez d'agréger. Pour choisir le type de graphique et éventuellement changer ses propriétés, il faut clic sur le bouton .

Chart Properties	
Chart Type	Horizontal Bar
Enable Drill Through	<input type="checkbox"/>
Chart Title	
Chart Title Font	SansSerif Bold 18
Horizontal axis label	
Vertical axis label	
Axes Label Font	SansSerif Plain 12
Axes Tick Label font	SansSerif Plain 12 30°
Show Legend	<input checked="" type="checkbox"/> Bottom
Legend Font	SansSerif Plain 10
Show Slicer	<input checked="" type="checkbox"/> Bottom Left
Slicer Font	SansSerif Plain 12
Chart Height	300
Chart Width	500
Background (R, G, B)	255 255 255






	Mesures		
Product	Unit Sales	Store Cost	Store Sales
Drink	7 102	5 662,27	14 203,24
Food	53 656	45 980,35	115 193,17
Non-Consumable	13 990	11 887,80	29 771,43

Slicer: [State Province=CA] [Year=1997]



7.2.6 Impression et export

Les trois derniers boutons    permettent de définir les options d'impression PDF et d'exporter en PDF ou en XLS (MS Excel) les résultats.

Print Properties	
Report Title	
Page Orientation	Portrait
Paper Size	A4
Custom Height/Width	29.70 cm 21.00 cm (0=default A4)
Table Width	<input type="checkbox"/> (off = auto) 0.00 cm
Chart on separate page	<input type="checkbox"/>

8. Business Intelligence

8.1.1 Définition

La **Business intelligence**, dénommée aussi par simplification "**Informatique Décisionnelle**" ou encore "**BI**", est vraisemblablement l'unique solution pour bâtir une entreprise proactive au sens propre du terme. La Business Intelligence désigne en effet un ensemble d'outils technologiques, méthodiquement assemblés, et déployés en parfaite cohérence avec la stratégie d'entreprise préalablement élaborée. La BI n'a d'autre finalité que de délivrer les informations pertinentes à chaque manager afin qu'ils puissent prendre le plus efficacement possible les meilleures décisions selon son contexte d'action, ses prérogatives et ses objectifs tactiques et stratégiques.

La tactique correspondant à des opérations ponctuelles.

La stratégie est-elle globale et planifiée, reprenant un certain nombre de tactiques

8.1.2 Oracle Business Intelligence

Oracle Business Intelligence est une solution intégrée, intuitif et interactif Business Intelligence qui permet de créer des rapports complets et des capacités de livraison, de la préparation des données pour la présentation finale, contre multidimensionnel OLAP ou sources de données relationnelles.

8.1.2.1 Oracle Business Intelligence Discoverer

Oracle BI Discoverer est un outil de business intelligence pour analyser les données. Il offre une solution de business intelligence intégrée qui comprend interrogation, de reporting, d'analyse et de publication Web fonctionnalité intuitive ad-hoc. Oracle BI Discoverer permet aux utilisateurs non techniques d'accéder immédiatement à l'information à partir de sources multidimensionnelles OLAP données, data marts, entrepôts de données, ou des systèmes de traitement des transactions en ligne. Le Figure 1 montre un exemple d'un tableau de bord de Business Intelligence, à l'aide des rapports d'Oracle BI Discoverer via le portail Oracle Application Server.



Figure 2.15 : Oracle Business Intelligence Discoverer exemple de tableau de bord

Ce graphique est une capture d'écran d'un tableau de bord de Discoverer Business Intelligence Oracle. Il contient plusieurs tableaux et de graphiques différents.

En utilisant les différentes composantes d'Oracle BI Discoverer, nous pouvons faire ce qui suit: Créer de nouvelles feuilles de calcul et d'analyser des données à partir de deux sources de données relationnelles et multidimensionnelles à travers le Web Analyser les données dans les feuilles de calcul existantes et enregistrer les personnalisations personnalisées de ces feuilles

Afficher découvreur des feuilles de calcul que des jauges, des tableaux, tableaux croisés, graphiques ou dans un portail style tableau de bord Avec Oracle BI Discoverer, les utilisateurs professionnels à tous les niveaux de l'organisation ont accès immédiat et sécurisé à des données provenant de diverses sources de données, le tout via un navigateur Web standard.

8. I.2.2 Reports Server Application Services Oracle (en anglais : Oracle Application Server Reports Services)

Dans un environnement d'intelligence d'affaires, de données est générée par des sources multiples dans divers formats. Publiant ces données de business intelligence doit être rapide et facile, sans nécessiter plusieurs étapes compliquées, de manipuler et de consolider les données dans un format unique.

- Reports Server Application Services Oracle fournit une plate-forme de déploiement robuste pour la création de rapports, l'utilisateur final générées dynamiquement haute qualité dans une solution évolutive, environnement sécurisé. Vous pouvez créer et publier des rapports sophistiqués à partir de n'importe quelle source de données, dans n'importe quel format, et de les déployer n'importe où avec Oracle Application Server.

Oracle AS Reports Services permet nous de faire ce qui suit:

- Accéder à des rapports dynamiquement à partir de n'importe quel navigateur sur demande
- Levier d'équilibrage de charge de niveau intermédiaire à fournir de gros volumes de rapports, sans exigences excessives sur les ressources limitées
- Eventuellement la distribution sécurisée des rapports en permettant aux utilisateurs seulement spécifiques d'accéder au rapport et à accorder un accès sécurisé.
- Lien vers les rapports publiés et fournir un soutien portlet utilisant Oracle AS Portal.

La figure 2.16 montre un exemple de rapport Oracle.

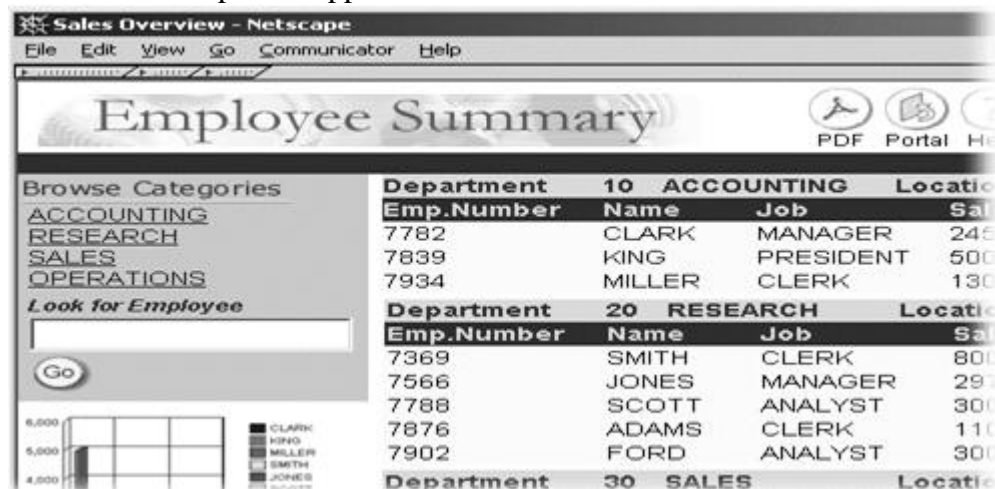


Figure 2.16 : Exemple Oracle Reports

- Ce graphique montre une capture d'écran d'un rapport Oracle de l'échantillon.

8. I.2.3 Intégration Oracle Business Intelligence

L'intégration étroite entre Oracle Application Server, Oracle Developer Suite, et la base de données Oracle offre des performances et une évolutivité. Oracle Business Intelligence Discoverer fournit la technologie qui rend les données visibles dans tous les secteurs de l'entreprise.

8. I.3 Microsoft Business Intelligence

Microsoft Business Intelligence (BI) est une suite de produits et d'outils que vous pouvez utiliser pour surveiller, analyser et planifier votre entreprise en utilisant les tableaux de bord, reporting de gestion et d'analyse.

Quelle est la Business Intelligence la plus utilisée (BI) de l'outil sur la planète?

Avec la combinaison de Microsoft Office, SharePoint, PowerPivot, PowerView et PerformancePoint Services, Business Intelligence de Microsoft (BI) empreinte est aussi grand que tout. Et avec les nouvelles améliorations apportées à Microsoft SQL Server 2005 Microsoft BI est plus puissant et plus facile à utiliser que toujours avant.



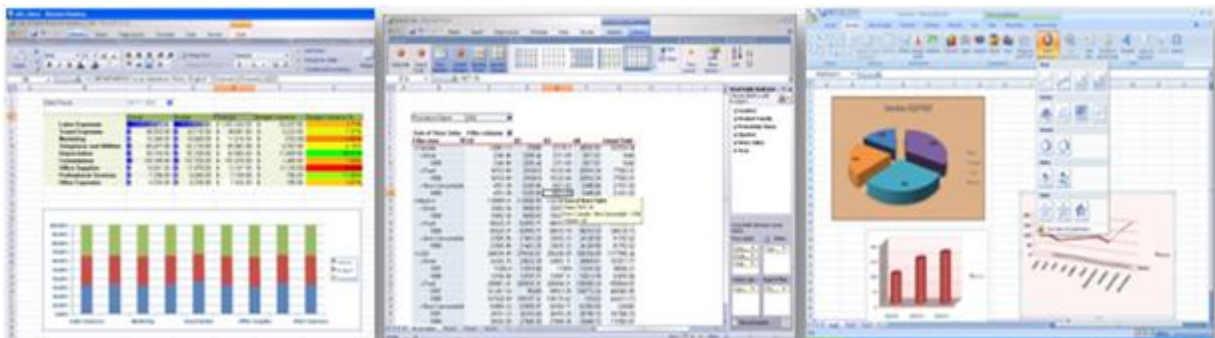
Figure 2.20 : Microsoft Business Intelligence

8. I.3.1 Microsoft Office

8. I.3.1.1 Outil client



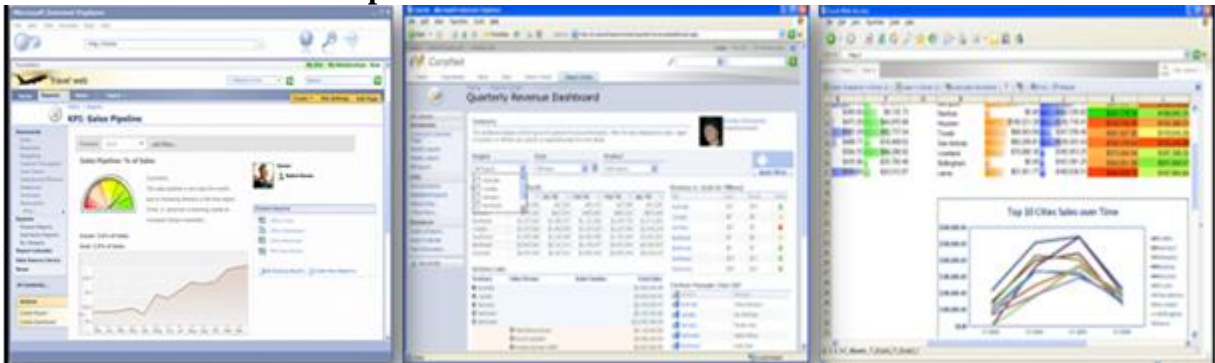
Microsoft Office Excel 2007



8. I.3.1.2 Portail d'accès



Microsoft Office Sharepoint Server 2007



Portail d'entreprise - BI Collaborative - Excel Services

8. I.3.1.3 Outil Fonctionnel



Microsoft Office PerformancePoint Server 2007

1. Monitoring : Tableau de bord pour décideurs



2. Analytics : Analyse de données avancée, Client riche ou client léger



3. Planning : Elaboration budgétaire, Traçabilité, sécurité, Excel comme outil de saisie



Analyse avec PerformancePoint Server :

- Facilité d'utilisation : Ergonomie, Assistants multiples.
- Visualisations avancées : Carte de Performance, Perspective, Vue Explorer, Arbre de décomposition.



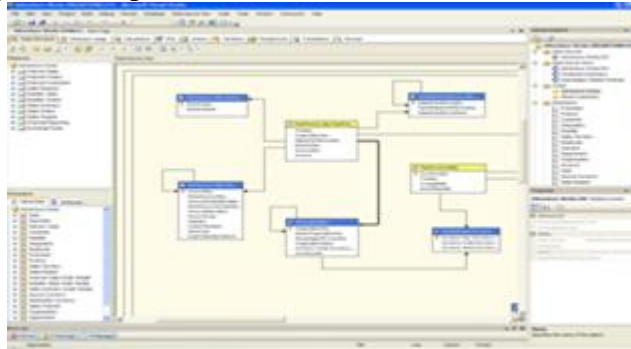
8. I.3.2 Socle Technique : SQL Server 2005

8. I.3.2.1 SQL Server Integration Services:

- ETL d'entreprise : Multi-sources, Multi-destinations, Multi-transformations
- Haute performance et évolutivité
- Nouveaux Outils : Environnement de développement, Gestion des versions, Analyse pas à pas visuelle du flux et des données, Nombreuses transformations natives

8. I.3.2.2 SQL Server Analysis Services :

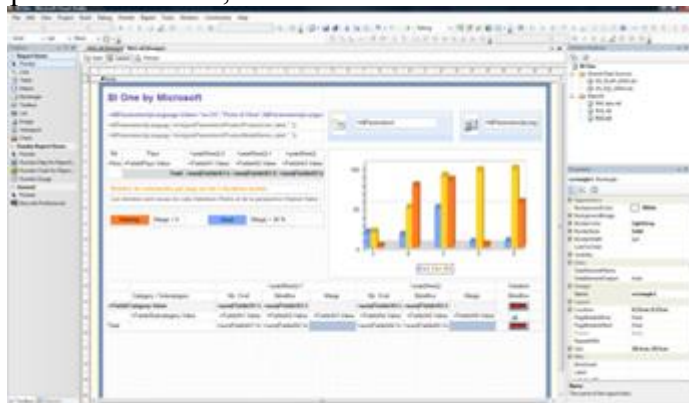
- Moteur OLAP (On-Line Analytical Processing) de SQL Server : Propose une vue multidimensionnelle et intuitive des données, Permet d'associer une vue métier aux données, Calcul d'agrégations permettant une récupération rapide des données



- Intégration complète avec Reporting Services et Integration Services.

8. I.3.2.3 SQL Server Reporting Services

- Infrastructure complète de création, d'administration et de distribution de rapports



- Basé sur XML, Web Services et .NET
- Ouvert : - Choix de la source (Oracle, SQL Server, OLAP...)
- Choix du format de sortie (HTML, PDF, Excel ...)
- Choix du canal de distribution (fichier, mail...)

9. Conclusion

Dans cet état de l'art nous avons résumé les principaux travaux réalisés sur les langages d'interrogation de base de données multidimensionnelle, Nous avons présenté le langage OLAP-SQL qui vise à intégrer les caractéristiques inhérentes à l'approche multidimensionnelle, tout en offrant la puissance et la simplicité d'un langage assertionnel. Ce langage assure les principales étapes pour développer et utiliser une base de données en constellation.

Dans le chapitre suivant nous allons expliquer en détail leurs principes d'opérations de manipulation OLAP pour bien comprendre leurs opérations OLAP, présenter l'implémentation de quelques d'opérations de la manipulation de données multidimensionnelles, ensuite nous allons proposer un nouvel opérateur OLAP.

CHAPITRE III

Contribution

Dans ce chapitre nous présentons les opérateurs OLAP classiques de base, Chaque opérateur prend en entrée un cube OLAP et fournit un autre cube en sortie. Un cube OLAP représente dans ses cellules des faits à analyser en fonction de dimensions (axes d'analyse) décrites par des attributs membres et susceptibles d'être hiérarchisées

Dans un premier temps, nous avons regroupé tous les opérateurs d'OLAP en trois catégories : paramétrage d'une TM (Forage, Rotation, Sélection), présentation d'une TM (permutation de paramètres, Calcul) et la transformation d'une TM (modification du fait, modification dimension), La table suivant résumé ces opérateurs.

Catégorie	Opérateur	Description	
Paramétrage d'une TM	Forage	DrillDown	Forage vers le bas
		RollUp	Forage vers le haut
	Rotation	Frotate	Rotation de faits
		Drotate	Rotation de dimensions
		Hrotate	Rotation de hiérarchies d'une dimension
	Sélection	Select	Sélection de valeurs de paramètres ou de mesures
UnSelect		Désélection	
Présentation d'une TM	Permutation de paramètres	Switch	Permutation de deux valeurs d'un même paramètre
		Order	Ordonnancement de valeurs de paramètres
		Nest	Permutation de deux paramètres
	Calcul	Cube	Permet de spécifier une fonction d'agrégation
		Uncube	Permet de supprimer une fonction d'agrégation
Transformation d'une TM	Modification du fait	DelM	Suppression d'une mesure
		AddM	Ajout d'une mesure
	Modification dimension	Push	Transformation d'un paramètre en mesure
		Pull	Transformation d'une mesure en paramètre.

Table 3.1 : tous les opérateurs d'OLAP

1. Existant

Les travaux qui est existe sont été de concevoir un tel langage, qui permet d'interroger une base de données multidimensionnelle, à travers des interfaces graphiques. Vue dans (Chapitre II : section 6 et 6-1)

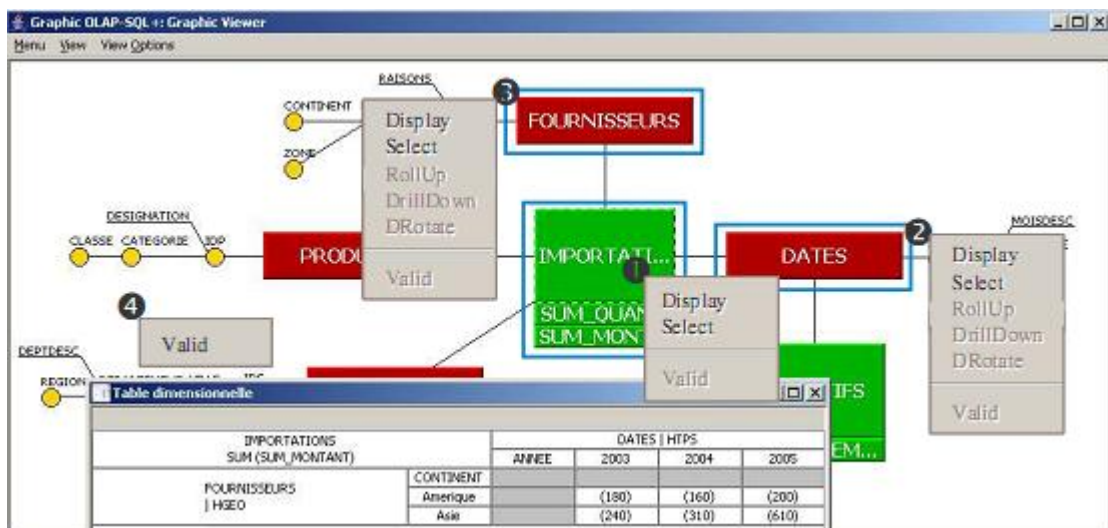


Figure 2.11 : Construction graphique d'une table multidimensionnelle.

Dans notre travail nous avons facilité la manipulation d'une TM et donner la main à l'utilisateur pour sélectionner les données.

2. Le langage utilisé

Le programme de ce travail a été écrit en langage Delphi, Delphi est un environnement de développement de type RAD (Rapid Application Development) basé sur le langage Pascal. Il permet de réaliser rapidement et simplement des applications Windows.

Cette rapidité et cette simplicité de développement sont dues à une conception visuelle de l'application. Delphi propose un ensemble très complet de composants visuels prêts à l'emploi incluant la quasi-totalité des composants Windows (boutons, boîtes de dialogue, menus, barres d'outils...) ainsi que des experts permettant de créer facilement divers types d'applications et de bibliothèques.

Pour maîtriser le développement d'une application sous Delphi, il est indispensable d'aborder les trois sujets suivants :

- le langage Pascal et la programmation orientée objet ;
- l'Environnement de Développement Intégré (EDI) de Delphi ;
- les objets de Delphi et la hiérarchie de classe de sa bibliothèque.

Nous compléterons cette approche par la connexion aux bases de données avec Delphi.

3. Implémentation des opérateurs OLAP

Nous avons implémenté quelques opérateurs d'OLAP, Ces opérateurs sont : **Display, DrillDown, RollUp, Frotate, Drotate, Hrotate, Select, UnSelect, Switch, Cube, Uncube, DelM** et **AddM**.

Ce premier opérateur est spécifique car il permet de construire une première TM à partir des composants d'une BDM. Les autres opérateurs manipulent et génèrent des TM.

Nous utilisons une base de données multidimensionnelle modèle en constellation qui contient deux faits : fait VENTES et fait ACHATS. Au niveau du fait VENTES il y a trois dimensions : Temps, Produits et Clients. Au niveau du fait ACHATS il y a aussi trois dimensions : Temps, Produits et Fournisseurs. Figure (3.1).

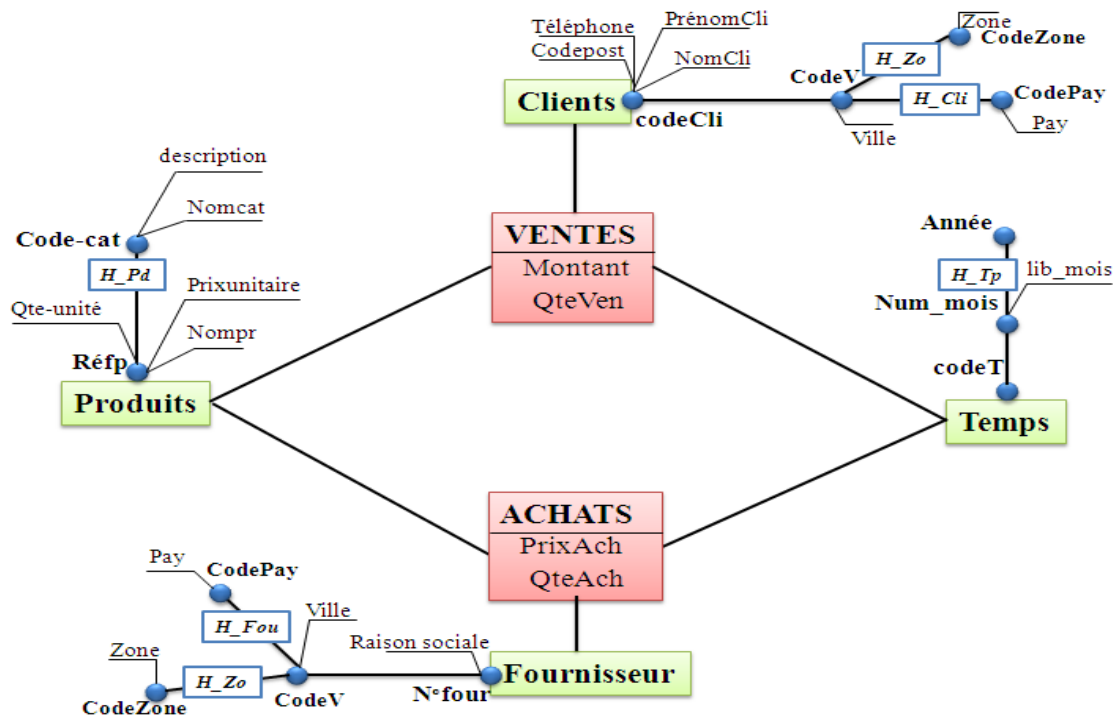


Figure 3.1 : Schéma d'une constellation de la BDM.

4. Les étapes du l'exécution

Etape 1 : la Sélection

On à sélectionner un fait (VENTES ou ACHATS) et deux dimensions qui on choisir. Comme suite :

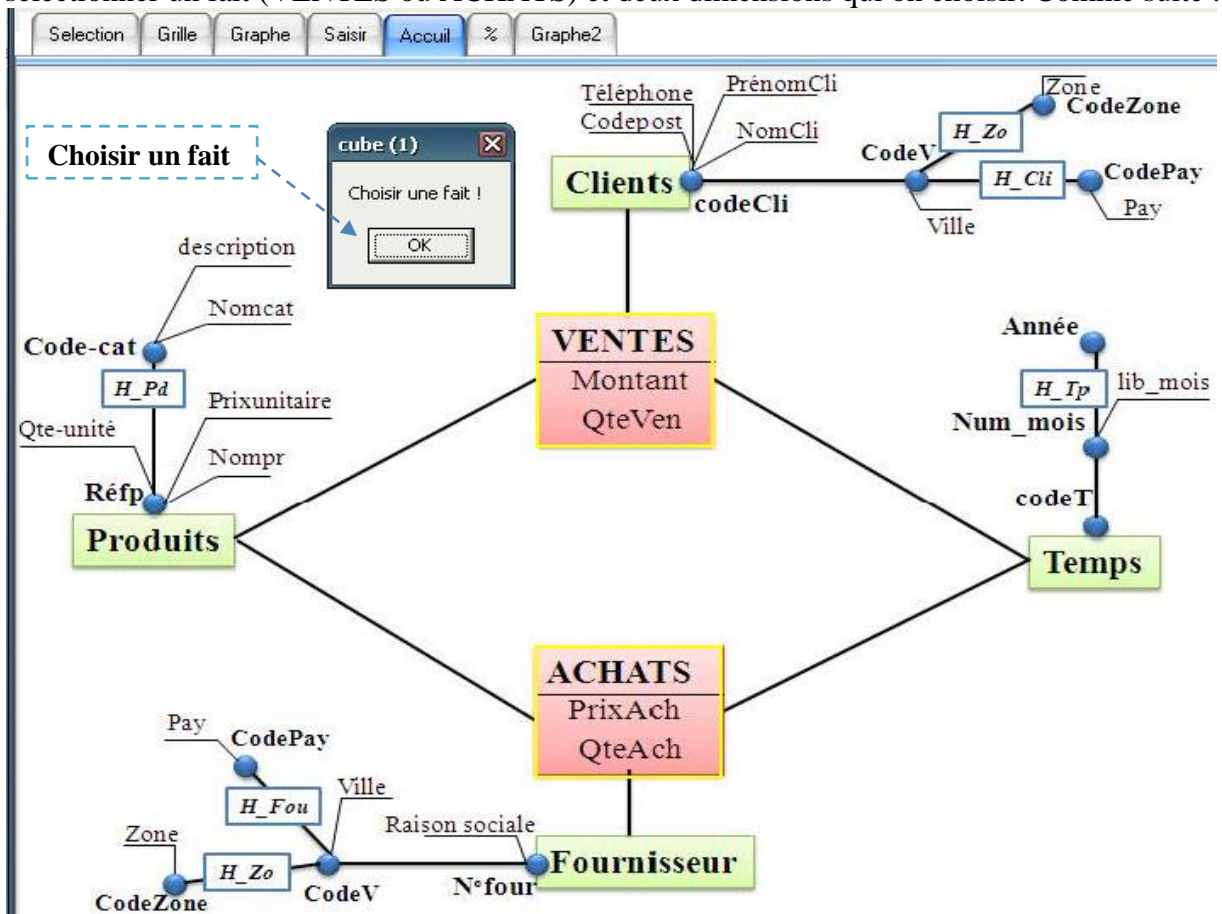


Figure 3.2 : choisir un fait.

Par exemple on choisir le fait **VENTES** et sélectionner deux dimensions (**Clients**, **Temps**).

FAIT : VENTES

Clients
Selection
Temps

CodeClient	NomCli	PrénomCli	Téléphone	CodeT	Num_mois
ALFKI	Dufour	kat	030-0074321	31/08/2000	08
ANATR	Brisetout	sed	(5) 555-4729	01/09/2000	09
ANTON	Duchemin	fate	(5) 555-3932	31/10/2000	10
AROUT	Satrob	ged	(71) 555-7788	03/11/2000	11
BERGS	das	hez	0921-12 34 65	30/12/2000	12
BLAUS	gad	deg	0621-08460	24/01/2001	01
				24/02/2001	02

Pays : France

Tout

États-Unis
Mexique
Espagne
France

Nombre = 4

Année :

Tout

2000
2001
2002
2003

Nombre = 4

DISPLAY

Figure 3.3 : Sélectionner les dimensions.

Etape 2 : l'opération de construction de TM

Algorithme DISPLAY

Variable i,j,k: Entier;

Debut

Si Fait='VENTES' Alors

////////// ***** Produits Clients

Si (Choix1.Text='Clients') Et (Choix2.Text='Produits') Alors

TM. Cells[1,0]:='Catégories'; TM. Cells[0,1]:='Pays'; TM. Cells[0,0]:='';

Pour i:=0 Jusqu'a Nbr_Produit -1 Faire TM. Cells[i+2,0]:= produit [i]; Fin Faire

Pour j:=0 Jusqu'a Nbr_Clie -1 Faire TM. Cells[i+2,j]:= client [i]; Fin Faire

remplirtab.TM; /**remplir la table

Si Fait='ACHATS' Alors// la même chose pour les autres cas

Fin Algo

En plus construire la TM par un clic sur le bouton **DISPLAY** qui dont le résultat suivant :

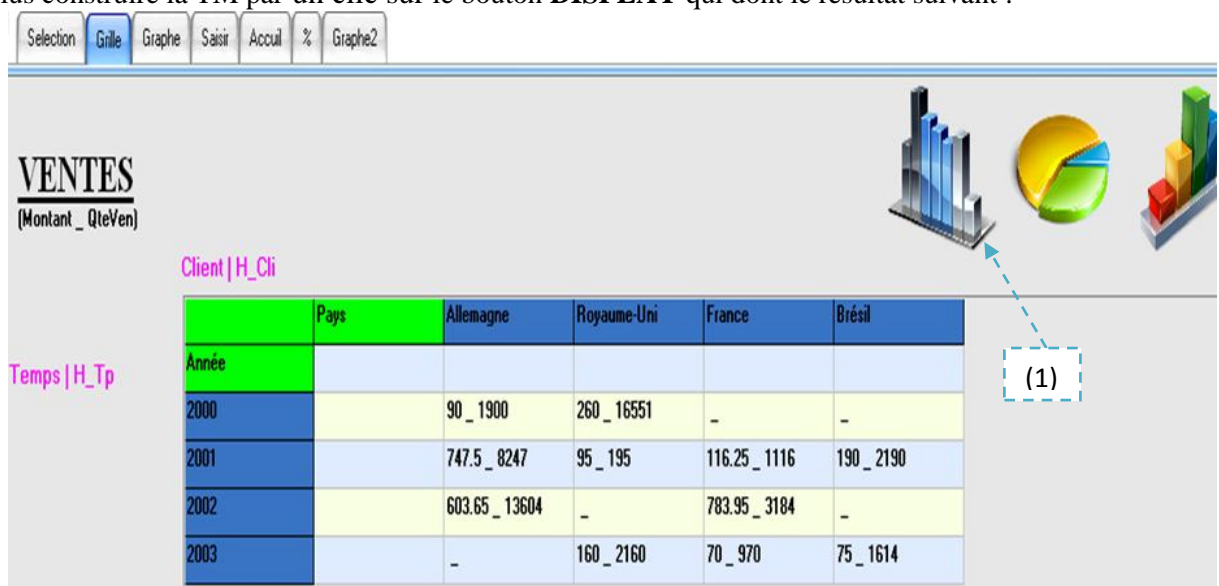


Figure 3.4 : La table multidimensionnelle de fait VENTE.

Etape 3 : afficher le graphe de la table multidimensionnel.

On clic sur ce bouton (1) qui dont les graphes suivant : (Montant et Quantité) par dimension Temps (Année) et dimension Clients (Pays)

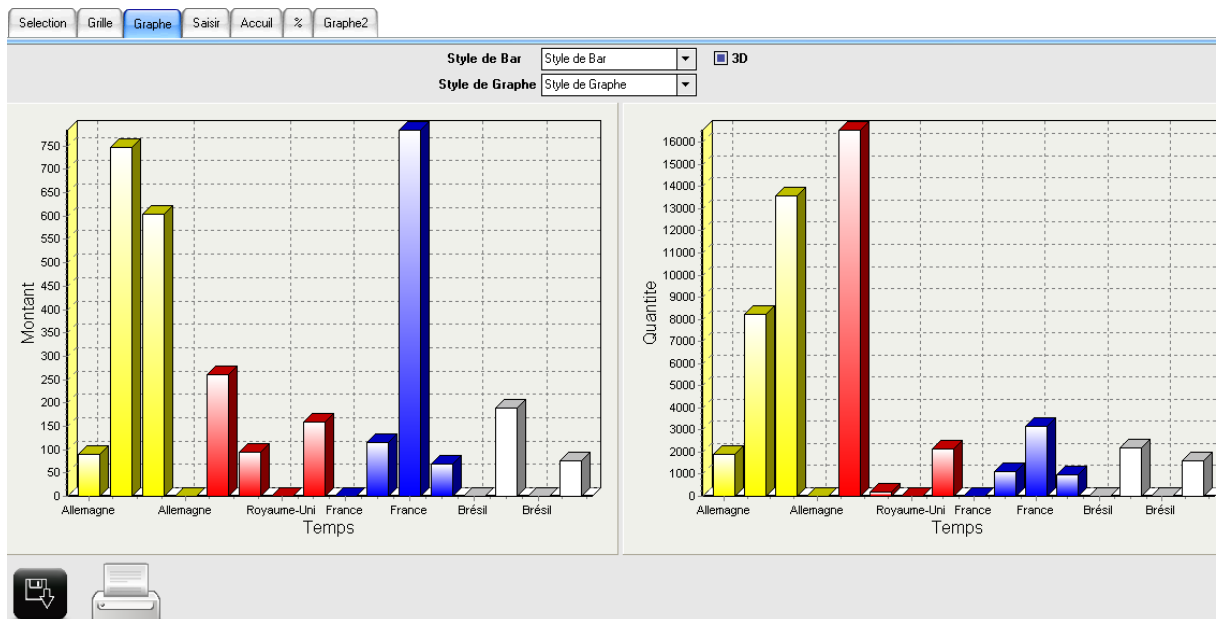


Figure 3.5 : Les graphes de la TM.

Etape 4 : les opérateurs manipulent de TM :

1-DrillDown et RollUp :

Algorithme DrillDown

Variable i,j,k: Entier;

Debut

Si Fait='VENTES' Alors

////////// **** Produits Clients

Si (Choix1.Text='Produits') Et (Choix2.Text='Clients') Alors

TM.Cells[2,0]:='Catégories'; TM.Cells[1,1]:='Villes'; TM.Cells[0,1]:='Pays'; TM.Cells[1,0]:='';

Pour i:=0 Jusqu'a Nombre_Produit -1 Faire TM.Cells[i+3,0]:= produit [i]; Fin Faire

Pour i:=0 Jusqu'a Nombre_Client -1 Faire Slect-vile.SQL=('select * from ville where codepay in (select codepay from pays where pays="" +Client[i]+"");

TM.rowCount:=TM.rowCount+ Slect-vile.RecordCount; Slect-vile.First; Fin Faire

Pour j:=1 Jusqu'a Slect-vile.RecordCount Faire

TM.Cells[0,k]:='Pays[i]; TM.Cells[1,k]:= Slect-vile.fieldbyname('ville');Slect-vile.next; k:=k+1;

Fin Faire

remplirtab.TM; /***remplir la table

Si Fait='ACHATS' Alors // la même chose pour les autres cas

Fin Algo

Par un clic droit sur la souris dimension.

Client | H_Cli

	Paramétrage	Forage	DrillDown		
	Présentation	Rotation	RollUp	ce	Brésil
Année	Transformation	Selection			
2000	Opération Proposer	0 _ 1900	260 _ 16551	-	-
2001		747.5 _ 8247	95 _ 195	116.25 _ 1116	190 _ 2190
2002		603.65 _ 13604	-	783.95 _ 3184	-
2003		-	160 _ 2160	70 _ 970	75 _ 1614

Dont le résultat suivant : pour chaque pays il donne tous leurs villes et l'opération RollUp retourner à la table précédent.

Client | H_Cli

	Pays	Allemagne	Allemagne	Allemagne	Allemagne	Royaume-Uni	Royaume-Uni
	Ville	Berlin	Aachen	Köln	Mannheim	Cowes	London
Année							
2000		90 _ 1900	-	-	-	-	260 _ 16551
2001		405 _ 405	30 _ 6530	-	312.5 _ 1312	95 _ 195	-
2002		-	-	384.15 _ 8384	219.5 _ 5220	-	-
2003		-	-	-	-	-	160 _ 2160

Figure 3.6 : Le forage vers le bas (« drill-down »)

2-Frotate : on change un sujet d'analyse (VENTES) par autre (ACHATS) comme suit :

VENTES
(Montant _ QteVen)

Client | H_Cli

Temps | H_Tp

	Paramétrage	Forage	Rotation	Frotate		
	Présentation	Rotation	Frotate	Drotate		
	Transformation	Selection	Hrotate			
2000	Opération Proposer	100	260 _ 16551			
2001		747.5 _ 8247	95 _ 195	116.25 _ 1116	190 _ 2190	
2002		603.65 _ 1360	-	783.95 _ 3184	-	
2003		-	160 _ 2160	70 _ 970	75 _ 1614	

ACHATS
(PrixAch _ QteAch)

Fournisseurs | H_Cli

Temps | H_Tp

	Pays	États-Unis	Allemagne	Mexique	France
Année					
2000		5228 _ 195	5220 _ 146	-	2619 _ 405
2001		-	1982 _ 3426	1899 _ 6325	602 _ 145
2002		-	6531 _ 1739	-	6725 _ 1653
2003		-	3748 _ 5694	-	295 _ 8008

Figure 3.7 : Rotation de fait VENTES (« FRotate »)

3. Drotate:

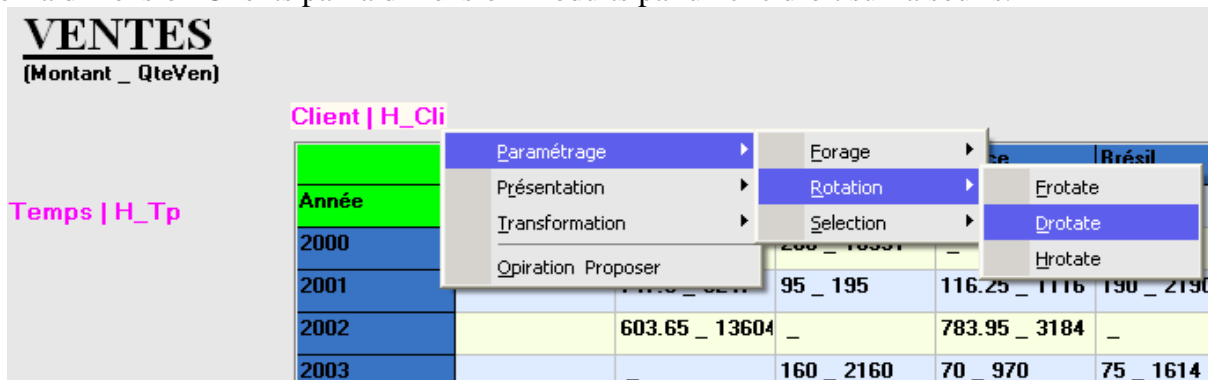
Algorithme Drotate

```

Variable i,j,k: entier;
Début
Si Fait='VENTES' Alors
    //////////// **** clients temps
Si (Choix1.Text='Clients') Et (Choix2.Text='Temps') Alors
    Pour i:=0 jusqu'a Nombre_Client-1 Faire
    Pour j:=0 jusqu'a Nombre_Temps-1 Faire
remplirtab1.SQL=('select sum(montant) from ventes where (Codet like "%'+suiStringGrid1.Cells[0,j+2]+'") Et
(CodeClient in(select CodeClient from Clients where Codeville in('+select CodeVille from Ville where
CodePay in(select Codepay from Pays where Pays="'+suiStringGrid1.Cells[i+2,0]+'"))));
remplirtab2.SQL=('select sum(qteven) from ventes where (Codet like "%'+suiStringGrid1.Cells[0,j+2]+'") Et
(CodeClient in(select CodeClient from Clients where Codeville in('+select CodeVille from Ville where
CodePay in(select Codepay from Pays where Pays="'+suiStringGrid1.Cells[i+2,0]+'"))));
    ///remplir la table
    TM.Cells[i+2,j+2]:= remplirtab1.Text+' _ '+ remplirtab2.Text;
Fin Pour;
Fin Pour;
Fin Si ;
Fin Si ;

Si Fait='ACHATS' Alors
    //////////// **** Fournisseurs temps
Si (Choix1.Text='Fournisseurs') Et (Choix2.Text='Temps') Alors
    Pour i:=0 jusqu'a Nombre_Fournisseurs-1 Faire
    Pour j:=0 jusqu'a Nombre_Temps -1 Faire
    remplirtab1.SQL=('select sum(prixach) from achats where (Codet like "%'+suiStringGrid1.Cells[0,j+2]+'")
Et (N°Four in(select N°Four from fournisseurs where Codeville in('+select CodeVille from Ville where
CodePay in(select Codepay from Pays where Pays="'+suiStringGrid1.Cells[i+2,0]+'"))));
    remplirtab2.SQL=('select sum(qteach) from achats where (Codet like "%'+suiStringGrid1.Cells[0,j+2]+'")
Et (N°Four in(select N°Four from fournisseurs where Codeville in('+select CodeVille from Ville where
CodePay in(select Codepay from Pays where Pays="'+suiStringGrid1.Cells[i+2,0]+'"))));
    ///remplir la table
    TM.Cells[i+2,j+2]:=DBEdit1 remplirtab1.Text+' _ '+ remplirtab2.Text;
Fin Pour;
Fin Pour;
Fin Si ;
Fin Si ;
Fin Algo.
    
```

Changer la dimension Clients par la dimension Produits par un clic droit sur la souris.



Dont le résultat suivant :

VENTES
(Montant _ QteVen)

Produits | H_Pd

Temps | H_Tp

	Catégories	Boissons	Produits secs	Desserts
Année				
2000		90 _ 1900	-	-
2001		95 _ 195	30 _ 6530	538.25 _ 738
2002		22.5 _ 322	228 _ 5228	495.65 _ 8796
2003		-	12.5 _ 1012	90 _ 690

Figure 3.8 : Rotation de Dimension Clients (« DRotate »)

4. Hrotate: toujours par un clic droite sur la souris dimension et choisir l’opération Hrotate. par exemple on change l’hierarchie H_Cli par l’hierarchie H_Zon sur la dimension Clients.

5. Select et UnSelect: Restriction sur les valeurs des paramètres et/ou les mesures. Par exemple : Select (TM, pays= 'France' & sum(montant)>100) = TM1.

France Montant >= 100

Client | H_Cli

	Pays	Allemagne	Royaume-Uni	France	Brésil
Année					
2000		90 _ 1900	260 _ 16551	-	-
2001		747.5 _ 8247	95 _ 195	116.25 _ 1116	190 _ 2190
2002		603.65 _ 13604	-	783.95 _ 3184	-
2003		-	160 _ 2160	70 _ 970	75 _ 1614

Client | H_Cli

	Pays	France
Année		
2000		-
2001		116.25 _ 1116
2002		783.95 _ 3184
2003		-

Pays :='France' et sum(Montant)>=100

Figure 3.9 : Opérateur Select

6. Switch: Permutation des valeurs v_1 et v_2 du paramètre p, **Switch (TS, D, P, V1, V2)**. Par exemple **Switch (TS, CLIENTS, pays, Allemagne, France) = TR**

7. Cube et Uncube :

Cube : Ajout de lignes et de colonnes contenant l'agrégation en fin de tableau **Cube (TS, (SUM,all)) =TR** et **Uncube** l’opération inverse comme suite :

Client H_Cli						
	Pays	Allemagne	Royaume-Uni	France	Brésil	All
Année						
2000		90 _ 1900	260 _ 16551	_	_	350 _ 18451
2001		747.5 _ 8247	95 _ 195	116.25 _ 1116	190 _ 2190	1148.75 _ 1174
2002		603.65 _ 13604	_	783.95 _ 3184	_	1387.6 _ 16788
2003		_	160 _ 2160	70 _ 970	75 _ 1614	305 _ 4744
All		1441.15 _ 2375	515 _ 18906	970.2 _ 5270	265 _ 3804	6382.7 _ 10348

Figure 3.10 : Opérateur Cube

8. DelM et AddM : ajout/suppression d'une mesure dans un fait : $ADDM / DelM (TS, f(m)) = TR$

9. L'opération proposée

Dans toutes les opérations précédant on travail toujours sur des informations des deux dimensions, maintenant dans cette opération proposée on ajoute une autre information sur la troisième dimension. Par exemple dans le fait VENTES on prend deux dimensions: Produits et Temps. Si on pose la question suivant :

Quels sont les pays qu'on prend la catégorie Dessert pendant les années 2001 et 2003 ?

Pour répondre il faut utiliser les opérations suivantes :

1. a **Drotate** : pour changer la dimension temps par la dimension clients :

$$Drotate (T1, Temps, Clients) = T2$$

1. b **Select** : pour sélectionner tout les pays qu'on prend la catégorie dessert :

$$Select (T2, catégorie='dessert') = T3 ; resultat.1 = ensemble des pays ou la catégorie='dessert'$$

2. a **Drotate** : pour changer la dimension produit par temps.

$$Drotate (T3, Produits, Temps) = T4$$

2. b **Select** : pour sélectionner tout les pays des années 2001 et 2003 :

$$Select (T4, années='2001' et '2003') = T5 ; resultat.2 = ensemble des pays ou années='2001' et '2003'$$

3. **Intersection** : resultat.1 \cap resultat.2

Alors nous propose une nouvelle opération qui facilite tout ces étapes.

- Algorithme Opération Proposer

Variable i, j, k: entier;

Début

Si Fait='VENTES' **Alors**

////////// ****** clients temps**

Si (Choix1.Text='Clients') **ET** (Choix2.Text='Temps') **Alors**

Pour i:=0 **Jusqu'a** Nombre_Client-1 **Faire**

Pour j:=0 **Jusqu'a** Nombre_Temps-1 **Faire**

Remplirtab.SQL=('select codecatégorie from catégories where codecatégorie

in(' + 'select codecatégorie from produits where réfproduit in(select réfproduit from ventes

where (Codet ... // // **** requête SQL retourne tous les catégories de la table Clients et temps.

remplirtab.TM; // // ****remplir la table

Fin Pour

Fin Pour

Fin Si

Fin Si

```

Si Fait='ACHATS' Alors
////////// **** Fournisseurs temps
    Si (Choix1.Text= 'Fournisseurs') ET (Choix2.Text='Temps') Alors
        Pour i:=0 Jusqu'a Nombre_Fournisseurs-1 Faire
            Pour j:=0 Jusqu'a Nombre_Temps-1 Faire
Remplirtab.SQL=('select codecategorie from categories where codecategorie
in('+ 'select codecategorie from produits where réproduit in(select réproduit from ventes
where (Codet ... //// **** requête SQL retourne tous les catégories de la table Fournisseur et
temps.
remplirtab.TM; ////**remplir la table
        Fin Pour
        Fin Pour
    Fin Si
Fin Si
Fin Algo
    
```

VENTES
(Montant _ QteVen)

Produits | H_Pd

Temps | H_Tp

Année	Condiments	Desserts	Produits secs	Viandes
2000	-	-	-	-
2001	116.25 _ 1116	538.25 _ 738	30 _ 6530	195 _ 4195
2002	22.5 _ 322	495.65 _ 8796	228 _ 5228	618.95 _ 2619
2003	280 _ 13080	90 _ 690	12.5 _ 1012	-

Qui dont le résultat suivant :

VENTES
(Montant _ QteVen)

Produits | H_Pd

Temps | H_Tp

Année	Catégories	Boissons	Condiments	Desserts	Produits secs	Viandes	Pâtes et cé
2000		P5 -					
2001		P6 -	P8 -	P5 - P1 - P13	P5 -	P7 -	
2002		P15 -		P8 - P5 - P8 - P5 -		P8 -	P14 - P8 -
2003			P6 - P14 - P8 P13 -	P9 -			

Figure 3.11 : Opérateur Proposer

Donc il affiche dans la TM tout les pays : dans l'année 2001(**Allemagne, Etats-Unis, Suède**) et dans l'année 2003 (**Suède**).

La même chose pour les autres dimensions, si on à la dimension Produit et Client on peut donne les dates de chaque pays prend une catégorie du produit comme suite.

VENTES (Montant _ QteVen)		Produits H_Pd				
Clients H_Cli		Catégories	Desserts	Condiments	Boissons	
		Pays				
		Allemagne		06-11-2001 - 07-05-2002		31-08-2000
		Royaume-Uni			03-06-2003 -	30-10-2001
		France		29-01-2002 - 06-05-2002	07-03-2001 - 05-06-2003	
		Brésil				

L'opération proposée donne un bon résultat pour certaines questions, mais ne peut pas être considéré comme optimal dans certains cas. Il est également juste une suggestion de notre part peut être développé.

Conclusion

Tout au long de ce mémoire, nous concluons que Les entrepôts de données constituent un terrain fertile pour effectuer de nouvelles recherches. Aussi, les perspectives associées à ce domaine de recherche sont nombreuses.

DataMart est un sous-ensemble du DataWarehouse qui concerne une branche particulière de l'entreprise (dépendant ou indépendant). Ex : On y fait appel au Business Intelligence (BI). Il s'agit d'un ensemble de procédés pour la collecte, la comparaison et l'analyse d'informations décisionnelles. Parmi les procédés existants du Business Intelligence, il y a OLAP qui fait l'objet de ce travail. OLAP ou On-Line Analytical Processing est une "catégorie de logiciels axés sur l'exploration et l'analyse rapide des données selon une approche multidimensionnelle à plusieurs niveaux d'agrégation."

La vocation d'un entrepôt de données est l'analyse de données pour l'aide à la décision dans les entreprises. Tout de même, la modélisation multidimensionnelle est la base des entrepôts de données et l'analyse en ligne (OLAP).

L'objectif de ce mémoire est de faire une étude comparative entre les langages d'interrogation de base de données multidimensionnelle (BDM), les principes de la manipulation OLAP pour nous proposons un nouvel opérateur OLAP.

Les objectifs attendus par l'utilisation d'OLAP sont les suivants :

1. Assistance pour une analyse optimale des données sans se focaliser sur les moyens utilisés (abstraction),
2. Rapidité et facilité,
3. Visualisation multidimensionnelle des données.

Bibliographie

[Abelló et al., 2002] A. ABELLO, J. SAMOS, F. SALTOR, «*YAM²: (Yet another multidimensional model): An Extension of UML*», International Database Engineering and Applications Symposium (IDEAS'02), Juillet 2002.

[Abelló et al., 2003] A. ABELLO, J. SAMOS, F. SALTOR, «*Implementing operations to navigate semantic star schemas*», International Workshop on Data Warehousing and OLAP (DOLAP 2003), New Orleans, (Louisiana, USA), 2003.

[Abelló et al., 2006] A. ABELLO, J. SAMOS, F. SALTOR, «*YAM²: a multidimensional conceptual model extending UML*», Information Systems (IS), Septembre 2006.

[Agrawal, et al., 97] R. AGRAWAL, A. GUPTA et S. SARAWAGI, «*Modeling Multidimensional Databases*». In A. Gupta et P.-A. Larson (Eds.), International Conference Data Engineering, 1997.

[Bédard et al, 97] BEDARD, Y., «*Spatial OLAP*», Vidéoconférence, 2ème Forum annuel sur la R-D, Géomatique VI: Un monde accessible, 13-14 novembre 1997, Montréal.

[Cabibbo et al., 1997] L. CABIBBO, R. TORLONE, «*Querying Multidimensional Databases*», International Workshop on Database Programming Languages (DBPL-6), 1997.

[Caron98] CARON, P.Y, Étude du potentiel de OLAP pour supporter l'analyse spatio-temporelle, Mémoire de M. Sc., Faculté de foresterie et géomatique, Université Laval, 1998.

[Cod93] CODD, E.F., S.B. CODD & C.T. SALLEY, «*Providing OLAP (On-Line Analytical Processing) to User-Analysts: An IT Mandate*», Rapport technique, E.F. Codd and associates, 1993.

[Datta et Thomas, 1999] A. DATTA, H. THOMAS, «*The cube data model: a conceptual model and algebra for on-line analytical processing in data warehouses*», Decision Support Systems (DSS), December 1999

[Franconi, et al., 2004] E. FRANCONNI, A. KAMBLE, «*A Data Warehouse Conceptual Data Model*», International Conference on Scientific and Statistical Database Management (SSDBM'04), Juin 2004.

[Gray et al.,1996] J. GRAY, A. BOSWORTH, A. LAYMAN et H. PIRAHESH, «*Data cube: a relational aggregation operator generalizing group-by, cross-tabs and sub-totals*», Int'l Conf. on Data Engineering, Mars 1996.

[Gyssens et al., 1997] M. GYSSENS, L. V. S. LAKSHMANAN, « *A Foundation for Multi-dimensional Databases* », International Conference on Very Large Data Bases (VLDB'97), Athens (Greece), Aout 1997.

[Inmon, 96] W. H. INMON, « *Building the Data Warehouse* », John Wiley and Sons, New York, NY, deuxième édition, ISBN: 04771-14161-5, 1996.

[Kimball, 1996] R. KIMBALL, « *The Data Warehouse Toolkit: Practical Techniques for Building Dimensional Data Warehouses* ». John Wiley & Sons, 1996.

[Lehner, 1998] W. LEHNER, « *Modeling Large Scale OLAP Scenarios* ». Proceedings of the International Conference on Extending Database Technology, Mars 1998.

[Li et Wang, 1996] C. LI, X.S. WANG, « *A Data Model for Supporting On-Line Analytical Processing* », International Conference on Information and Knowledge Management (CIKM'96), Novembre 1996.

[O.Teste,2010] Olivier Teste « *Modélisation et manipulation des systèmes OLAP : de l'intégration des documents à l'utilisateur* », 2010.

[Pedersen, et al., 2001] T. B. PEDERSEN, C. S. JENSEN, C. E. DYRESON, « *A foundation for capturing and querying complex multidimensional data* », Information Systems (IS), Juillet 2001.

[Rafanelli, 2003] M. RAFANELLI, « *Operators for Multidimensional Aggregate Data* », 2003.

[Ravat, Teste, Zurfluh, 02] F. RAVAT, O. TESTE, G. ZURFLUH, « *Langages pour Bases Multidimensionnelles : OLAP-SQL* ». Article de recherche, Revue des Sciences et Technologies de l'Information, ISI (Ingénierie des Systèmes d'Information), novembre 2002.

Cet article définit un langage de définition de données, de manipulation de données et de contrôle de données pour bases de données multidimensionnelles implantées en relationnel. Ce langage se repose sur l'extension multidimensionnelle du standard SQL.

[Ravat, Teste, Zurfluh, 06] F. RAVAT, O. TESTE, G. ZURFLUH, « *Algèbre OLAP et langage graphique* », Informatique des Organisations et Systèmes d'Information et de Décision (INFORSID'06), Mai 2006.

[Ravat, Teste, Tournier, Zurfluh, 2007] F. RAVAT, O. TESTE, R. TOURNIER, G. ZURFLUH, « *Querying Multidimensional Databases* ». 11th East-European Conference on Advances in Databases and Information Systems (ADBIS'07), Springer-Verlag, LNCS 4690, Y.E. Ioannidis, B. Novikov, B. Rachev, septembre 2007.

[Ravat, Teste, Tournier, Zurfluh, 2008] F. RAVAT, O. TESTE, R. TOURNIER, G. ZURFLUH, « *Algebraic and graphic languages for OLAP manipulations* », International Journal of Data Warehousing and Mining, 2008.

[Torlone, 2003] R. TORLONE, « *Conceptual Multidimensional Models* », 2003.

[Widom, 95] J WIDOM, « *Research problems in data warehousing* » International Conference on Information and Knowledge Management (CIKM'95), Baltimore (Maryland, USA), Novembre 1995.