

ÉCOLE DES HAUTES ÉTUDES COMMERCIALES
AFFILIÉE À L'UNIVERSITÉ DE MONTRÉAL

**Étude comparative de démarches de conception de modèles
multidimensionnels utilisés pour des entrepôts de données.**

par

Antoine ROBERT

Sciences de la gestion

Mémoire présenté en vue de l'obtention
du grade de maître ès sciences
(M. Sc.)

Juin 2002

m 2002
No 45

© Antoine Robert, 2002



DÉCLARATION DE L'ÉTUDIANT
ÉTHIQUE EN RECHERCHE AUPRÈS DES ÊTRES HUMAINS

Recherche sans collecte directe d'informations

Cette recherche n'impliquait pas une collecte directe d'informations auprès de personnes (exemples : entrevues, questionnaires, appels téléphoniques, groupes de discussion, tests, observations participantes, communications écrites ou électroniques, etc.).

Cette recherche n'impliquait pas une consultation de documents, de dossiers ou de banques de données existants qui ne font pas partie du domaine public et qui contiennent des informations sur des personnes.

Titre de la
recherche :

Étude comparative de démarches de
conception de modèles multidimensionnels
utilisés pour des entrepôts de données.

Nom de l'étudiant :

Antoine ROBERT

Signature :

A. Robert

Date :

25 avril 2002

ÉCOLE DES HAUTES ÉTUDES COMMERCIALES
AFFILIÉE À L'UNIVERSITÉ DE MONTRÉAL

Ce mémoire intitulé :

**Étude comparative de démarches de conception de modèles
multidimensionnels utilisés pour des entrepôts de données.**

présenté par

Antoine ROBERT

A été évalué par un jury composé
des personnes suivantes :

Paul Mireault

Olivier Gerbé

Denis Paradis

Mémoire accepté le :

Sommaire

Ce travail a pour but d'étudier différentes démarches de conception de modèles multidimensionnels utilisés pour des entrepôts de données et de comparer les modèles construits à partir des différentes démarches. Notre étude permet de guider le choix de la démarche à suivre pour mettre en place des entrepôts de données. La démarche pourra être choisie en fonction de la disponibilité d'un modèle des données de l'entreprise, de la qualité de ce modèle, de la possibilité et de l'opportunité de faire appel à un groupe d'utilisateurs pour mener à bien le projet d'entrepôt de données et de la disponibilité d'une liste de requêtes que les utilisateurs utiliseront pour effectuer leurs analyses. Dans le cadre de ce travail, nous avons choisi quatre démarches de conception en fonction de la manière et de l'ordre de détermination des éléments du modèle multi dimensionnel.

En fonction des résultats de notre travail, le choix de la démarche doit être adapté à la situation. En effet, s'il n'est pas possible de compter sur la participation des utilisateurs à cause de leur éloignement ou encore à cause de la nature du domaine des affaires, il est possible d'utiliser une démarche qui ne fait appel qu'au modèle opérationnel. Il faut toutefois prêter une grande attention à la qualité des modèles entité-relation dans la mesure où le développement de l'entrepôt de données repose uniquement sur celui-ci. Il faut aussi avoir conscience que la granularité de l'entrepôt de données peut empêcher certaines analyses. Un entrepôt de données pourra être construit en utilisant une démarche faisant appel à la fois au modèle entité-relation des données de l'entreprise lorsque ce modèle existe et qu'il est possible de faire appel aux utilisateurs pour dresser une liste de requêtes ou un tableau des rôles des attributs de ce modèle. Dans les cas où il n'existe pas de modèle entité-relation des données de l'entreprise, il sera possible d'appliquer la démarche proposée par Kimball (1996) qui fait appel à l'analyse des processus d'affaire.

Remerciements

Je tiens à remercier Monsieur le Professeur Paul Mireault qui m'a suivi tout au long de ce travail et qui m'a guidé au cours de la recherche. Ses précieux conseils et ses recommandations judicieuses m'ont permis de mener à bien cette recherche et de la conduire de manière efficace.

Je dédie ce mémoire à Lyne, Charlène et Thomas qui m'ont soutenu tout au long de ces études.

Sommaire.....	i
Remerciements.....	i
1. Introduction	1
2. Revue de la littérature.....	6
2.1. Définition d'un entrepôt de données.....	6
2.2. Techniques d'analyse.....	10
2.2.1. Requêtes et rapports.....	11
2.2.2. Analyse multidimensionnelle	12
2.2.3. Forage des données.....	13
2.3. Architecture des données.....	14
2.3.1. Données opérationnelles.....	15
2.3.2. Données réconciliées	15
2.3.3. Données dérivées.....	15
2.3.4. Rôle des métadonnées	15
2.4. Modélisation multidimensionnelle	16
2.4.1. Modèle étoile.....	16
2.4.2. Le modèle de constellation (« Snowflake »).....	20
2.5. Mini entrepôt (data Mart)	21
2.6. Outils OLAP	22
2.6.1. Multidimensional OLAP (MOLAP).....	24
2.6.2. Relational OLAP (ROLAP).....	24
2.7. Le processus de développement	25
3. Méthodologie.....	28
4. Le modèle de Tarigomi et les requêtes	30
4.1. Tarigomi	30
4.2. Requêtes.....	32
4.2.1. Valeur moyenne des montants mensuels totaux par client.....	32
4.2.2. Nombre de commandes et ventes totales par vendeur, par client, par mois, par année.....	33
4.2.3. Délai moyen de livraison, par client, par mois, par année	33

4.2.4. Nombre moyen de lignes de commande par commande pour chaque vendeur, pour chaque mois de toutes les années confondues	34
4.2.5. Valeur commandée par marque par jour de la semaine en pour cent de la valeur totale des commandes	34
4.2.6. Rabais en pour-cent par vendeur, par marque, par mois et par année	34
4.2.7. Ventes totales par année, par client, par marque	35
4.2.8. Prix moyen de vente effectif par produit par année, par mois	35
4.2.9. Nombre de transferts entre entrepôt par produit, par année, par mois	36
4.2.10. Quantité moyenne quotidienne en stock par produit, par année, par mois	36
4.2.11. Synthèse des requêtes	37
5. Construction des modèles	39
5.1. Moody et Kortink	39
5.1.1. Exemple	40
5.1.2. Classification des entités	41
5.1.3. Identification des hiérarchies	45
5.1.4. Production du modèle multidimensionnel	47
5.1.5. Évaluation et raffinement	55
5.1.6. Analyse par requêtes SQL	55
5.1.7. Analyse par Discoverer	56
5.2. Hüsemann	57
5.2.1. Terminologie et notation	58
5.2.2. Démarche de conception	59
5.2.3. Production du modèle multidimensionnel	72
5.2.4. Analyse par requêtes SQL	73
5.2.5. Analyse par Discoverer	74
5.3. Kimball	75
5.3.1. Analyse du processus d'affaire et identification des tables de fait	75
5.3.2. Détermination de la granularité de chaque table de fait	76
5.3.3. Détermination des dimensions et de leur granularité	76
5.3.4. Détermination des mesures des tables de fait	79
5.3.5. Attributs de chaque dimension	79
5.3.6. Analyse par requêtes SQL	80
5.3.7. Analyse par Discoverer	81
5.4. Ballard	82
5.4.1. Granularité des mesures	82
5.4.2. Détermination des dimensions	82
5.4.3. Détermination des mesures	83
5.4.4. Détermination des relations entre les requêtes et les dimensions ou les mesures	84

5.4.5. Ajout d'une dimension temporelle et construction des attributs des dimensions.....	85
5.4.6. Création des faits	86
5.4.7. Modification des faits en fonction de la granularité et de l'additivité des mesures.....	86
5.4.8. Fusion des faits	88
5.4.9. Production du modèle multidimensionnel	89
5.4.10. Analyse par requêtes SQL	92
5.4.11. Analyse par Discoverer	93
6. Comparaison et critique des démarches.....	94
6.1. Pré-requis à la démarche.....	94
6.2. Mise en place de la démarche	95
6.2.1. Implication des utilisateurs	95
6.2.2. Importance de la qualité du modèle entité-relation.....	96
6.2.3. Détermination des mesures.....	97
6.2.4. Détermination des dimensions.....	97
6.2.5. Détermination des faits	97
6.2.6. Description explicite des hiérarchies	98
6.2.7. Définition des contraintes d'agrégation	99
6.2.8. Prise en compte de la granularité	99
6.3. Résultats des démarches	100
6.4. Utilisation des modèles par Discoverer	101
7. Conclusions	103
8. Références	105
9. Annexes	107
9.1. Discoverer.....	107
9.2. Données simplifiées.....	110
9.3. Requêtes.....	111
9.3.1. Valeur moyenne des montants mensuels totaux par client.....	111
9.3.2. Nombre de commandes et ventes totales par vendeur, par client, par mois, par année.....	112
9.3.3. Délai moyen de livraison, par client, par mois, par année	114
9.3.4. Nombre moyen de lignes de commande par commande pour chaque vendeur, pour chaque mois de toutes les années confondues	115
9.3.5. Valeur commandée par marque par jour de la semaine en pour cent de la valeur totale des commandes.....	116
9.3.6. Rabais en pour-cent par vendeur, par marque, par mois et par année	117
9.3.7. Ventes totales par année, par client, par magasin, par marque.....	118

9.3.8. Prix moyen de vente effectif par produit par année, par mois	119
9.4. Création des vues pour les modèles multi dimensionnels	120
9.4.1. Modèle de Moody et Kortink (2000).....	120
9.4.2. Modèle de Husemann et al (2000).....	122
9.4.3. Modèle de Kimball (1996).....	124
9.4.4. Modèle de Ballard et al (1998)	126
9.5. Commandes SQL pour les requêtes.....	128
9.5.1. Sur le modèle selon MK	128
9.5.2. Sur le modèle de HU	131
9.5.3. Sur le modèle de KI.....	135
9.5.4. Sur le modèle de BA.....	138

Liste des figures

Figure 1.1: Taux de succès des entrepôts de données (Meta Group, 1999).....	2
Figure 1.2: Éléments d'un entrepôt de données.....	4
Figure 2.1: définition d'un entrepôt de données selon Inmon (2000).	7
Figure 2.2 : les données de l'entrepôt de données sont intégrées à partir de différentes sources.....	8
Figure 2.3: les données de l'entrepôt de données dépendent du temps.	8
Figure 2.4: les données de l'entrepôt de données sont permanentes.	9
Figure 2.5: Le processus de requête et rapport débute par la définition de la requête et se termine par la livraison du rapport.....	12
Figure 2.6: Navigation à l'intérieur d'une dimension (drill down ou roll up) pour obtenir plus de détails ou au contraire regrouper les données.....	13
Figure 2.7: Le modèle étoile.....	17
Figure 2.8: La représentation du modèle multidimensionnel par un cube.....	18
Figure 2.9: Exemple de schéma de constellation.....	20
Figure 2.10 : Étapes de développement d'un entrepôt de données selon C. Ballard et al. (1998).....	26
Figure 4.1 : Diagramme de structure des données de Tarigomi tel qu'il a été développé dans ce travail pour tenir compte des approvisionnements et introduire la notion d'entrepôt.	31
Figure 5.1 : Exemple de modèle entité-relation.....	41
Figure 5.2 : Classement des entités.....	43
Figure 5.3 : Schéma entité-relation de Tarigomi dans lequel les entités de transaction ont été représentées en gris foncé, les entités composantes en gris clair et les entités de classification en blanc.	44

Figure 5.4 : Rabattement de l'entité State dans l'entité Region.	47
Figure 5.5 : Opération de regroupement.	48
Figure 5.6 : Schéma « star cluster » créé à partir du schéma entité-relation.	50
Figure 5.7 : Modèle conceptuel de l'étoile COMMANDE selon Moody et Kortink.	51
Figure 5.8 : Modèle conceptuel de l'étoile DET COMMANDE selon Moody et Kortink.	51
Figure 5.9 : Modèle conceptuel de l'étoile TRANSFERT selon Moody et Kortink.	52
Figure 5.10 : Modèle conceptuel de Étoile Stock selon Moody et Kortink.	52
Figure 5.11 : Modèle conceptuel de l'étoile Approvisionnement selon Moody et Kortink.	53
Figure 5.12 : Modèle conceptuel de l'étoile Det Appro selon Moody et Kortink.	53
Figure 5.13 : Tables de fait du modèle multidimensionnel selon Moody et Kortink.	54
Figure 5.14 : Dimensions du modèle multidimensionnel construit selon Moody et Kortink.	54
Figure 5.15 : Modèle multidimensionnel du fait HU_COMMANDE selon Hüsemann.	63
Figure 5.16 : Modèle multidimensionnel du fait HU_LI_COMMANDE selon Hüsemann.	63
Figure 5.17 : Modèle multidimensionnel du fait HU_TRANSFERT selon Hüsemann.	64
Figure 5.18 : Modèle multidimensionnel du fait HU_INVENTAIRE selon Hüsemann.	64
Figure 5.19 : Modèle multidimensionnel du fait HU_APPRO selon Hüsemann.	64
Figure 5.20 : Modèle multidimensionnel du fait HU_DET_APPRO selon Hüsemann.	65
Figure 5.21 : Représentation graphique de la dimension COMMANDE construite selon Hüsemann.	
La notation graphique a été décrite dans le paragraphe 5.2.1.3.	68
Figure 5.22 : Représentation graphique de la dimension PRODUIT construite selon Hüsemann.	69
Figure 5.23 : Représentation graphique de la dimension TRANSFERT construite selon Hüsemann.	69
Figure 5.24 : Représentation graphique de la dimension ENTREPOT construite selon Hüsemann.	69
Figure 5.25 : Représentation graphique de la dimension PERIODE construite selon Hüsemann.	70
Figure 5.26 : Tables des faits du modèle multidimensionnel selon Hüsemann.	72
Figure 5.27 : Tables des dimensions du modèle multidimensionnel selon Hüsemann.	73
Figure 5.28: Modèle multidimensionnel de la gestion des commandes de Tarigomi selon Kimball.	77
Figure 5.29 : Modèle multidimensionnel de la gestion des lignes de commande de Tarigomi.	77
Figure 5.30 : Modèle multidimensionnel de la gestion de l'inventaire de Tarigomi selon Kimball.	77
Figure 5.31 : Modèle multidimensionnel de la gestion des approvisionnements de Tarigomi selon Kimball.	78
Figure 5.32 : Modèle multidimensionnel de la gestion des détails des approvisionnements de Tarigomi selon Kimball.	78
Figure 5.33 : Modèle multidimensionnel de la gestion des transferts de Tarigomi selon Kimball.	79

Figure 5.34 : Les tables de fait de Tarigomi selon Kimball.	79
Figure 5.35 : Les tables des dimensions et leurs attributs pour les modèles multidimensionnels de Tarigomi selon Kimball.	80
Figure 5.36 : Modèle entité-relation de Tarigomi dans lequel on a enlevé toutes les entités associatives et on les a remplacées par des relations de N à M.	83
Figure 5.37 : Regroupement des entités qui sont aux extrémités des relations N à M.	83
Figure 5.38 : Dimensions de Tarigomi.	85
Figure 5.39 : Version initiale des faits pour les requêtes Q1 à Q10. Les attributs des tables de fait sont les mesures du Tableau 5.10.	86
Figure 5.40 : Les attributs des faits indiqués dans la Figure 5.39 sont modifiés pour harmoniser la granularité et ne garder que des mesures additives.	87
Figure 5.41 : Les faits de la Figure 5.40 ont été fusionnés.	88
Figure 5.42 : Version finale des faits pour le cas de Tarigomi.	89
Figure 5.43 : Modèle multidimensionnel des commandes mensuelles.	90
Figure 5.44 : Modèle multidimensionnel des commandes.	90
Figure 5.45 : Modèle multidimensionnel des lignes de commande.	91
Figure 5.46 : Modèle multidimensionnel de l'inventaire.	91

1. Introduction

Les entreprises, qui font face à une augmentation de la concurrence et une mondialisation des intervenants, doivent de plus en plus avoir recours aux informations qu'elles détiennent afin de prendre les décisions les mieux éclairées. Pour utiliser efficacement l'actif que représente l'accumulation des données mises à disposition par les systèmes d'informations qui prennent en charge les transactions et les opérations, les entreprises ont, de plus en plus, recours à des entrepôts de données qui permettent d'intégrer l'ensemble des données de l'organisation. En effet, plusieurs auteurs (Meta Group, 1999 ; Barquin, 1997) considèrent que la technologie d'entrepôt de donnée (data warehouse (DW)) est devenue la manière la plus utilisée pour obtenir des informations et pour combler les besoins d'analyses de la plus grande majorité des organisations.

Selon Kimball (1996), la grande difficulté de l'utilisation d'un entrepôt de données réside dans la nature des questions que les managers doivent résoudre pour prendre des décisions et pour lesquelles les bases de données traditionnelles ne sont pas conçues

Pour Devlin (1997), un entrepôt de données est simple, complet et les données sont cohérentes. Il est obtenu à partir d'un grand nombre de sources variées. De plus, son accès est simple et compréhensible de manière à ce que l'utilisateur final le comprenne et puisse l'utiliser dans un contexte d'affaire.

Inmon (1993), qui est considéré comme l'un des pionniers des entrepôts de données et qui a publié de nombreux articles et livres sur le sujet, définit quant à lui un entrepôt de données comme un ensemble de données orienté en fonction d'un sujet, intégré, variable dans le temps, non volatile et qui est utilisé pour supporter la prise de décision des gestionnaires. Nous verrons au chapitre 2 une définition plus détaillée de l'entrepôt de données.

À la fin des années 90, la grande majorité des entreprises de Fortune 1000 avaient démarré des projets de data warehouse (Barquin, 1997). D'après une étude du Meta Group (1999), les organisations qui ne sont pas actuellement en train d'entretenir et de maintenir des entrepôts de données en développent ou planifient d'implanter ce genre de bases de données. Les mêmes auteurs relient la nécessité de disposer d'un entrepôt de données à un certain nombre de facteurs (Meta Group, 1999) :

- Les bénéfices tacites et les effets combinés du succès des entrepôts de données et des possibilités des réalisations qui découlent de leur mise en place,

- L'utilisation de plus en plus grande des ERP qui permettent de disposer de beaucoup de données, tout en ne fournissant pas des moyens d'analyse et des rapports satisfaisants,
- L'apparition d'outils d'entrepôt de données de plus en plus complets et faciles à utiliser,
- Le besoin toujours plus important d'informations et de résultats d'analyse,
- La perception que la gestion compétente d'information est indispensable pour faire un management efficace.

Les entrepôts de données sont des outils d'aide à la prise de décision qui permettent de d'utiliser la grande quantité de données générées par les ERP. Ils offrent la possibilité de supporter la gestion des relations avec les clients (CRM), celle des chaînes d'approvisionnement (SCM) et de subvenir aux besoins du commerce électronique. L'augmentation de la vitesse des affaires et le niveau d'intégration des systèmes d'affaire se traduisent par une réduction du temps nécessaire pour la collecte et l'analyse des données. De plus, les décisions que doivent prendre les gestionnaires sont basées de plus en plus sur des informations globales (Meta Group, 1999).

La construction des entrepôts de données a abouti dans un certain nombre de cas à des échecs importants. En effet, au cours d'une étude réalisée par le Meta Group (1999), seulement 59% des répondants ont indiqué que les projets d'implantation d'un entrepôt de données ont été un succès alors que 41% ont estimé que ces projets ont été des échecs (Figure 1.1).

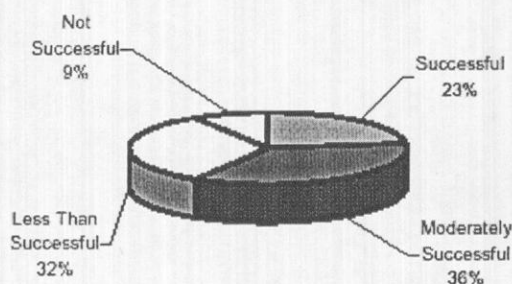


Figure 1.1: Taux de succès des entrepôts de données (Meta Group, 1999).

Le TDWI (The Data Warehouse Institute) est moins négatif et indique que 25% des projets de construction d'un entrepôt de données ont échoué. Ces chiffres relativement élevés ont amené plusieurs auteurs à évaluer des raisons d'échec et à proposer des solutions pour éviter que les projets d'entrepôt de données ne conduisent à des échecs (Fosdick, 1997 ; Mimno, 1999 ; Few, 1999 ; Kosar, 1997 ; Barquin et al, 1997).

Parmi les erreurs à éviter pour concevoir, implanter et gérer efficacement un entrepôt de données, Barquin et al. (1997) relèvent la croyance que ce genre de base de données peut être conçu et administré de la même manière que les bases de données traditionnelles.

Comme on peut le voir sur la Figure 1.2, un entrepôt de données est composé des éléments suivants :

- Gestion :
 - Extraction des données,
 - Transformation des données,
 - Chargement,
 - Scheduler,
 - Gestion des métadonnées
- Organisation :
 - Système de gestion de base de données,
- Exploitation :
 - Outils de traitement analytique interactif (On Line Analytical Processing ou OLAP),
 - Système d'information d'une entreprise (Enterprise Information System ou EIS),
 - Outils de requête,
 - Outils de visualisation des données,
 - Outils de forage des données.

La visualisation et l'analyse des données contenues dans un entrepôt de données sont effectuées avec des outils OLAP qui permettent une analyse multidimensionnelle des informations. Ces outils présentent l'avantage de permettre aux utilisateurs de manipuler simplement et efficacement les données en vue de les aider à prendre des décisions. On verra au chapitre 2 la définition de ces outils et de leurs différentes variantes.

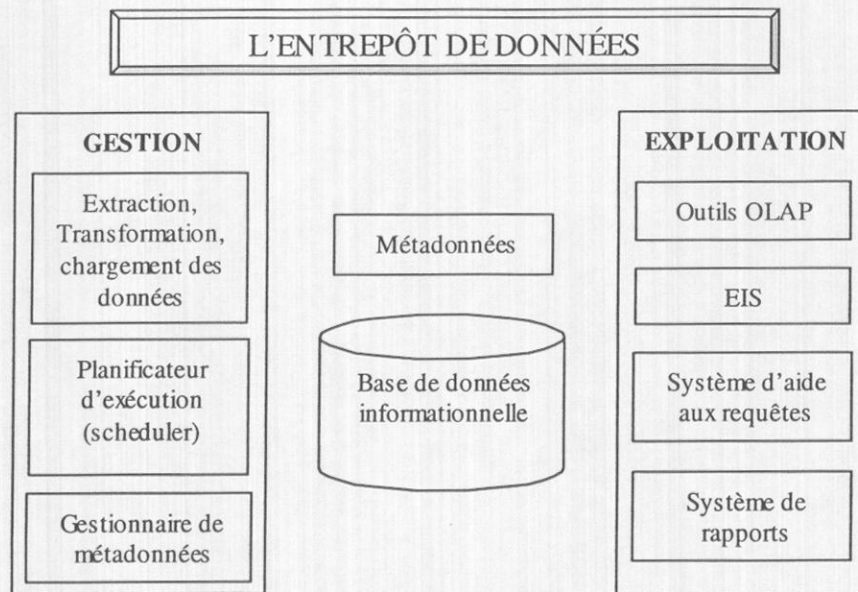


Figure 1.2: Éléments d'un entrepôt de données.

Les données qui sont utilisées par les outils OLAP sont en général modélisées avec le modèle multidimensionnel (Moody et al. (2000), Wu et Buchmann (1997)). Cette technique de modélisation est décrite dans le chapitre 2.

La démarche de conception des modèles multidimensionnels n'est toutefois pas complètement maîtrisée et ni formalisée (Hüsemann et al., 2000). Vassiliadis (1998) de son côté, présente une liste de raisons d'échec des projets d'entrepôt de données et remarque que parmi les facteurs liés au design, il n'existe pas, entre autres, de standard pour la méthodologie de conception des modèles multi dimensionnels.

L'étude a pour but d'étudier différentes démarches de conception de modèles multidimensionnels utilisés pour des entrepôts de données et de comparer les modèles construits à partir des différentes démarches.

L'objectif de recherche est le suivant :

Comparer des démarches de conception de modèles multidimensionnels utilisés pour des entrepôts de données.

Importance et bénéfices de l'étude

La méthode de conception des modèles multidimensionnels fait l'objet d'une grande polémique dans la communauté des personnes qui fabriquent et qui implantent des entrepôts de données (Devlin, 1997). La question de savoir quelle démarche de conception utiliser et comment construire un entrepôt de données est d'une grande importance.

Notre étude permettra de guider le choix de la démarche à suivre pour éviter une partie des échecs des projets des entrepôts de données. La démarche pourra être choisie en fonction de la disponibilité d'un modèle des données de l'entreprise, de la qualité de ce modèle, de la possibilité et de l'opportunité de faire appel à un groupe d'utilisateurs pour mener à bien le projet d'entrepôt de données et de la disponibilité d'une liste de requêtes que les utilisateurs utiliseront pour effectuer leurs analyses.

Pour illustrer l'emploi de chacune des démarches que nous aurons choisies, nous utiliserons des données fictives d'une entreprise de distribution de matériel électronique. Nous construirons les modèles en suivant les démarches et nous les analyserons au moyen du logiciel Discoverer. Ce logiciel est brièvement décrit en annexe.

On trouvera dans le deuxième chapitre une revue de la littérature qui est consacrée au domaine des entrepôts de données et une étude des différents thèmes reliés à ce sujet. Au cours du troisième chapitre, nous présenterons la méthodologie de notre étude. Le quatrième chapitre sera consacré à la présentation du modèle de données d'une entreprise fictive grossiste en matériel informatique qui s'appelle Tarigomi et d'une liste de requêtes qui seront utilisées pour analyser ou pour construire les modèles. Au cours de ce chapitre, nous présenterons le modèle de Tarigomi tel qu'il a été utilisé dans ce travail. Le cinquième chapitre présentera quatre démarches différentes de conception des entrepôts de données et l'application de ces méthodes de modélisation aux données de Tarigomi. Notre travail nous conduira à comparer et à critiquer les différents modèles, ce qui constituera le sixième chapitre. Nous présenterons ensuite les conclusions de notre étude. On trouvera en annexe une présentation d'un ensemble de données simplifiées, une liste des requêtes qui ont été appliquées aux différents modèles, les requêtes SQL pour la création des modèles qui ont été conçu en suivant les quatre démarches et les résultats de l'application de Discoverer aux modèles d'entrepôt de données.

2. Revue de la littérature

2.1. Définition d'un entrepôt de données

Un entrepôt de données est un outil qui aide la prise de décision en faisant partie d'un ensemble de processus qui vont de l'origine des données, de leur extraction, de leur transformation à partir des bases de données opérationnelles, de leur gestion et de leur utilisation (Ballard et al., 1998). C'est en effet plus qu'un produit que l'on peut acheter et utiliser tel quel pour interroger les bases de données de l'entreprise et obtenir des réponses pour améliorer la prise de décision.

Ainsi, la mise en place d'un entrepôt de données concerne la conception de processus et d'outils pour gérer et fournir des informations de qualité, complètes, à jour et compréhensibles.

Les entreprises entreprennent de mettre en place un entrepôt de données non seulement pour accéder facilement à leurs données mais aussi pour disposer d'un moyen efficace et sûr d'entreposage d'informations de qualité. De plus en plus, les organisations ont à leur disposition une très vaste quantité de données mais il leur est très difficile d'en extraire des informations de qualité du fait en partie de la grande diversité de leur origine et de leur incompatibilité. Elles doivent mettre en place et maintenir parfois plusieurs centaines de programmes pour interroger les différentes bases de données et consolider les données extraites. La mise en place d'un entrepôt de données peut être une réponse à cette difficulté toujours croissante de disposer d'informations qui vont aider la prise de décision.

Pour Inmon (2000), un entrepôt de données est le centre d'une architecture de systèmes d'information et constitue une plate-forme solide qui fournit des données intégrées et historiques pour faire des analyses. Un entrepôt de données est donc une collection de données orientée en fonction d'un sujet, intégrées, dépendantes du temps et permanentes (Figure 2.1).

Par la suite nous allons développer le sens de ces quatre caractéristiques des données de l'entrepôt de données.



un entrepôt de données est un ensemble de données:

- orientées en fonction d'un sujet,
- intégrées,
- dépendantes du temps,
- permanentes.

Figure 2.1: définition d'un entrepôt de données selon Inmon (2000).

Orientation en fonction d'un sujet

La première particularité d'un entrepôt de données réside dans le fait que les données sont organisées en fonction d'un sujet plutôt qu'en fonction d'un processus ou d'une entité fonctionnelle d'une entreprise. Classiquement dans le cas d'une compagnie financière, par exemple, les données opérationnelles sont conçues à partir de processus et de fonctions telles que les prêts, les épargnes, les cartes de crédit ou les investissements. Dans le cas d'un entrepôt de données, les données sont organisées au contraire en fonction de sujets tels que les clients, les vendeurs, les produits et les activités.

Le contenu des bases de données opérationnelles et des entrepôts de données sera différent. En effet, les entrepôts de données ne contiendront que des données utiles à l'analyse, alors que les bases de données opérationnelles peuvent contenir des données nécessaires pour les processus de transaction mais qui ne sont pas utiles à l'analyse.

De plus, les données opérationnelles concernent souvent des relations entre deux ou plusieurs tables en fonction de règles d'affaire alors que celles contenues dans un entrepôt de données ne concernent que rarement ce type de relations.

Intégration

Les données d'un entrepôt de données proviennent de différentes sources et sont intégrées de manière à assurer leur cohérence, tant au niveau des conventions de noms, de la mesure de variables, de la structure d'encodage que de leurs attributs. Ainsi, par exemple, les systèmes opérationnels utiliseront des définitions différentes pour le genre des personnes. Le système A représentera le genre par les lettres « m » et « f », le système B par les chiffres « 1 » et « 0 », le système C par les mêmes chiffres mais inversés, le système D par un énoncé complet « masculin » et « féminin » et le système E par les lettres « Y » et « X ». (Figure 2.2). Lorsque les données seront insérées dans l'entrepôt de données, elles seront transformées pour qu'il n'y ait qu'une seule représentation du genre des personnes. De la même manière,

la représentation des dates peut être différente pour chaque système opérationnel et doit être transformée pour qu'il ne reste qu'une seule représentation dans l'entrepôt de données.

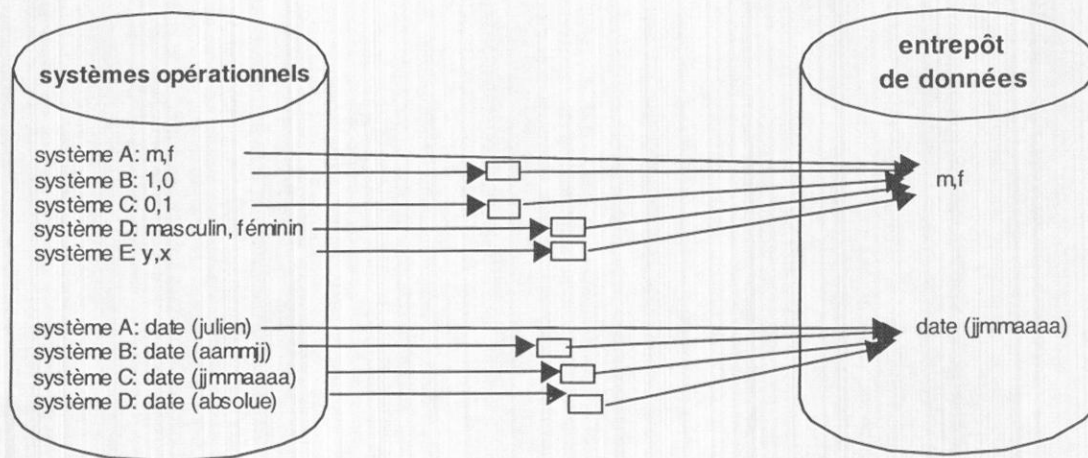


Figure 2.2 : les données de l'entrepôt de données sont intégrées à partir de différentes sources.

Dépendance temporelle

Toutes les données d'un entrepôt de données dépendent d'un certain instant et doivent être justes en tout temps alors que dans une base de données opérationnelle, elles ne le sont qu'au moment de l'accès (Figure 2.3).

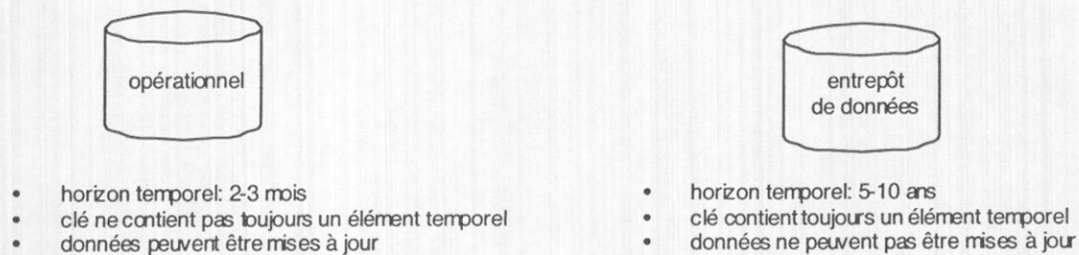


Figure 2.3: les données de l'entrepôt de données dépendent du temps.

La dépendance temporelle des données se manifeste de plusieurs manières. Tout d'abord, l'horizon temporel d'un entrepôt de données est de l'ordre de cinq à dix ans alors qu'il est de l'ordre de deux ou trois mois pour une base de données opérationnelle. De ce fait, un entrepôt de données est composé d'un grand nombre de données, ce qui peut nuire aux performances d'une application opérationnelle. D'autre part, le temps dans un entrepôt de données fait partie de la clé primaire des données qui contiennent

toujours un élément temporel tel que la date, l'année, le semestre ou le mois. Finalement, la dépendance temporelle se manifeste par le fait que les données d'un entrepôt de données ne sont plus mises à jour lorsqu'elles ont été correctement enregistrées.

Permanence

Les données d'un entrepôt de données sont en plus permanentes, ce qui signifie qu'elles ne sont pas mises à jour une fois qu'elles ont été chargées dans l'entrepôt de données. Dans le cas d'une base de données opérationnelle, les données sont modifiées presque à chaque intervention par des instructions d'insertion, d'effacement ou de modification (Figure 2.4).

La permanence des données a pour conséquence qu'il n'est pas nécessaire, dans le cas d'un entrepôt de données, d'être très attentif aux anomalies de mise à jour, alors que dans le cas d'une base de données opérationnelle, ces anomalies ne peuvent pas être tolérées. Cela signifie qu'il est possible de dénormaliser les données afin d'améliorer les performances de l'entrepôt. De plus, les systèmes de gestion des bases de données opérationnelles sont très complexes pour éviter les problèmes de cohérence et de modification simultanée par plusieurs usagers alors que dans le cas d'un entrepôt de données, ces systèmes ne font que consulter les données et sont beaucoup plus simples.

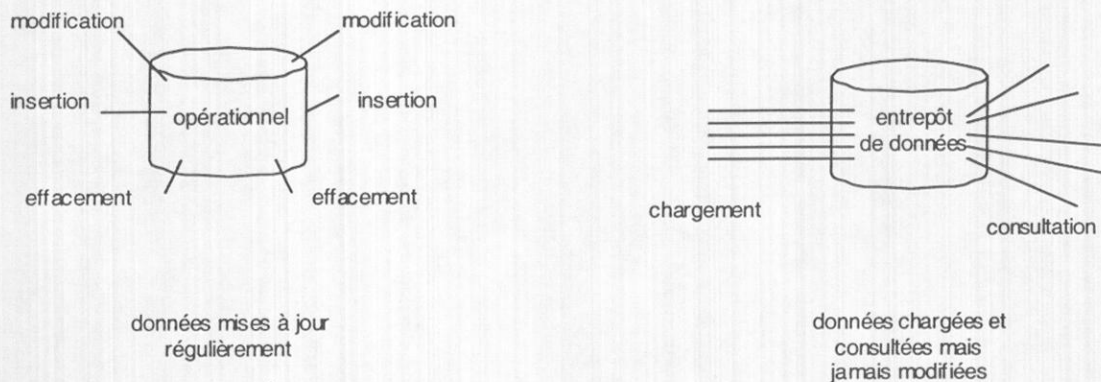


Figure 2.4: les données de l'entrepôt de données sont permanentes.

En conclusion, les différences entre les entrepôts de données et les bases de données opérationnelles sont très grandes et impliquent de grandes différences de conception et d'opération.

À première vue, il peut sembler qu'il existe une grande redondance entre les données d'un entrepôt de données et celles des bases de données opérationnelles puisque que ces dernières servent à alimenter les entrepôts de données. Toutefois, il faut remarquer que les données de l'entrepôt sont très différentes

dans la mesure où elles ont été filtrées et que seules celles qui sont utiles à l'analyse sont transférées. De plus, l'horizon temporel est très différent et l'entrepôt de données contient des agrégations qui n'existent pas dans une base de données opérationnelle. Finalement, les données d'un entrepôt de données ont subi des transformations au cours de l'intégration.

Pour Chaudhuri et al., 1997, les bases de données opérationnelles permettent l'automatisation de processus de routine qui correspondent aux opérations quotidiennes d'une organisation. Ces opérations sont des tâches structurées et répétitives, courtes et correspondent à des transactions isolées.

La taille des bases de données est de quelques dizaines jusqu'à quelques gigabytes. La cohérence et la capacité à retrouver des données sont critiques. Le moyen le plus important pour améliorer la performance est de maximiser l'efficacité des transactions. Les bases de données opérationnelles sont conçues pour refléter la sémantique d'opérations connues et pour minimiser les conflits de concurrence.

Dans le cas d'un entrepôt de données, les données historiques, résumées et consolidées sont de la plus haute importance. Le travail d'analyse fait appel à une grande quantité de requêtes essentiellement ad hoc et complexes qui nécessitent d'accéder à des millions d'enregistrements et qui effectuent une grande quantité d'interrogations, de jointures et d'agrégations. L'efficacité des requêtes et le temps de réponse sont plus importants que l'efficacité des transactions.

La taille des entrepôts de données est plus grande que celle des bases de données opérationnelles et peut atteindre le Terabyte.

2.2. Techniques d'analyse

Comme on l'a dit plus haut, un entrepôt de données est un moyen conçu pour analyser les données et fournir des informations de qualité, complètes, à jour et compréhensibles. Parmi les différentes techniques d'analyse pour interroger un entrepôt de données, Ballard et al. (1998) citent l'utilisation de requêtes et de rapports, l'analyse multidimensionnelle et le forage des données (data mining). Les techniques d'analyse peuvent avoir un impact sur le type de modèle de données et sur son contenu. Par exemple, si l'analyse se limite à la mise en place de requêtes et à la production de rapports, le modèle de données qui structure les données de manière normalisée sera probablement plus efficace et offrira l'accès le plus facile et le plus rapide aux données. Pour ce genre d'analyse, le modèle entité-relation avec une structure normalisée des données sera donc probablement la plus appropriée.

Par contre, si l'analyse est multidimensionnelle, un modèle de données multidimensionnel sera plus approprié. Ce type d'analyse requiert que la structure du modèle de données permette un accès facile et rapide aux données sur la base de la combinaison d'un certain nombre de dimensions. Par exemple, on peut vouloir connaître la quantité d'un produit particulier vendu à une date particulière, dans un magasin particulier et dans un intervalle particulier de prix. Ensuite, lors d'une analyse plus poussée, on peut vouloir connaître combien de magasins ont vendu un produit particulier, dans un intervalle donné de prix et lors d'un jour particulier. Ces deux questions nécessitent des informations semblables mais l'analyse est effectuée une fois dans une perspective de produit alors que l'autre fois elle est réalisée dans une perspective de magasin. L'analyse multidimensionnelle nécessite un modèle de données où les données sont organisées de telle manière qu'elles puissent être analysées selon plusieurs perspectives (dimensions). Étant donné qu'un grand nombre de dimensions sera utilisé, il faut que le modèle permette un accès rapide aux données. Si on utilise un modèle très normalisé, il sera nécessaire d'effectuer un grand nombre de jointures entre les tables qui possèdent les informations et les performances seront faibles.

2.2.1. Requêtes et rapports

L'analyse par des requêtes et des rapports est effectuée en posant une question, en retrouvant les données pertinentes de l'entrepôt de données, en les transformant pour convenir au contexte et en les représentant dans un format lisible. Elle est conduite par un analyste qui doit poser ce type de question et recevoir les rapports.

Habituellement, les requêtes utilisent deux dimensions ou deux facteurs en même temps. Par exemple, une question pourrait être : « Quelle est la quantité d'un produit qui a été vendue cette semaine » ? Par la suite, d'autres requêtes pourraient être effectuées pour déterminer la quantité d'un produit vendue par un magasin particulier (Figure 2.5). La définition de la requête est le processus d'établissement d'une question ou d'une hypothèse d'affaire et de traduction dans un format de requête qui pourra être utilisé dans un outil d'aide à la décision. Lorsque la requête est effectuée, l'outil génère les commandes dans un langage approprié pour accéder et extraire les données nécessaires qui sont ensuite retournées dans un ensemble de données réponse. L'analyste effectue ensuite les calculs et les manipulations sur cet ensemble pour obtenir les résultats attendus. Ces résultats sont finalement mis sous la forme de graphique et de rapport afin d'être compréhensibles par l'utilisateur final. Souvent, les résultats conduisent les analystes à formuler d'autres requêtes pour clarifier ou pour détailler la réponse. Le processus de mise en

place et d'exécution de requête est donc poursuivi jusqu'à ce que l'utilisateur final soit satisfait des résultats.

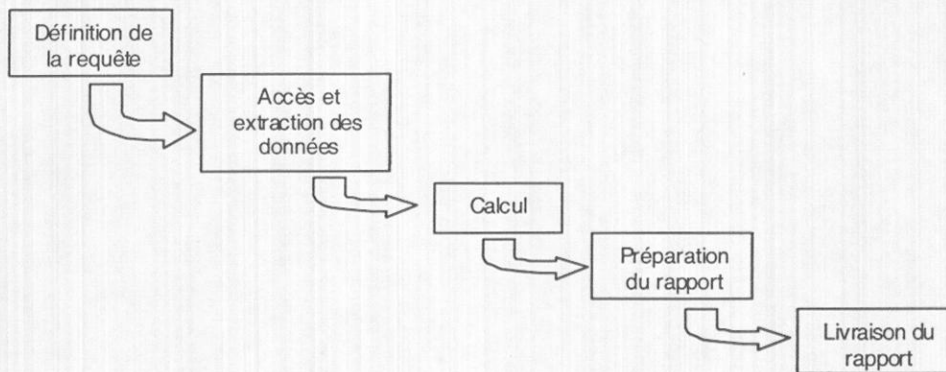


Figure 2.5: Le processus de requête et rapport débute par la définition de la requête et se termine par la livraison du rapport.

2.2.2. Analyse multidimensionnelle

L'analyse multidimensionnelle permet d'étendre l'analyse par requête et rapport et d'éviter de soumettre de multiples questions en structurant les données pour permettre une réponse rapide aux questions les plus fréquentes. Par exemple, les données seront structurées pour inclure la réponse à la question « Quelle est la quantité vendue d'un produit particulier, par un vendeur particulier, dans une région particulière lors d'une journée particulière ? ». Chaque partie de cette question est appelée une dimension. Si l'analyste pose ensuite la question « Quelle est la quantité vendue d'un produit particulier, par un vendeur particulier ? », la réponse peut être obtenue facilement si les données ont été organisées en fonction de catégories qui correspondent aux dimensions. En calculant à l'avance les réponses dans un contexte large, les résultats peuvent être obtenus facilement parce qu'il n'est pas nécessaire de les recalculer à chaque fois.

L'analyse multidimensionnelle donne la possibilité à l'utilisateur d'étudier un grand nombre de facteurs interdépendants qui participent à un problème d'affaire et d'observer les données dans un contexte complexe d'inter relations.

Les utilisateurs finaux sont intéressés à explorer les données à différents niveaux de détail qui sont déterminés de façon dynamique. Les relations complexes peuvent être analysées au cours d'un processus itératif qui inclut la possibilité de naviguer vers un plus grand niveau de détails ou au contraire vers un

plus grand regroupement des données (voir Figure 2.6). Sur cet exemple, l'utilisateur peut débuter par un client, regrouper les résultats par pays, par région ou par continent pour atteindre finalement le volume total des ventes.

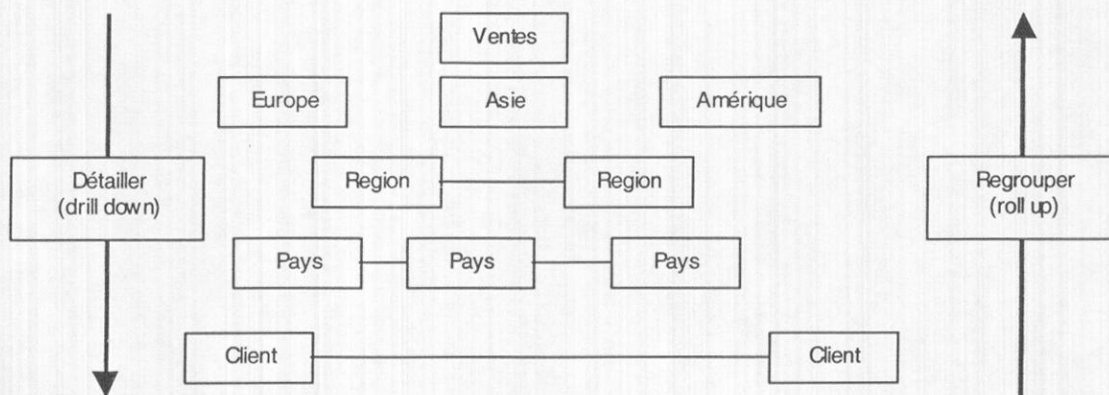


Figure 2.6: Navigation à l'intérieur d'une dimension (drill down ou roll up) pour obtenir plus de détails ou au contraire regrouper les données.

2.2.3. Forage des données

Le forage des données (data mining) est une technique de découverte qui est très différente des deux techniques précédentes. Ainsi, au lieu d'utiliser des questions et plutôt que de s'appuyer sur des hypothèses pour bâtir des requêtes, le forage des données utilise des algorithmes particuliers pour effectuer des analyses et chercher des réponses à des questions qui n'avaient pas particulièrement été posées. Par exemple, le forage des données peut aider à trouver des relations entre certains éléments de donnée, à proposer des groupements entre des données ou identifier des patterns qui font usage de certains ensembles de données.

Le forage de données est utilisé en majeure partie pour l'analyse statistique et la découverte de la connaissance. L'analyse statistique permet de détecter des patterns inhabituels et utilise des techniques de modélisation statistique ou mathématique, pour expliquer ces patterns. Parmi les techniques d'analyse statistique utilisées en forage des données, on peut mentionner l'analyse linéaire et non linéaire, la régression, l'analyse multi variée ou l'analyse des séries temporelles.

Le forage des données est piloté par les données et permet de découvrir des relations insoupçonnées que l'analyse par requête et rapport ou l'analyse multidimensionnelle ne permettent pas d'identifier.

Le genre d'analyse qui sera conduite sur les données peut déterminer le type de modèle et le contenu des modèles de données. Ainsi, le modèle devra contenir un grand degré de regroupement et une gestion efficace des métadonnées pour convenir à l'analyse par requête et rapport et l'analyse multidimensionnelle. De plus, le modèle devra contenir des moyens pour détailler (drill up) ou regrouper (roll up) les données puisque ce mécanisme est très utilisé au cours de l'analyse multi dimensionnelle. En fin, les données devront être très détaillées lorsqu'elles seront analysées par forage des données.

Il faut remarquer que le design conceptuel est indépendant du design logique et l'utilisation d'un modèle conceptuel particulier n'implique pas qu'il soit utilisé pour le design logique ou physique (Wu et Buchmann, 1997).

2.3. Architecture des données

Comme on l'a vu précédemment, un entrepôt de données est par définition une collection orientée en fonction d'un sujet, intégrées, dépendantes du temps et permanentes. Un des concepts de base d'un entrepôt de données est de nettoyer, filtrer, transformer, regrouper et agréger les données et de les stocker ensuite dans un entrepôt qui permette un accès facile, une analyse adaptée aux besoins et une bonne compréhension de la part des usagers. Mais en premier lieu, la structure du modèle et l'architecture des données doivent être définies.

Dans un entrepôt de données, on peut distinguer trois types de données (Ballard et al., 1998 ; Devlin, 1997) :

- Les données opérationnelles,
- Les données réconciliées,
- Les données dérivées.

L'entrepôt de données sera configuré à partir de ces trois types en donnant plus ou moins d'importance à un type en fonction des données opérationnelles, du type de business et du nombre d'utilisateurs qui doivent avoir accès à l'entrepôt de données. Par exemple, dans le cas d'une compagnie de géophysique où toutes les applications tirent les données d'une même source et où la quantité de données est considérable, il est préférable d'adopter une structure avec une seule couche, la couche opérationnelle.

2.3.1. Données opérationnelles

Les données de la couche opérationnelle représentent l'état actuel des affaires. Elles sont utilisées par les applications opérationnelles pour faire fonctionner l'entreprise et sont constamment mises à jour au fur et à mesure que des transactions sont effectuées. Les données opérationnelles sont d'un grand niveau de détail, ce qui signifie une grande granularité.

Pour utiliser les données opérationnelles dans un entrepôt de données, il faut tout d'abord les nettoyer pour assurer leur qualité et les filtrer. En effet, les bases de données opérationnelles contiennent des informations qui sont nécessaires pour les opérations mais qui n'ont pas d'utilité pour la prise de décision. De plus, comme ces données proviennent de sources multiples, leur représentation et leur signification peuvent différer d'une source à l'autre.

2.3.2. Données réconciliées

Les données réconciliées sont détaillées et temporelles et constituent la seule source pour toutes les applications d'aide à la prise de décision. Les données réconciliées ne sont en général pas prévues pour être interrogées directement par l'utilisateur final. Ces données ont été nettoyées et transformées pour assurer leur qualité et leur cohérence.

2.3.3. Données dérivées

Les données dérivées ont été créées par regroupement, calcul de moyenne ou agrégation des données réconciliées. Ces données peuvent être soit détaillées ou regroupées suivant les besoins d'analyse. Elles sont utilisées pour l'analyse et la prise de décision.

2.3.4. Rôle des métadonnées

Rappelons que les métadonnées sont des données qui décrivent les propriétés ou les caractéristiques d'autres données. Dans le cas d'un entrepôt de données, un ensemble de métadonnées est associé à chaque couche de données. Les métadonnées opérationnelles décrivent les données disponibles dans les différents systèmes opérationnels de l'entreprise et qui seront utilisées pour peupler l'entrepôt de données. Les métadonnées de l'entrepôt de données de l'entreprise dépendent du modèle de données de l'entreprise. Elles décrivent les données réconciliées et les transformations nécessaires pour passer des données opérationnelles aux données réconciliées.

Les métadonnées dérivées décrivent les règles de transformation des données réconciliées pour les regrouper et agréger en fonction des besoins de l'analyse et de la prise de décision.

2.4. Modélisation multidimensionnelle

La modélisation multidimensionnelle est une technique qui est utilisée pour modéliser les données sous la forme d'un ensemble de mesures décrites par des aspects communs des affaires. Cette modélisation est particulièrement utile pour naviguer dans les différents niveaux de données, que ce soit pour rechercher des détails ou au contraire pour les regrouper. Elle convient spécialement bien à l'analyse multidimensionnelle. Le modèle multidimensionnel est aussi appelé modèle étoile dans la mesure où sa représentation graphique ressemble à une étoile composée d'une table centrale et un ensemble de petites tables représentées par des rayons autour de la table centrale. Ce modèle prendra le nom de modèle constellation (« snowflake ») lorsque les rayons de l'étoile seront décomposés en troisième forme normale.

2.4.1. Modèle étoile

Le modèle étoile est composé d'une table centrale dominante, la table de faits, qui est jointe à une série de tables de caractérisation appelées dimensions. Contrairement à la modélisation entité-relation, la modélisation multidimensionnelle est très asymétrique. Dans un modèle étoile pur, cette table est la seule à être jointe à plusieurs autres tables. Les autres tables ne sont pas reliées entre elles (Figure 2.7).

Dans la Figure 2.7, nous modélisons un modèle simple d'organisation qui reçoit des commandes de produits de la part de clients et qui analyse les performances des commandes. La table de fait contient la quantité expédiée par mois, par client et par produit, ce qui constitue le niveau de détail de la table de fait. En général, la taille de la table de faits est très importante.

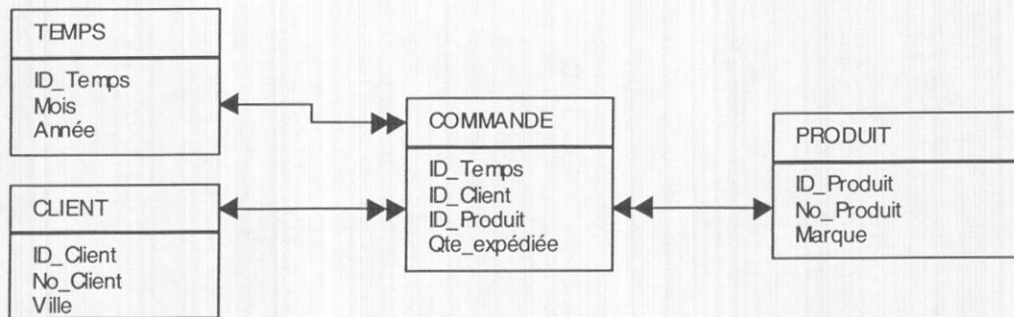


Figure 2.7: Le modèle étoile.

Dans notre exemple, les dimensions contiennent les éléments suivants :

Temps		Client		Produit	
Année	Mois	Ville	No Client	Marque	No Produit
1996	Janvier	Montréal	1001	Panasonic	101
	Février		1002		102
	Mars	Laval	1011	Sony	103
	avril		1012		104
		Longueuil	1021	A.D.S	105
			1022		

Table 2.1 : Données de l'exemple.

Une autre manière de représenter un modèle multidimensionnel est de dessiner un cube. Chaque dimension constitue l'arrête du cube et les cellules représentent les différentes valeurs que les mesures pourront prendre en fonction des dimensions. En général, le nombre de dimensions est supérieur à trois et il faudrait pouvoir représenter un hypercube. L'hypercube est représenté sous la forme d'un cube dont les dimensions qui ne sont pas représentées sont regroupées (Figure 2.8).

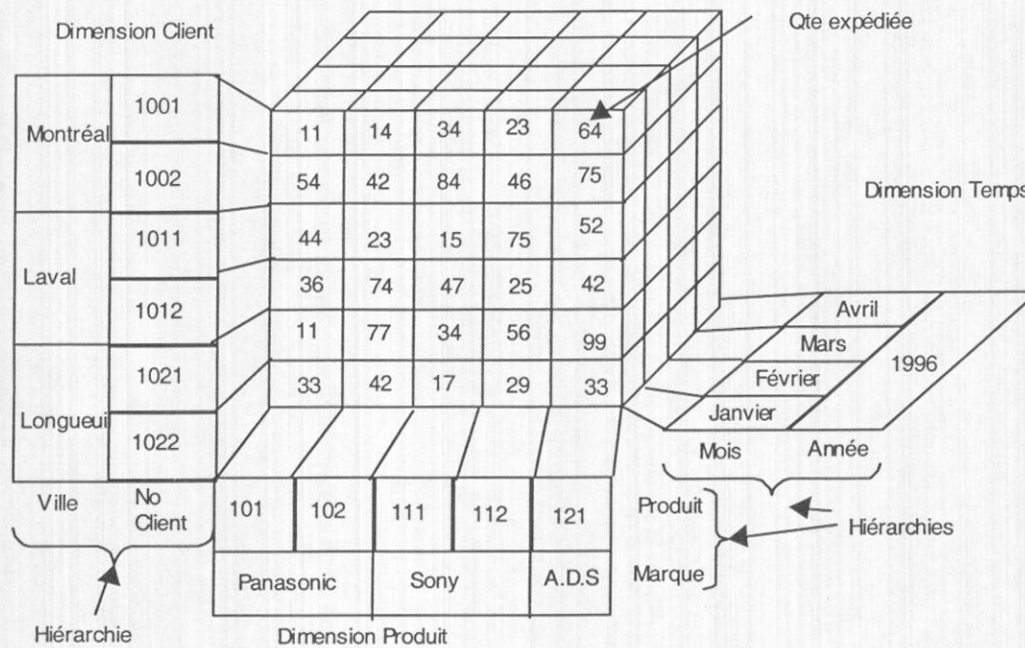


Figure 2.8: La représentation du modèle multidimensionnel par un cube.

Ainsi, les éléments de base de la modélisation multidimensionnelle sont les suivants :

- Faits,
- Dimensions,
- Mesures.

2.4.1.1. ____ Faits

Un fait est une collection de données formée des mesures et des informations qui caractérisent le domaine d'affaires. Chaque fait représente un élément d'affaire, une transaction ou un événement qui peut être utilisé pour analyser un processus d'affaire. Dans l'exemple précédent, le fait contient la quantité expédiée par produit, par mois et par client.

2.4.1.2. ____ Dimension

Les dimensions sont les caractérisations et les descriptions textuelles du domaine d'affaires. Chaque caractérisation ou chaque description permet de décrire un membre d'une dimension. Dans l'exemple précédent, les commandes sont caractérisées par les dimensions Temps, Produit et Client. La dimension temporelle réunit toutes les informations qui concernent la date des commandes. La dimension Produit décrit ce qui a été commandé et la dimension Client caractérise la personne ou l'organisation par laquelle la commande a été passée. La table de dimension peut être composée de plusieurs attributs textuels et discrets qui peuvent être utilisés pour placer des contraintes sur l'ensemble des réponses. Elles sont en général très dénormalisées et contiennent une ou plusieurs hiérarchies..

2.4.1.3. ____ Mesures

Les mesures sont prises à l'intersection de toutes les dimensions et ne sont pas connues à l'avance mais sont enregistrées au fur et à mesure que des transactions sont effectuées. Les mesures sont généralement des nombres additifs dont les valeurs sont continues. Parfois, les mesures sont semi-additives ou ne sont pas additives du tout. Les mesures sont semi-additives lorsqu'elles ne peuvent être additionnées que pour certaines dimensions. Lorsque les mesures ne sont pas additives, la seule opération d'agrégation possible est de compter leur nombre. Dans l'exemple précédent, la mesure est la quantité expédiée par mois, par produit et par client.

2.4.1.4. ____ Granularité

La granularité correspond au niveau de détail qui est mémorisé dans chaque table. En général, la granularité est fixée de manière à correspondre au plus petit niveau de détail disponible dans les différentes tables dont on extrait des données. On pourra aussi fixer la granularité à partir des besoins de l'analyse qui sont déterminés par les utilisateurs finaux.

2.4.1.5. ____ Hiérarchie

Les hiérarchies représentent un concept très important en modélisation multi dimensionnelle. Elles décrivent les relations entre les différents niveaux des dimensions. Dans notre exemple, chaque dimension possède deux niveaux. La dimension temps contient le niveau « Année » et le niveau « Mois », Produit contient les niveaux « Marque » et « No Produit » et Client contient les niveaux « Ville » et No

Client ». Dans un cas réel, les dimensions pourront contenir un grand nombre de niveaux et plusieurs hiérarchies.

2.4.2. Le modèle de constellation (« Snowflake »)

Le modèle de constellation résulte de la décomposition d'une ou plusieurs dimensions qui possèdent des hiérarchies. Chaque niveau de hiérarchie est à l'origine d'une table ce qui permet d'éviter de dénormaliser les dimensions (Figure 2.9). La structure de constellation permet de représenter très efficacement les hiérarchies des dimensions. Elle est facile à comprendre à la fois par les créateurs et par les utilisateurs de ce type de modèle. Par contre, la structure en constellation nécessite des jointures entre toutes les tables qui constituent les dimensions et les performances de l'analyse peuvent devenir très mauvaises.

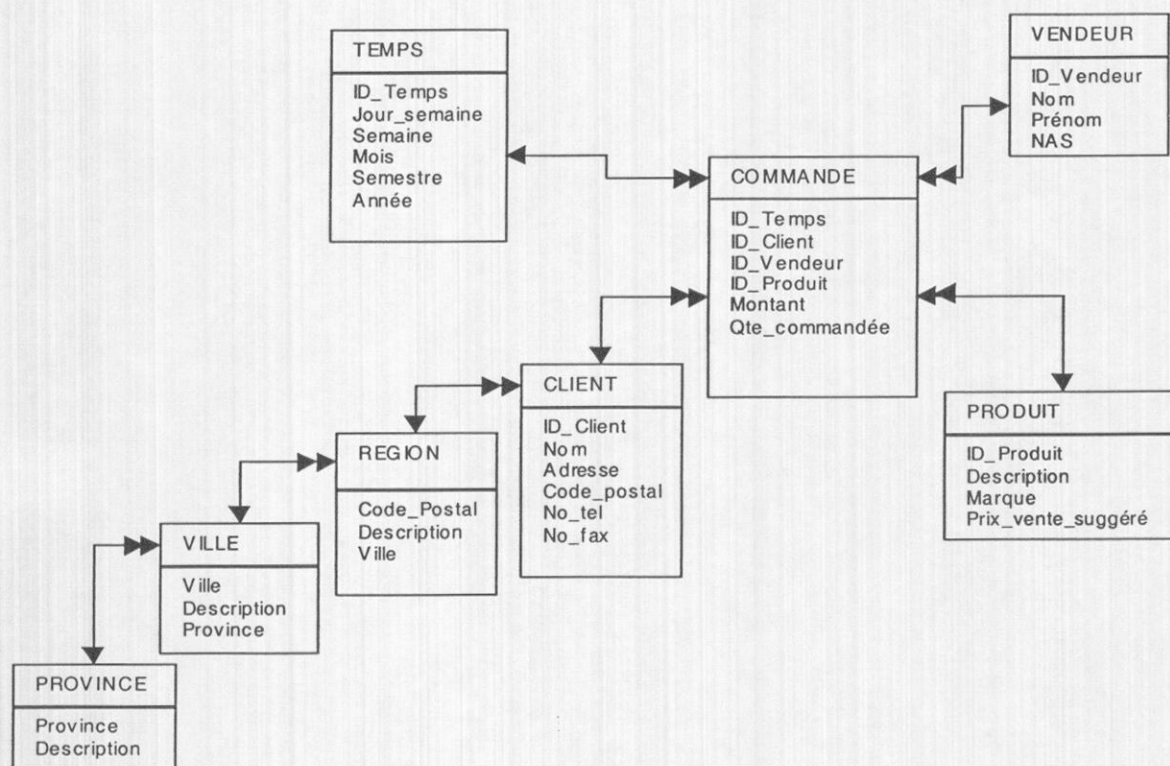


Figure 2.9: Exemple de schéma de constellation.

2.5. Mini entrepôt (data Mart)

Un mini-entrepôt de données peut être défini comme un entrepôt avec une perspective plus étroite qu'un entrepôt de données d'une compagnie (Ballard et al., 1998 ; Kimball, 1996). Les mini-entrepôts de données sont focalisés sur l'analyse de certains processus d'affaire et correspondent aux besoins d'un département particulier. Ils regroupent des sujets d'analyse d'intérêt pour un seul département qui possède en général l'ensemble du hardware, du software et des données. Comme chaque département a sa propre définition des données et de l'analyse qu'il veut effectuer, les mini-entrepôts sont construits pour répondre à ses besoins spécifiques et le modèle étoile développé est optimisé pour son analyse particulière.

Les mini-entrepôts peuvent être dépendants ou indépendants de l'entrepôt de données de l'entreprise. Lorsqu'il est dépendant, les données proviennent de l'entrepôt de données alors que s'il est indépendant, la base de donnée est peuplée de manière unique et séparée à partir des bases de données opérationnelles.

Le niveau de complexité d'un mini-entrepôt est relativement plus faible que celui d'un entrepôt de données et la tentation est grande de renoncer à développer l'entrepôt de données général de l'entreprise et de ne réaliser que des mini-entrepôts. Pour Inmon (2000), regrouper un ensemble de mini-entrepôts ne revient pas à créer un entrepôt de données complet et le choix de se limiter à la construction de mini-entrepôts est dangereux. En effet, selon lui la conception d'un mini-entrepôt est très différente de celle d'un entrepôt de données et la granularité des données d'un mini-entrepôt ne correspond pas à celle d'un entrepôt de données. Le mini-entrepôt contient des données très agrégées en fonction des besoins d'un département alors que l'entrepôt de données contient les données nécessaires à l'analyse de l'ensemble de l'entreprise. Par exemple, les personnes en charge des finances seront intéressées à certains types d'analyse qui ne seront évidemment pas les mêmes que celles qui sont en charge du marketing ou de la recherche-développement. De la même manière, la granularité des données sera différente pour le groupe qui s'occupe de la finance sur une base mensuelle et celui qui étudie la comptabilité sur une base trimestrielle. Cela signifie que les étoiles définies pour les différents groupes ne seront pas les mêmes et qu'il est impossible de définir une étoile qui puisse satisfaire tous les utilisateurs.

Ainsi, pour Inmon (2000) et Moddy et al. (2000), la modélisation multidimensionnelle est très efficace tant que le modèle se limite à une seule étoile. Elle peut être utilisée pour modéliser les données utilisées dans les mini-entrepôts (data mart). Par contre, elle ne peut pas l'être pour bâtir le modèle de l'entrepôt de données qui sera utilisé pour l'ensemble d'une compagnie.

Par contre, certains autres auteurs (Ballard, 1998 ; Kimball, 1996, Hüsemann et al., 2000 ; Pedersen et al., 1999) considèrent que les mini-entrepôts sont semblables aux entrepôts de données et font référence indistinctement à un mini-entrepôt ou à un entrepôt de données.

2.6. Outils OLAP

Codd et al. (1993) ont défini les caractéristiques de ce type d'analyse et ont introduit pour la première fois le terme OLAP (On Line Analytical Processing) en opposition à l'analyse des données opérationnelles (On Line Transactional Processing ou OLTP). Ils ont défini douze caractéristiques pour qu'un système puisse être appelé OLAP :

- Conception multidimensionnelle,
- Transparence,
- Accessibilité,
- Performance de production de rapport cohérente,
- Architecture Client-Serveur,
- Dimensionnalité générique,
- Gestion dynamique de matrices à faible densité,
- Support multi utilisateur,
- Opérations entre les dimensions non restreinte,
- Manipulation intuitive des données,
- Production de rapports flexible,
- Capacité à prendre en charge un nombre illimité de dimensions et de niveaux d'agrégation.

White (1997) a repris une définition fournie par le OLAP Council (organisme qui a pour objectif de promouvoir l'utilisation des outils OLAP) et dont la formulation est la suivante :

OLAP est une catégorie de software qui permet aux analystes, aux gestionnaires et aux membres de la haute direction d'une entreprise d'obtenir des informations grâce à un accès rapide, cohérent et interactif à une grande variété de vues des données qui ont été transformées à partir des données brutes pour refléter

le caractère multi dimensionnel de l'entreprise tel qu'il est compris par les utilisateurs (traduction libre de l'auteur).

Les opérations de base de l'analyse OLAP sont les suivantes :

- Détailler (drill down) et regrouper (roll up),
- Découper (slice) et pivoter (dice).

Détailler et regrouper sont les opérations pour se déplacer le long des niveaux de hiérarchie d'une dimension. Découper (slice) et pivoter (dice) permettent de naviguer au travers du cube lorsque les niveaux des dimensions sont fixés. Découper permet d'extraire des informations de manière à ce que les utilisateurs puissent analyser les données selon une certaine perspective. Pivoter permet de faire tourner le cube afin d'analyser les données d'une manière spécifique.

D'après Raden (1997), l'impact des règles définies par Codd et al. (1993) est relativement limité parce qu'elles sont trop vagues pour conduire à une définition précise des outils OLAP. Pour cet auteur, l'analyse OLAP signifie toujours la mise en place interactive de requêtes avec la poursuite de l'analyse en détaillant les résultats ou en regroupant suivant les hiérarchies des dimensions. L'analyse OLAP est heuristique, ce qui signifie qu'elle débute par une question et que la réponse est à l'origine de la question suivante. L'analyse interactive et heuristique nécessite des performances exceptionnelles de peur que la lenteur des réponses aux requêtes ne freine les réflexions de l'analyste.

Pour Raden (1997), les aspects des applications OLAP les plus importants parmi ceux qui ont été définis par Codd et al. (1993) sont la conception multi dimensionnelle, la faible densité des matrices et la possibilité de manipuler les données de manière interactive.

Conception multi dimensionnelle

Le modèle sous-jacent à toute application OLAP est le modèle multi dimensionnel que nous avons décrit plus haut, alors que les applications OLTP reposent sur des bases de données construites avec le modèle entité-relation.

Faible densité des données

La densité faible des données provient du fait que les mesures ne sont pas définies pour toutes les valeurs possibles des dimensions. Les mesures ont ainsi un grand nombre de valeur NULL et la table de

fait a une densité faible. Pour être efficaces, les outils OLAP doivent être capables de gérer cette densité faible et de présenter des performances qui ne sont pas trop affectées par la présence de valeur NULL.

Manipulation interactive

Les données multi dimensionnelles sont représentées par des tableaux croisés et les outils OLAP doivent offrir la possibilité de détailler les données ou au contraire de les regrouper selon certaines dimensions. De plus, il doit être possible de faire pivoter les données pour les analyser selon différentes perspectives.

Les outils OLAP peuvent être construits sur des tables relationnelles (ROLAP) ou sur des bases de données multi dimensionnelles (MOLAP). De plus en plus, les vendeurs de logiciel proposent des outils OLAP hybrident qui permettent d'analyser à la fois des bases de données relationnelles et multi dimensionnelles (DOLAP pour Database OLAP ou HOLAP pour Hybrid OLAP). Nous verrons par la suite la différence entre ces différents types d'outils OLAP.

2.6.1. Multidimensional OLAP (MOLAP)

Les outils MOLAP permettent d'analyser des bases de données multi dimensionnelles. Plutôt que de stocker les données sous forme d'enregistrements dans une table, les bases de données les stockent (au niveau logique) sous la forme de tableaux. Malheureusement ce type de stockage n'est pas normalisé et chaque compagnie de software propose son produit maison qui n'est pas compatible avec celui des autres marques. En particulier, les produits diffèrent parce qu'il n'y a pas de consensus sur le modèle multi dimensionnel, parce que les méthodes d'accès aux bases de données ne sont pas normalisées (comme le langage SQL), parce que chaque produit peut être placé dans sa propre catégorie et parce le domaine d'application de chaque produit diffère, passant d'une analyse très étroite effectuée par un seul opérateur à une analyse très générale. De plus, les produits construits selon le modèle MOLAP ne permettent pas de gérer les très grandes quantités de données présentes dans un entrepôt de données.

2.6.2. Relational OLAP (ROLAP)

Les outils ROLAP (Relationnal OLAP) permettent de faire des analyses multi dimensionnelles sur des tables relationnelles. Raden (1997) considère qu'un outil d'analyse sera de type ROLAP si :

- Il possède un générateur efficace de requête SQL, capable de créer des requêtes multiples et synchronisées permettant de réaliser des classements, des comparaison et des calculs de pourcentage.

- Il génère des requêtes optimisées,
- Il fournit un mécanisme pour décrire le modèle avec des métadonnées et qu'il utilise ces métadonnées en temps réel pour construire les requêtes,
- Il propose des moyens de construire des tables de résumé pour améliorer les performances,
- Il permet le partage des requêtes entre plusieurs clients et serveurs.

En résumé, les caractéristiques des outils MOLAP et ROLAP sont les suivantes :

	MOLAP	ROLAP
Capacité de dimensionnement	Faible	Grande
Maintenance	Facile	Lourde (en général)
Technologie	Pas normalisée	Normalisée, fait appel à des technologies éprouvées
Temps de réponse	Faible	Long

Tableau 2.1 : Caractéristiques des outils MOLAP et ROLAP.

Les outils DOLAP ou HOLAP offrent la possibilité d'analyser des bases de données relationnelles et multidimensionnelles simultanément. Ce genre de solution permet de profiter des avantages des outils MOLAP et ROLAP et de configurer les données sous la forme multi dimensionnelle ou relationnelle suivant le genre d'application (Moorman, 2000).

2.7. Le processus de développement

Au cours de ce paragraphe, nous allons situer le développement du modèle conceptuel dans le processus général de développement d'un entrepôt de données.

Le processus de développement a fait l'objet de plusieurs études (Ballard et al, 1998 ; Ladley, 1998). La Figure 2.10 présente les différentes étapes du cycle de développement considérées par Ballard et al. (1998). Cette représentation semble montrer qu'il n'y a qu'une seule instance du cycle, alors que celui ci peut se répéter plusieurs fois et peut être utilisé de manière parallèle lorsque l'organisation entreprend la mise en place de plusieurs mini-entrepôts. Les différentes étapes du processus de développement sont les suivantes :

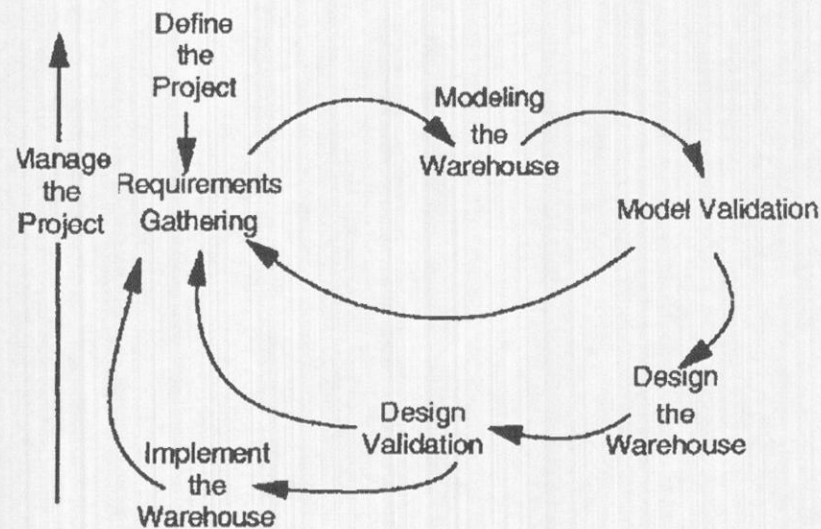


Figure 2.10 : Étapes de développement d'un entrepôt de données selon C. Ballard et al. (1998)

Notre travail de mémoire concerne la partie conceptuelle de l'étape de modélisation de l'entrepôt de données. Comme le montre la Figure 2.10, cette étape suit directement l'analyse des besoins qui ont été recueillis auprès des utilisateurs. Cette analyse conduira à la création d'une représentation schématique du modèle conceptuel qui pourra être interprété directement par les analystes. L'analyse des besoins permettra de déterminer les mesures, les faits, les dimensions avec leurs hiérarchies et la granularité.

Plusieurs approches peuvent être utilisées pour déterminer les éléments qui sont à la base d'un modèle multi dimensionnel. Ces approches diffèrent par la méthode d'établissement des différents éléments et par l'ordre de détermination des mesures, des faits et des dimensions. Les mesures, les dimensions et les faits peuvent être déterminés à partir des requêtes que les analystes effectueront pour analyser les processus d'affaire. Les mesures, les dimensions et les faits peuvent aussi être établis à partir des modèles des bases de données. Ces modèles sont en général sous la forme de modèle entité-relation et sont analysés pour en extraire les éléments des modèles multi dimensionnels. En ce qui concerne l'ordre de détermination des mesures, des dimensions et des faits, les approches les plus courantes sont les suivantes :

- Fait $\Rightarrow$ mesure $\Rightarrow$ dimension : on part du modèle entité-relation de l'entreprise pour identifier les faits à partir des entités de transaction, ensuite, on dresse la liste des mesures qui seront analysées et finalement on répertorie les dimensions et leurs hiérarchies qui vont permettre de caractériser les mesures.

-
- Fait $\Rightarrow$ dimension $\Rightarrow$ mesure : cette approche part des faits qui sont les processus d'affaire à analyser. Les dimensions sont déterminées en fonction de la capacité des entités à caractériser les entités de transaction et finalement les mesures sont les attributs des faits qui fourniront les résultats d'analyse.
 - Mesure $\Rightarrow$ dimension $\Rightarrow$ fait : cette approche part de la liste des attributs qui correspondent à des mesures d'analyse et détermine les dimensions qui caractérisent chacune des mesures. Les faits sont établis en regroupant les mesures qui dépendent fonctionnellement des mêmes dimensions.

Le chapitre suivant décrira la méthodologie qui a été suivie pour atteindre l'objectif de recherche.

3. Méthodologie

Rappelons que l'objectif de recherche est d'effectuer une étude comparative des différentes démarches de conception de modèles multidimensionnels utilisés pour des entrepôts de données et de comparer les modèles construits à partir des différentes démarches. Il s'agit de guider le choix de la démarche pour construire un entrepôt de données d'une entreprise à partir des analyses qui seront effectuées ou des modèles de données déjà disponibles.

Pour atteindre cet objectif de recherche, nous avons réalisé dans un premier temps, un recensement des méthodes de design conceptuel multidimensionnel des entrepôts de données. Nous avons ensuite sélectionné quatre démarches de conception en fonction des différentes approches proposées dans la littérature pour la détermination des éléments du modèle multidimensionnel et de l'ordre suivant lequel ces éléments sont établis.

On a vu en effet au chapitre précédent que les mesures, les dimensions et les faits d'un modèle multidimensionnel peuvent être déterminés à partir d'un ensemble de requêtes définies par un groupe d'utilisateurs ou à partir de l'analyse des modèles entité-relation des bases de données opérationnelles. Nous avons vu aussi que les déterminations des mesures, des dimensions et des faits sont réalisées dans des ordres différents suivant la démarche adoptée.

La construction d'un entrepôt de données à partir de l'analyse du modèle entité-relation des bases de données opérationnelles a été effectuée en prenant le modèle de Tarigomi que nous avons modifié pour permettre l'analyse de la gestion du stock et des approvisionnements. De plus, nous avons rajouté la possibilité à l'entreprise de disposer de plusieurs entrepôts puisque le modèle original ne comporte pas de notion d'entrepôt. On trouvera au chapitre 4 une description du modèle original des données de l'entreprise Tarigomi ainsi que les modifications qui ont été apportées.

Pour construire l'entrepôt de données à partir des requêtes des utilisateurs, nous avons établi des sujets sur lesquels des analyses seront effectuées. Ces sujets seront la prise et le suivi des commandes, la prise et le suivi des lignes de commande qui correspondent chacune à des produits différents, le suivi de l'inventaire, la réquisition et le suivi des approvisionnements et de chaque ligne d'approvisionnement.

Nous avons développé quatre modèles multi dimensionnels différents qui ont été analysés en effectuant des requêtes soit au moyen de commandes SQL, soit en les implantant dans le logiciel Discoverer de Oracle.

Pour effectuer cette analyse, nous avons établi une liste de requêtes que nous avons appliquées aux différents modèles qui ont été construits. On trouvera en annexe une description des requêtes ainsi que le code SQL qu'il a été nécessaire de créer pour les effectuer sur les quatre modèles.

Les analyses qui sont effectuées par Discoverer nécessitent l'emploi de deux logiciels, Discoverer Administrator et Discoverer User. Dans Discoverer Administrator, l'administrateur organise les données et crée un environnement que l'utilisateur pourra explorer en effectuant des requêtes. Il sera amené, entre autres, à créer des dossiers complexes (« complex folder »), à rajouter des champs calculés ou à créer des jointures entre les dossiers. L'utilisateur ne pourra que consulter les données qui ont été organisées par l'administrateur. Dans notre travail, nous avons utilisé un ensemble de requêtes identiques pour les quatre démarches. Le travail sous Discoverer User est donc pratiquement identique dans les quatre cas. Seule l'origine des données changera. Par contre, le travail d'organisation des données sera très différent d'un modèle multi dimensionnel à l'autre. Nous présenterons dans le chapitre 5 les différentes activités qui ont été nécessaires dans Discoverer Administrator pour créer l'environnement d'analyse par Discoverer User.

L'analyse des modèles développés et des requêtes nous a conduit à comparer les différentes démarches et à proposer un guide de choix de la démarche en fonction de la disponibilité d'un modèle entité-relation de données de l'entreprise, de la qualité de ce modèle et de l'implication des utilisateurs pour développer le modèle multi dimensionnel.

4. Le modèle de Tarigomi et les requêtes

Au cours du chapitre 4, nous allons décrire brièvement le modèle de Tarigomi et les données dont nous disposons. Ensuite, nous allons décrire une liste de requêtes qui ont été utilisées pour construire le modèle lorsque la démarche le nécessitait ou pour effectuer les analyses. Pour expliquer les requêtes, nous avons constitué une petite série de données très simples mais qui sont organisées selon le même modèle que Tarigomi.

4.1. Tarigomi

Les données de Tarigomi, qui ont été simulées, correspondent aux ventes effectuées par un grossiste de produits électroniques. Ces produits pourront être des télévisions, des radios, des balladeurs, etc. Le modèle de Tarigomi tel qu'il sera utilisé dans ce travail tient compte des approvisionnements et permet une analyse de la gestion des stocks (voir Figure 4.1). De plus, la structure du modèle permet à l'organisation de disposer de plusieurs entrepôts qui sont chacun dédiés à un ensemble de magasin. Les magasins ne reçoivent leurs fournitures que d'un seul entrepôt. Les entrepôts sont approvisionnés par des fournisseurs qui reçoivent des réquisitions de la part de Tarigomi. Chaque réquisition est effectuée auprès d'un fournisseur mais peut concerner plusieurs produits. La structure des approvisionnements est la même que celle des commandes. Il est prévu une procédure de transfert pour équilibrer le stock entre les entrepôts. En effet, si un client fait la commande pour un magasin d'un produit qui ne se trouve pas dans l'entrepôt assigné au magasin mais qui se trouve dans un autre entrepôt de Tarigomi, l'entreprise fera le transfert entre les deux entrepôts pour que le client soit servi le plus rapidement possible.

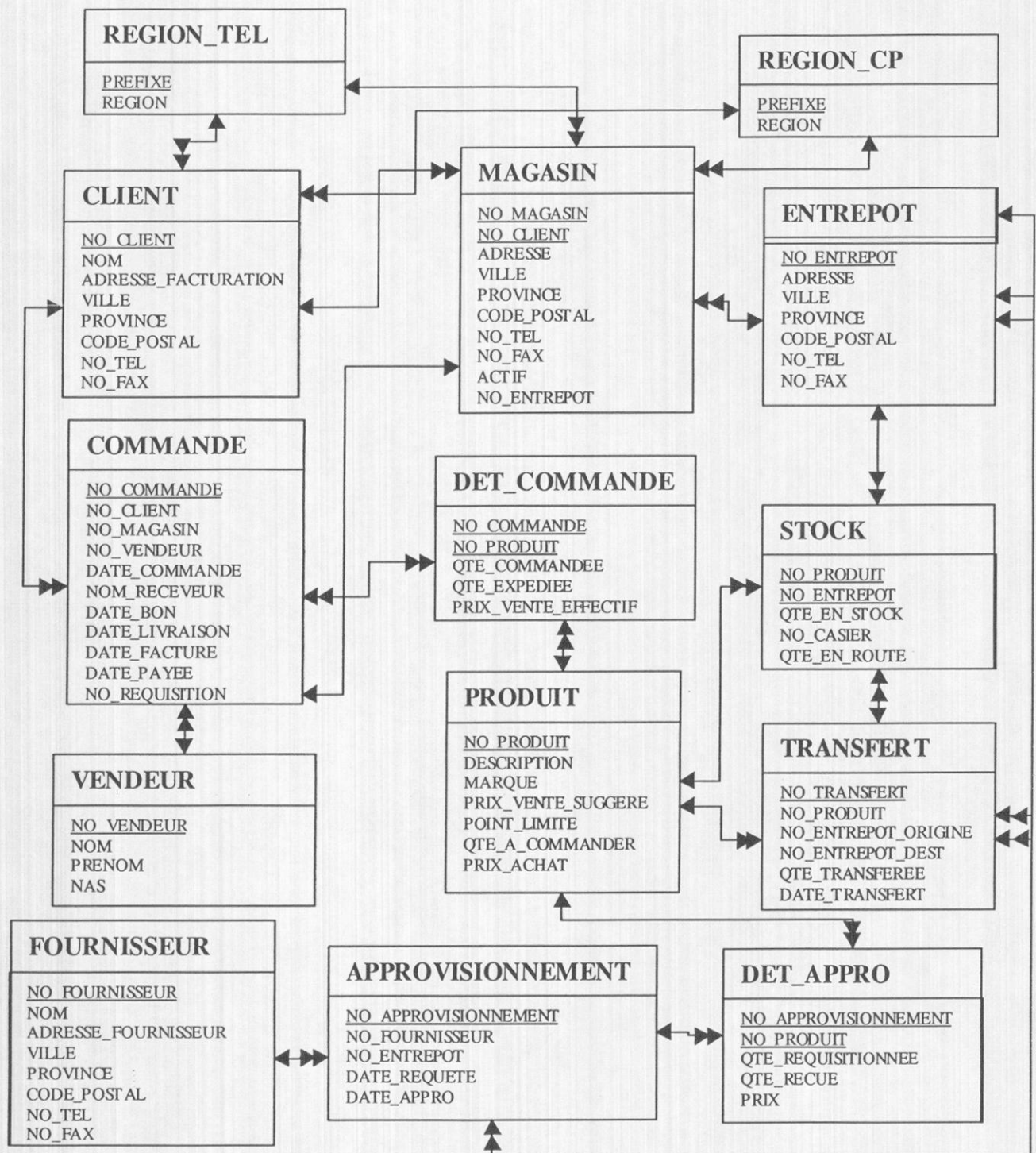


Figure 4.1 : Diagramme de structure des données de Tarigomi tel qu'il a été développé dans ce travail pour tenir compte des approvisionnements et introduire la notion d'entrepôt.

4.2. Requêtes

Nous avons dressé une liste de requêtes pour analyser les commandes et les lignes de commande de Tarigomi. De plus, nous avons préparé deux requêtes du suivi de l'inventaire des produits, même si les données dont nous disposons ne contiennent pas d'information sur l'état du stock au cours du temps. Nous ne pourrions donc pas effectuer les requêtes pour faire le suivi de l'inventaire, mais nous pourrions construire le modèle multi dimensionnel.

Le Tableau 4.1 contient la liste des requêtes que nous avons définies sur Tarigomi.

No	Requêtes
R1	Valeur moyenne des montants mensuels totaux par client.
R2	Nombre de commandes et ventes totales par vendeur, par client, par mois, par année.
R3	Délai moyen de livraison, par client, par mois, par année.
R4	Nombre moyen de lignes de commande par commande pour chaque vendeur, pour chaque mois de toutes les années confondues.
R5	Valeur commandée par marque par jour de la semaine en pour cent de la valeur totale des commandes.
R6	Rabais en pour-cent par vendeur, par marque, par mois et par année.
R7	Ventes totales par année, par client, par magasin, par marque.
R8	Prix moyen de vente effectif par produit par année, par mois.
R9	Nombre de transferts entre entrepôt par produit, par année, par mois.
R10	Quantité moyenne quotidienne en stock par produit, par année, par mois.

Tableau 4.1: Liste des requêtes.

En annexe 9, nous présentons des données simplifiées et le résultat de chacune de ces requêtes. Nous reprenons ici une partie des informations de l'annexe pour décrire les requêtes et pour en illustrer les résultats.

4.2.1. Valeur moyenne des montants mensuels totaux par client.

La première requête fait l'analyse des montants mensuels moyens pour chaque client. La valeur moyenne commandée est calculée entre les mois identiques d'une année à l'autre en prenant la valeur totale commandée pendant un mois de l'année par client. Le client 1001 par exemple a commandé en janvier 90 pour un montant total de 1100\$, en janvier 91 pour 2500\$ et en janvier 92 pour 5400\$. Le total des commandes du client pour les mois de janvier pendant les 3 années étudiées est donc de 9000\$.

ce qui fait un montant moyen des commandes mensuelles de 3000\$. Le résultat de l'analyse donne le tableau suivant sur les données simplifiées:

Client	Jan	Fev
1001	3 000,00 \$	- \$
1002	- \$	2 066,67 \$

Tableau 4.2 : Valeur moyenne des montants mensuels totaux par client (requête 1).

4.2.2. Nombre de commandes et ventes totales par vendeur, par client, par mois, par année

La deuxième requête fournit le nombre de commandes et le montant total des ventes pour chaque client, pour chaque vendeur, pour chaque mois de chaque année. Par exemple, le vendeur numéro 1 a vendu 3 commandes d'un montant total de 650\$ au client 1001 pendant le mois de janvier 1990. Le résultat de la requête fournit le tableau suivant:

Vendeur	Client	90				91				92			
		Janvier		Février		Janvier		Février		Janvier		Février	
		Nb	Valeur	Nb	Valeur	Nb	Valeur	Nb	Valeur	Nb	Valeur	Nb	Valeur
1	1001	3	650 \$	0	- \$	1	600 \$	0	- \$	2	2800	0	- \$
	1002	0	- \$	2	350 \$	0	- \$	0	- \$	0	0	1	2 000 \$
2	1001	2	450 \$	0	- \$	2	1 900 \$	0	- \$	2	2600	0	- \$
	1002	0	- \$	2	350 \$	0	- \$	1	1 500 \$	0	0	1	2 000 \$

Tableau 4.3 : Nombre de commandes et valeur totale par vendeur par client (requête 2).

4.2.3. Délai moyen de livraison, par client, par mois, par année

La troisième requête calcule le délai de livraison moyen par client, par mois et par année. Dans le cas des données simplifiées, les commandes ont été livrées 0, 1, 2 et 3 jours après leur enregistrement. Ainsi le délai total des commandes passées par le client 1001 pendant le mois de janvier 1990 est de 7 jours et il y a eu 5 commandes. Le délai moyen a donc été de 1.4 jours en janvier 1990. Le résultat de la requête sur les données simplifiées est le suivant :

Client	90		91		92	
	Janvier	Février	Janvier	Février	Janvier	Février
1001	1,4	0	1,33	0	1,25	0
1002	0	1	0	2	0	1,5

Tableau 4.4 : Délai moyen de livraison, par client, par mois, par année (requête 3).

4.2.4. Nombre moyen de lignes de commande par commande pour chaque vendeur, pour chaque mois de toutes les années confondues

Cette requête permet de calculer la charge de travail pour chaque vendeur en fonction du mois de l'année. Le vendeur numéro 1 par exemple, a effectué 6 commandes en janvier qui comprenaient un nombre total de 8 lignes de commande. Le nombre moyen de lignes de commande par commande en janvier est donc de 1.33. Le résultat de la requête sur les données simplifiées est donc :

Vendeur	Janvier	Février
1	1,33	1
2	2,17	2,25

Tableau 4.5 : Nombre moyen de lignes de commande par commande pour chaque vendeur, pour chaque mois de toutes les années confondues (requête 4).

4.2.5. Valeur commandée par marque par jour de la semaine en pour cent de la valeur totale des commandes

La requête numéro 5 permet d'analyser l'importance des commandes d'un produit en fonction du jour de la semaine. Par exemple, les commandes des produits de la marque Candle ont atteint 2350\$ les lundis et la valeur totale de l'ensemble des commandes est de 15200\$. La valeur en pour cent est donc de 15.46%.

La requête 5 sur les données simplifiées fournit les résultats suivants:

Marque	Lundi	Mardi	Mercredi	Jeudi	Vendredi	Samedi	Dimanche	Total
Candle	15,46%	13,49%	1,32%	28,36%	9,08%	5,66%	14,80%	88,17%
Sony	1,31%	2,96%	0,00%	0,26%	3,75%	3,55%	0,00%	11,83%
Total par marque	16,77%	16,45%	1,32%	28,62%	12,83%	9,21%	14,80%	100,00%

Tableau 4.6 : Valeur commandée par marque par jour de la semaine en pour cent de la valeur totale des commandes (requête 5).

4.2.6. Rabais en pour-cent par vendeur, par marque, par mois et par année

Cette requête permet de calculer le rabais accordé par chaque vendeur sur les produits de chaque marque au cours de chaque mois de chaque année. Le vendeur numéro 1, par exemple, a accordé des

rabais de 26,21% sur les produits de la marque Candle au cours du mois de janvier 1990. Le Tableau 4.7 montre les résultats de la requête sur les données simplifiées.

Vendeur	Marque	90		91		92	
		Janvier	Février	Janvier	Février	Janvier	Février
1	Candle	26,21%	35,34%	8,95%	0,00%	6,90%	3,75%
1	Sony	0,00%	0,00%	0,00%	0,00%	34,71%	0,00%
2	Candle	10,59%	11,09%	15,49%	19,61%	3,79%	0,47%
2	Sony	12,95%	12,95%	34,71%	34,71%	34,71%	0,00%

Tableau 4.7 : Rabais en pour-cent par vendeur, par marque, par mois et par année (requête 6).

Le résultat de cette requête pourra être faux si le prix suggéré de vente a changé entre le moment où la commande a été passée et celui où l'analyse est effectuée. Pour garantir la justesse du calcul, il faudrait mémoriser l'évolution temporelle du prix suggéré.

4.2.7. Ventes totales par année, par client, par marque.

La requête permet de calculer l'importance de chaque marque pour chaque client par année. Par exemple, la marque Candle a généré des commandes de 1 900 \$ qui ont été passées par le client 1001 pendant 1990.

Le résultat de la requête sur les données simplifiées est représenté sur le Tableau 4.8.

Client	Marque	1990	1991	1992
1001	Candle	1 900 \$	2 050 \$	4 860 \$
1001	Sony	200 \$	450 \$	540 \$
1002	Candle	540 \$	1 050 \$	4 000 \$
1002	Sony	160 \$	450 \$	4 000 \$

Tableau 4.8 : Ventes totales par année, par client, par magasin, par marque (requête 7).

4.2.8. Prix moyen de vente effectif par produit par année, par mois

La requête 8 permet de faire l'étude de l'évolution du prix effectif en fonction du temps par mois et par année. Le prix effectif moyen est pondéré par le nombre d'item commandés. Le produit 1001, par exemple, a été commandé 10 fois à 25\$ et 2 fois à 40\$. Le montant total commandé est donc de $250\$ + 80\$ = 330\$$ et le prix moyen du produit de 27.50\$. Le tableau suivant montre les résultats de la requête obtenus sur les données simplifiées.

Produit	90			91			92		
	Janvier	Février	...	Janvier	Février	...	Janvier	Février	...
1001	27,50 \$	27,50 \$		37,00 \$	35,00 \$		35,00 \$	- \$	
1002	40,00 \$	40,00 \$		30,00 \$	30,00 \$		30,00 \$	-	
1003	26,43 \$	26,25 \$		27,44 \$	25,00 \$		25,00 \$	32,95 \$	
1005	50,00 \$	-		-	-		50,00 \$	50,00 \$	
1009	-	-		-	-		200,00 \$	200,00 \$	

Tableau 4.9 : Évolution du prix moyen de vente effectif par produit par année, par mois (requête 8).

4.2.9. Nombre de transferts entre entrepôt par produit, par année, par mois

L'entité qui représente les transferts est un fichier de transaction et contient les informations temporelles nécessaires. Il serait donc possible d'appliquer la requête sur la table de Tarigomi si cette table existait. Le résultat de la requête prendrait la forme suivante :

Produit	90			91			92		
	Janvier	Février	...	Janvier	Février	...	Janvier	Février	...
1001	10	20		30	10		15	4	
1002	8	0		5	25		70	24	
1003	44	34		12	45		8	16	

Tableau 4.10 : Nombre de transferts par produit, par mois et par année (requête 9).

4.2.10. Quantité moyenne quotidienne en stock par produit, par année, par mois

Pour effectuer cette requête, il faut disposer des informations sur l'état du stock en fonction du temps. La base de données originale de Tarigomi ne contient pas ces informations. La base de données de Tarigomi modifié contient un fichier de stock mais celui-ci n'est pas un fichier de transaction et ne contient pas d'information temporelle. Seul l'entrepôt de données dans lequel on fait des snapshots à intervalle régulier permet de faire cette analyse.

Si on avait les données pour effectuer cette requête, le résultat aurait l'allure suivante :

Produit	90		91		92	
	Janvier	Février	Janvier	Février	Janvier	Février
1001	30	50	20	40	30	10
1002	5	0	2	15	4	6
1003	25	20	23	21	20	15

Tableau 4.11 : Quantité moyenne quotidienne en stock par produit, par mois et par année (requête 10).

4.2.11. Synthèse des requêtes

Le tableau suivant fait la synthèse des requêtes et indique s'il est nécessaire de créer une vue pour la réaliser sur les données simplifiées.

Requête	Regroupement	Analyse	Création de vues
Requête 1	NO_CLIENT, MOIS, ANNEE	Montant mensuel des commandes	Oui car moyenne calculée n'est pas calculée sur le regroupement
Requête 2	NO_VENDEUR, NO_CLIENT, MOIS, ANNEE	Nombre de commandes, Montant total	Oui car la mesure du montant des commandes n'existe pas et qu'on ne peut pas calculer le nombre de commandes sur DET_COMMANDE
Requête 3	NO_CLIENT, MOIS, ANNEE	Délai moyen	Non
Requête 4	NO_VENDEUR, MOIS	Nombre moyen de lignes de commande par commande	Oui car la mesure du nombre de lignes par commande n'existe pas
Requête 5	MARQUE, JOUR_DE_SEMAINE	Montant commandé / valeur totale	Non
Requête 6	NO_VENDEUR, MARQUE, MOIS, ANNEE	Rabais moyen / montant de la commande	Non
Requête 7	NO_CLIENT, MARQUE, ANNEE	Montant total	Non
Requête 8	NO_PRODUIIT, MARQUE, MOIS, ANNEE	Prix de vente moyen	Non
Requête 9	NO_PRODUIIT, MOIS, ANNEE	Nombre de transferts	Non
Requête 10	NO_ENTREPOT, MOIS, ANNEE	Quantité moyenne en stock	Non

Le chapitre suivant donne la description des démarches qui ont été choisies et leur application aux données de Tarigomi. L'utilisation de ces démarches a abouti à la construction de quatre modèles multi dimensionnels qui ont été analysés avec les requêtes qui ont été présentées dans ce chapitre.

5. Construction des modèles

Au cours de ce chapitre, nous allons présenter en détail les démarches de conception qui ont été choisies en fonction de l'origine des requêtes et de l'ordre des éléments qui constituent le schéma multidimensionnel. Nous avons retenu quatre démarches de conception qui sont proposées par différents auteurs et nous les avons appliquées à l'entreprise fictive Tarigomi. Nous avons ensuite effectué les analyses qui ont été décrites au chapitre 4. Ces analyses ont été effectuées directement avec des commandes SQL ou en utilisant le logiciel Discoverer.

La première démarche que nous avons choisie s'appuie uniquement sur le diagramme entité-relation de l'entreprise (Moody et Kortink, 2000). Le modèle multidimensionnel est bâti en analysant les entités et les attributs de ce modèle. L'ordre de détermination des éléments du modèle multidimensionnel est le suivant : fait $\Rightarrow$ mesure $\Rightarrow$ dimension.

La deuxième démarche retenue a été proposée par Hüsemann et al. (2000) et s'appuie aussi sur le modèle entité-relation de l'organisation. Les éléments du modèle multidimensionnel sont déterminés à partir d'un tableau des attributs qui a été établi par un groupe d'experts du domaine des affaires. L'ordre de détermination des éléments du modèle multidimensionnel est le suivant : mesure $\Rightarrow$ dimension $\Rightarrow$ fait.

La troisième démarche a été proposée par Kimball (1996) qui propose de construire le modèle multidimensionnel à partir de l'analyse des requêtes des utilisateurs. Les besoins sont déterminés par des entretiens avec les différents groupes des usagers et sont ensuite analysés pour déterminer les faits, les dimensions et les mesures (Fait $\Rightarrow$ Dimension $\Rightarrow$ Mesure).

Finalement, la quatrième démarche a été proposée par Ballard et al. (1998) qui construisent un modèle multidimensionnel en partant à la fois d'une liste de requêtes pour déterminer les mesures qui seront nécessaire pour mener à bien les analyses et du modèle entité-relation pour choisir les dimensions. L'ordre de détermination des éléments du modèle multidimensionnel est le suivant :

Mesure $\Rightarrow$ Dimension $\Rightarrow$ Fait

5.1. Moody et Kortink

Moody et Kortink (2000) proposent une démarche pour construire un modèle multidimensionnel à partir d'un schéma du modèle de données de l'entreprise représenté sous la forme d'un modèle entité-relation. Par rapport à l'approche centrée sur la définition des besoins par les utilisateurs, cette approche

constitue à leurs yeux un moyen plus structuré pour construire l'entrepôt de données et assure que ce dernier reflète la structure sémantique des données. Elle conduit aussi à une conception plus souple de l'entrepôt de données qui résiste mieux aux changements des besoins d'analyse au cours du temps.

La démarche de Moody et Kortink consiste à déterminer tout d'abord les faits et les mesures avant de terminer par la création des dimensions.

Fait $\Rightarrow$ mesure $\Rightarrow$ dimension (FMD)

5.1.1. Exemple

Pour illustrer leur démarche, Moody et Kortink (2000) s'appuient sur l'exemple d'une compagnie de vente au détail. La Figure 5.1 montre le schéma entité-relation de la compagnie avec les différentes entités présentes dans les bases de données opérationnelles.

Ce modèle ne contient pas de redondances et maximise ainsi l'efficacité des mises à jours et représente explicitement les données et les relations entre elles. Malheureusement, la plupart des gestionnaires ou des preneurs de décision ne sont pas à l'aise avec ce type de modèle et ne le comprennent tout simplement pas. De plus, les performances des requêtes peuvent s'avérer catastrophiques du fait qu'elles nécessitent de nombreuses jointures et des sous requêtes complexes. Il en résulte que les utilisateurs vont dépendre de spécialistes techniques qui vont devoir rédiger les requêtes pour eux.

La construction du modèle multidimensionnel est effectuée en réalisant les étapes suivantes :

- Classification des entités,
- Identification des hiérarchies
- Production du modèle dimensionnel
- Évaluation et raffinements

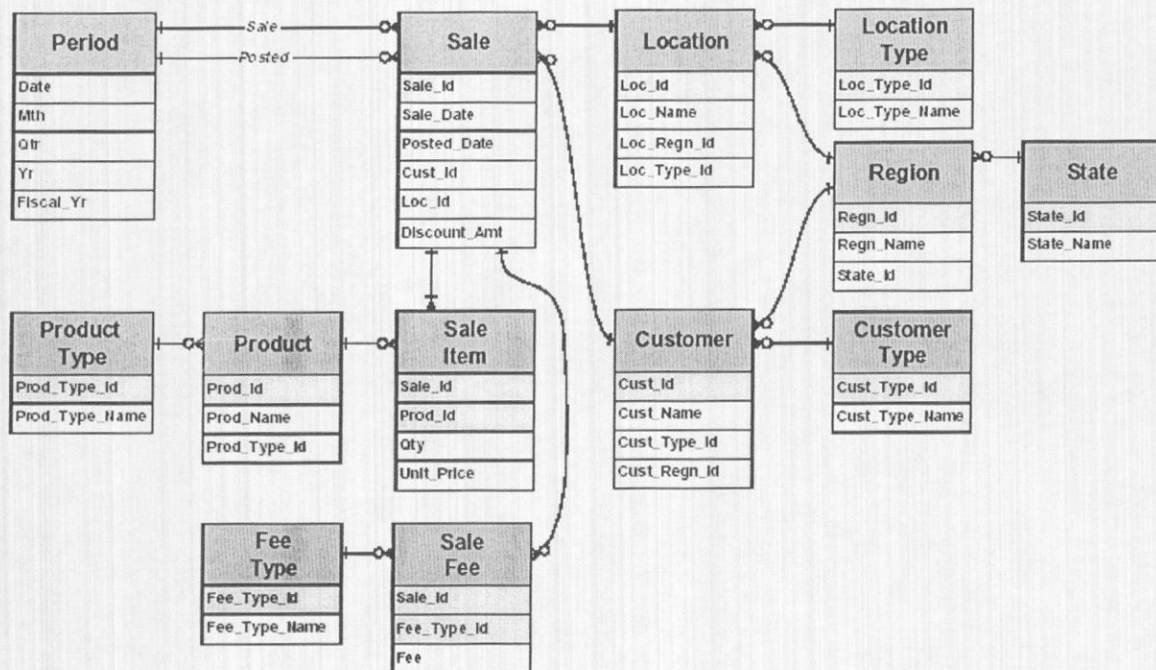


Figure 5.1 : Exemple de modèle entité-relation.

5.1.2. Classification des entités

La première étape consiste à classer les entités du modèle entité-relation en trois catégories :

- Les entités de transaction,
- Les entités de composition,
- Les entités de classification.

Les **entités de transaction** contiennent des détails sur des événements particuliers du déroulement des affaires. Par exemple, les commandes, les plaintes d'assurance, les paiements de salaire ou les réservations d'hôtel. Ce sont toujours ces éléments que le preneur de décision voudra comprendre et analyser. Les caractéristiques fondamentales d'une entité de transaction sont les suivantes :

- Elles décrivent un événement qui a eu lieu à un instant précis,
- Elles contiennent des mesures ou des quantités qui peuvent être additionnées ou regroupées comme par exemple des montants en dollar, des poids, des durées ou des volumes.

Les entités de transaction sont les entités les plus importantes d'un entrepôt de données et forment la base pour la construction des tables de faits dans un modèle étoile. Toutes les entités de transaction ne seront pas utiles pour la prise de décision et il est important que les utilisateurs identifient lesquelles seront importantes et feront partie du modèle multidimensionnel.

Une **entité de composition** est directement reliée à une entité de transaction par une relation de 1 à N. Les entités de composition définissent les détails (les composantes) de chaque transaction. Elles répondent aux questions « qui », « quoi », « quand », « où », « comment » et « pourquoi » qui sont posées à propos d'une transaction. Par exemple, une transaction de commande peut être définie par les composantes suivantes :

- Client : qui a passé la commande ?
- Produit : qu'est-ce qui a été commandé ?
- Vendeur : par qui la commande a-t-elle été prise ?
- Date : quand la commande a-t-elle été passée ?

L'analyse temporelle d'une transaction est très importante et cette entité doit faire partie de tout entrepôt de données. Les entités composantes constituent la base pour la construction des dimensions dans un modèle étoile.

Les **entités de classification** sont reliées aux entités composantes par une chaîne de relations de 1 à N qui dépendent fonctionnellement d'une entité composante. Les entités de classification représentent les hiérarchies qui font partie du modèle de données. Ces hiérarchies peuvent être rabattues pour former les tables de dimension du modèle étoile.

La Figure 5.2 montre le classement des entités du modèle de l'exemple. En noir, Moody&Kortink ont représenté les entités de transaction, en gris les entités composantes et en blanc les entités de classification.

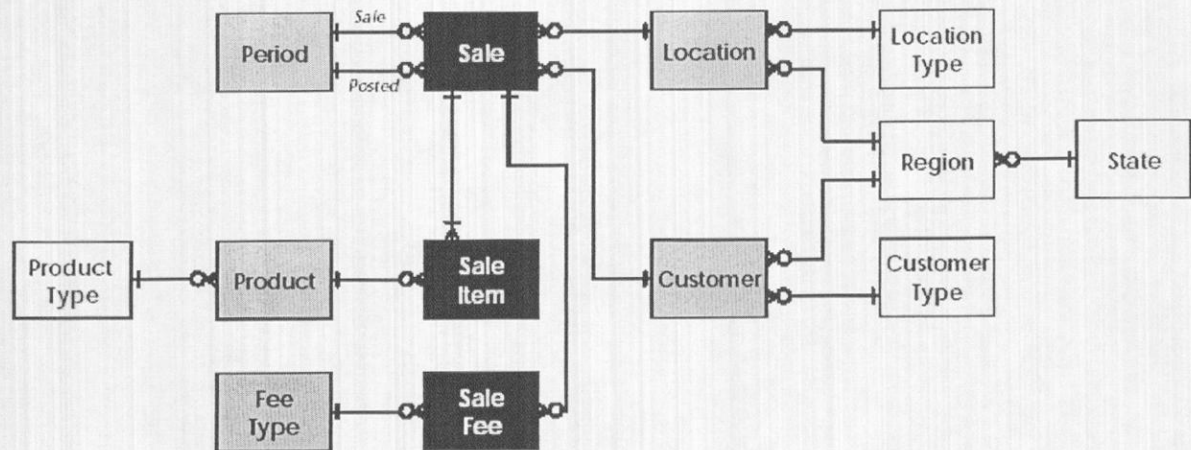


Figure 5.2 : Classement des entités.

Dans certains cas, les entités peuvent être classées dans plusieurs catégories et peuvent correspondre par exemple à la fois à des entités de transaction et à des entités composantes. Moody&Kortink proposent alors de résoudre ce type d'ambiguïté en fixant un ordre de préséance entre les différentes entités :

Entité de transaction $\Rightarrow$ entité de classification $\Rightarrow$ entité composante

Application à Tarigomi

Dans le cas de Tarigomi, le schéma entité-relation se présente sous la forme de la Figure 5.3. L'entité STOCK a été insérée même si elle ne correspond ni à une entité de transaction, ni à une entité composante. Elle a été rajoutée pour pouvoir faire des analyses de stock.

Les entités temporelles ont été rajoutées dans le schéma de l'entrepôt de données pour permettre d'effectuer des analyses en fonction du temps. Ce rajout est un travail préalable avant d'appliquer la méthode de Moody&Kortink. L'introduction des entités Période Commande et Période Approvisionnement est relativement simple dans la mesure où Commande et Approvisionnement sont des entités de transaction qui contiennent déjà un attribut temporel. Par contre, l'entité Fait Stock ne contient pas un tel attribut. Pour que l'entrepôt de données contienne un historique du stock, l'entité de l'entrepôt de données sera constituée d'une série d'images instantanées (snapshots) de l'entité Stock.

Dans le schéma de structure des données de Tarigomi présenté à la Figure 4.1, l'entité COMMANDE contenait les attributs DATE_BON, DATE_LIVRAISON, DATE_FACTURE et DATE_PAYEE. Ces attributs ont été transformés en délai, c'est-à-dire l'écart entre chacune des dates et la date de commande. Ces attributs deviennent ainsi des nombres sur lesquels il sera possible de faire des analyses. Pour

simplifier les tables construites, seul le délai de livraison sera calculé. Les entités de transaction ont été dessinées en gris foncé, les entités composantes en gris clair et les entités de classification en blanc.

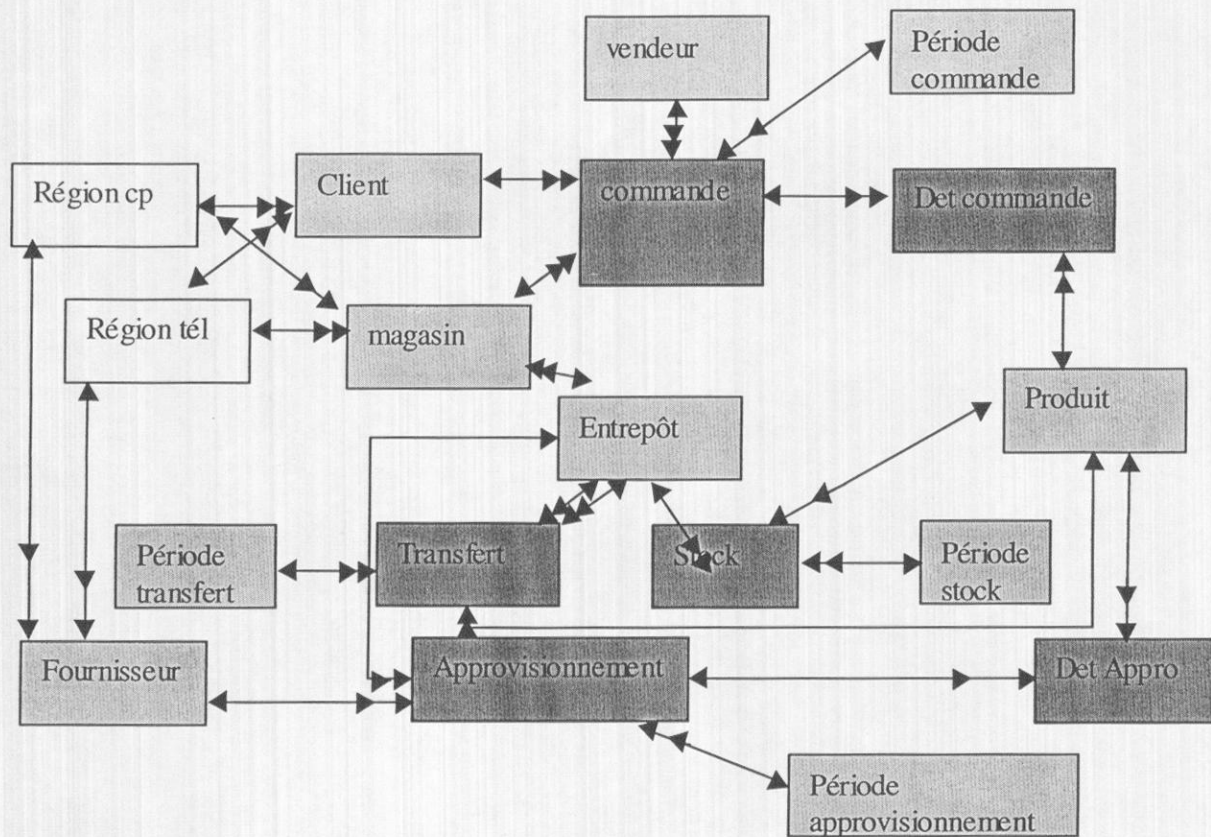


Figure 5.3 : Schéma entité-relation de Tarigomi dans lequel les entités de transaction ont été représentées en gris foncé, les entités composantes en gris clair et les entités de classification en blanc.

Les entités de transaction sont les suivantes:

- Commande,
- Det commande
- Approvisionnement
- Det Appro
- Transfert
- Stock

Les entités composantes

- Client,
- Magasin,
- Vendeur
- Produit
- Période
- Entrepôt
- Fournisseur

Et les entités de classification

- Région tél
- Région cp

5.1.3. Identification des hiérarchies

Les hiérarchies sont un concept très important de la modélisation multidimensionnelle et la plupart des dimensions d'un modèle étoile en contiennent. Une hiérarchie dans un modèle entité-relation est représentée par une séquence d'entités reliées par des relations de 1 à N qui sont toutes alignées dans la même direction.

Hiérarchie maximale

Une hiérarchie est dite maximale si elle ne peut pas être étendue vers le haut ou vers le bas en incluant d'autres entités.

On appelle entité minimale l'entité qui se trouve à l'extrémité inférieure de la hiérarchie maximale et entité maximale celle qui se trouve à l'extrémité supérieure. Les entités minimales sont facilement identifiées par le fait qu'elles ne possèdent pas de relation de 1 à N et les entités maximales n'ont pas de relation de N à 1.

Application à Tarigomi

Dans le cas de Tarigomi, on peut identifier 18 hiérarchies maximales :

- Produit – det commande
- Vendeur – commande – det commande
- Region cp – magasin – commande – det commande

- Region tél – magasin – commande – det commande
- Region cp – client – commande – det commande
- Region tél – client – commande – det commande
- Période(commande) – commande – det commande
- Entrepôt – magasin – commande – det commande
- Produit – det appro
- Fournisseur – approvisionnement – det appro
- Entrepôt – approvisionnement – det appro
- Période (approvisionnement) – approvisionnement – det appro
- Produit – transfert
- Entrepôt – transfert
- Période (transfert) – transfert
- Produit – stock
- Entrepôt – stock
- Période (stock) – stock

Les entités minimales sont les suivantes :

- Det commande
- Det appro
- Transfert
- stock

Les entités maximales

- Période(commande)
- Période (approvisionnement)
- Période (transfert)
- Période (stock)

- Région cp
- Région tél
- Vendeur
- Produit
- Entrepôt
- Fournisseur

5.1.4. Production du modèle multidimensionnel

Moody&Kortink utilisent deux opérateurs pour produire le modèle multidimensionnel à partir du modèle entité-relation. Il s'agit de l'opérateur de rabattement et de l'opérateur de regroupement.

Rabattement d'une hiérarchie

Les entités maximales et celles qui font partie d'une hiérarchie peuvent être rabattues dans les entités des niveaux inférieurs. Cette opération introduit de la redondance et le schéma n'est plus en troisième forme normale. Le rabattement est donc une forme de dénormalisation du schéma.

Dans l'exemple fourni par Moody&Kortink, l'entité « State » est rabattue dans l'entité « Region » et tous les attributs de l'entité « State » font partie de l'entité « Region ».

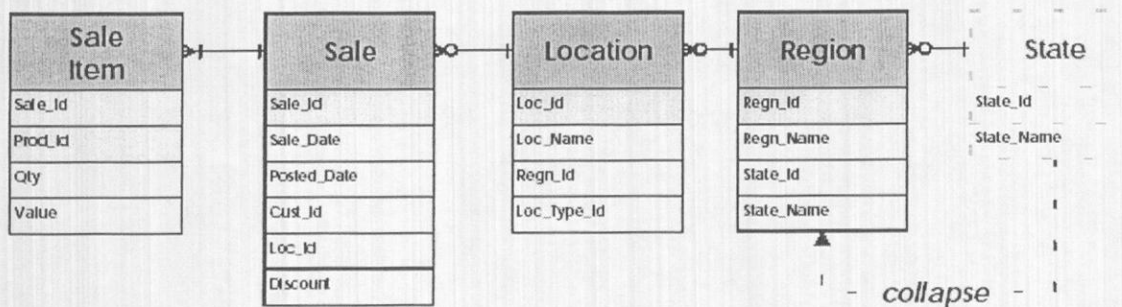


Figure 5.4 : Rabattement de l'entité State dans l'entité Region.

L'opération de rabattement est poursuivie jusqu'au niveau le plus bas de la hiérarchie et que le schéma ne soit formé que d'une seule table, Sale Item dans cet exemple.

Regroupement

L'opération de regroupement peut être appliquée à une entité de transaction afin de créer une nouvelle entité qui contient des données récapitulatives. Le regroupement sera effectué sur un sous-ensemble d'attributs qui seront agrégés en fonction d'autres attributs. La clé de cette nouvelle entité est la concaténation des attributs utilisés pour faire le regroupement. Par exemple, l'opérateur de regroupement peut être appliqué à l'entité Sale Item afin de créer une nouvelle entité intitulée Product Summary (Figure 5.5). Cette nouvelle entité montre le montant des ventes, la quantité moyenne commandée et le prix moyen de chaque produit, calculés sur une base journalière. Les attributs de regroupement sont la quantité et le prix et les attributs sur lesquels le regroupement est effectué sont le Produit ID et la date. La clé de cette nouvelle entité est la concaténation des attributs utilisés pour faire le regroupement. Il faut remarquer que le regroupement entraîne une perte d'information et qu'il n'est pas possible de reconstruire les détails de l'entité de départ à partir de la nouvelle entité.

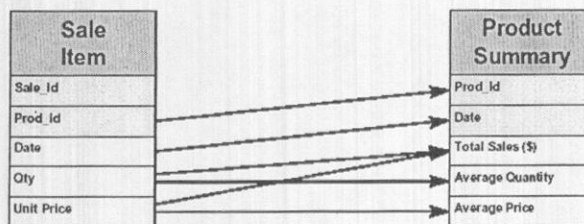


Figure 5.5 : Opération de regroupement.

Pour Moody&Kortink, le modèle multidimensionnel peut être produit à partir du schéma entité-relation sous la forme suivante :

- Schéma plat (Flat schema),
- Schéma en terrasse,
- Schéma étoile,
- Schéma en constellation,
- Schéma nuage d'étoile (star cluster).

Chacune de ces options représente différents compromis entre la complexité et la redondance. Seul le schéma « star cluster » sera décrit. Moody&Kortink définissent le « star cluster » comme un schéma intermédiaire entre le schéma étoile et le schéma de constellation. Rappelons que le schéma de constellation est un schéma étoile dans lequel les dimensions ne sont pas regroupées mais où les hiérarchies sont représentées explicitement. Ce schéma contient un grand nombre de tables ce qui nuit aux

performances du système d'analyse (Kimball, 1996). Par contre, ce schéma est beaucoup plus compréhensible pour l'utilisateur et évite les problèmes de redondance dans les dimensions.

Moody&Kortink considèrent que ni le schéma étoile ni le schéma constellation ne sont idéal et proposent le schéma « star cluster ». Dans ce schéma, les dimensions sont orthogonales et ne se recoupent pas. Le recoupement entre les dimensions peut être identifié à partir des embranchements au sein des hiérarchies. Cet embranchement apparaît quand une entité est « parent » de deux hiérarchies dimensionnelles. Cette entité et tous ses ancêtres sont alors rabattues en deux tables distinctes.

La procédure de création du schéma « star cluster » est la suivante:

- Une table de fait est formée pour chaque entité de transaction, la clé de la table est la concaténation des clés de chaque entité composante,
- Les entités de classification sont rabattues le long de leurs hiérarchies jusqu'à la rencontre d'un embranchement ou une entité composante. Si un embranchement est atteint, une table de sous-dimension est créée. Cette table de sous-dimension sera formée de l'entité du recoupement et de tous ses ancêtres. Le rabattement sera effectué à nouveau après l'embranchement. La table de dimension est créée quand on atteint l'entité composante.
- Lorsqu'il existe une relation hiérarchique entre les entités de transaction, les "enfants" héritent de toutes les dimensions et les attributs clés de l'entité "parent".
- Les attributs numériques des entités de transaction sont regroupés en fonction des dimensions (attributs clés).

La Figure 5.6 montre le schéma "star cluster" créé en appliquant la procédure au schéma entité-relation de l'exemple. Un embranchement apparaît au niveau de l'entité « Region » qui est « parent » des entités « Location » et de « Customer ». Dans un schéma étoile, les deux entités composantes « Location » et « Customer » contiendraient chacune les attributs de « Region » et de « State ». Il en résulterait un recoupement entre les deux dimensions. Le schéma « star cluster » est formé de manière à minimiser le nombre de tables tout en évitant ce genre de recoupement.

Application à Tarigomi

Dans le cadre de l'application de la méthodologie au cas de Tarigomi, nous avons créé une étoile (ou un « star cluster ») par entité de transaction. Chaque entité de transaction est représentée en gris foncé sur les graphiques, les entités de composition en gris clair et les entités de classification en blanc.

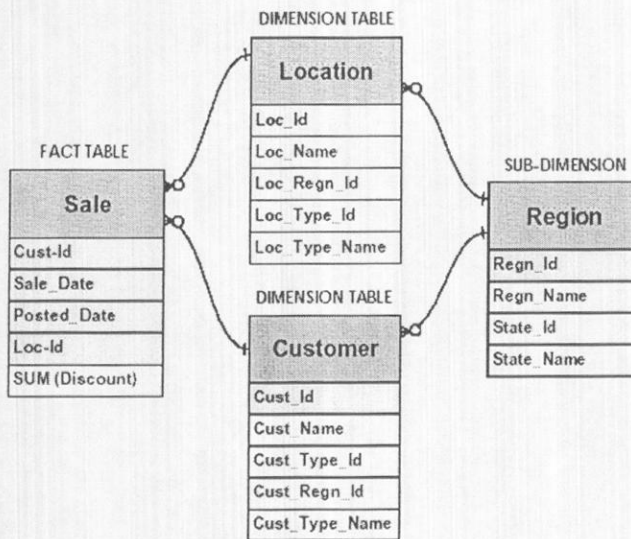


Figure 5.6 : Schéma « star cluster » créé à partir du schéma entité-relation.

Le classement des entités a fourni six entités de transaction qui définissent chacune une table de faits et qui sont placées au centre d'une étoile. La clé de ces tables de faits est la concaténation des clés des dimensions. Le schéma de Tarigomi développé en appliquant la démarche de Moody&Kortink ne contient pas d'entité composante qui décrive une commande ou une ligne de commande. Il en résulte que les étoiles créées ne contiennent pas de dimension de commande et les clés des faits commande et de commande ne contiennent donc pas d'information sur le numéro de commande. La granularité de ces tables de faits ne correspond ainsi pas au numéro de commande et les mesures sont regroupées par jour. On verra que les requêtes qui permettent de faire l'analyse au niveau des commandes ne seront donc pas possibles.

L'opération de rabattement a été utilisée dans le cas de l'entité Entrepôt et tous les attributs de cette entité ont été rajoutés à Magasin.

Étoile commande

La première entité de transaction est l'entité COMMANDE qui est au centre de l'étoile commande.

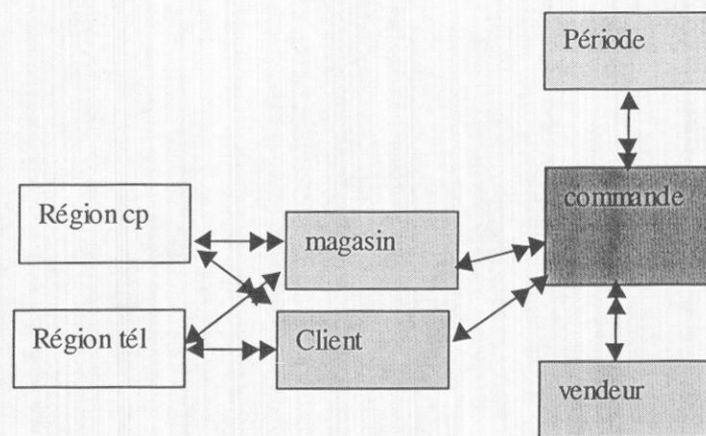


Figure 5.7 : Modèle conceptuel de l'étoile COMMANDE selon Moody et Kortink.

Pour les besoins de l'analyse, seul le délai de livraison est calculé, les autres délais n'ajoutant rien aux concepts que nous illustrons.

Étoile det commande

L'étoile DET COMMANDE est construite à partir de l'entité de transaction DET_COMMANDE de Tarigomi qui est reliée à l'entité de transaction COMMANDE par une relation de 1 à N. Cette relation implique que l'étoile DET COMMANDE hérite toutes les dimensions de l'étoile COMMANDE. Le numéro de la commande n'est pas mémorisé et la granularité correspond à la somme des lignes de commande passées par un client, par un magasin auprès d'un vendeur pendant une journée. On a rajouté un attribut pour indiquer le nombre de lignes de commande qui ont été regroupées.

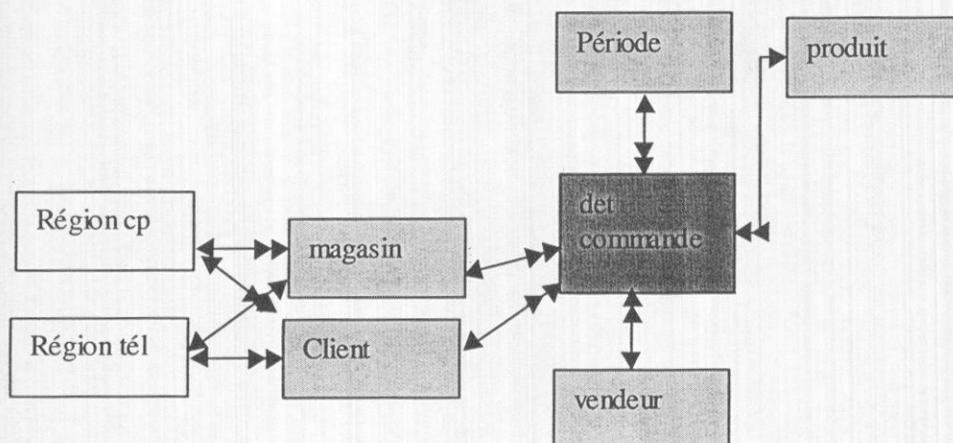


Figure 5.8 : Modèle conceptuel de l'étoile DET COMMANDE selon Moody et Kortink.

Pratiquement, on a créé une vue intitulée MK_LI_COMMANDE qui a été calculée sur la table DET_COMMANDE et dans laquelle on a importé les clés des dimensions de la table COMMANDE. On a supprimé le NO_COMMANDE et on a agrégé les dimensions sur les dimensions de la vue. La nouvelle vue a donc une granularité différente de la table DET_COMMANDE dans la mesure où elle ne correspond plus à chaque commande particulière mais à l'ensemble des commandes qui ont été passées le même jour pour un produit donné, par un client, pour un magasin et par l'intermédiaire d'un vendeur. La vue comprend aussi un champ calculé qui indique le nombre de lignes qui ont été groupées en fonction des clés des dimensions.

Les modèles conceptuels des étoiles Transfert, Stock, Approvisionnement et Det Appro sont présentés sur les figures intitulées Figure 5.9, Figure 5.10, Figure 5.11 et Figure 5.12. On remarque que l'entité Entrepôt qui avait été rabattue dans Magasin fait partie maintenant des dimensions des étoiles Transfert et Stock. Les tables de fait qui sont situées au centre des étoiles ne peuvent pas être créées dans la mesure où les informations n'ont pas été enregistrées dans Tarigomi et ne sont donc pas disponibles.

Étoile Transfert

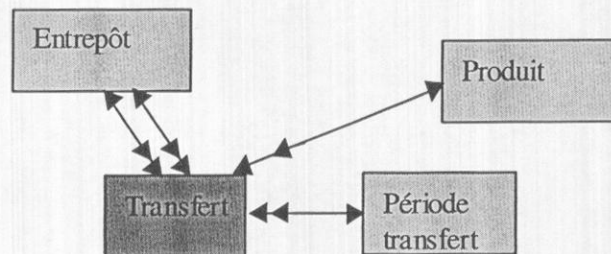


Figure 5.9 : Modèle conceptuel de l'étoile TRANSFERT selon Moody et Kortink.

Étoile stock

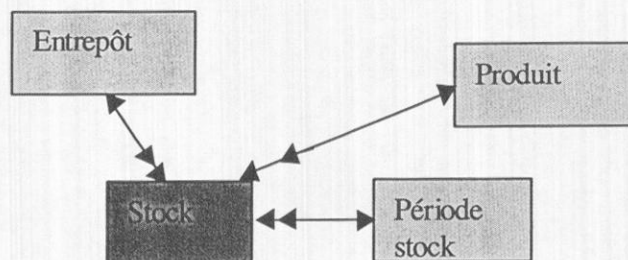


Figure 5.10 : Modèle conceptuel de Étoile Stock selon Moody et Kortink.

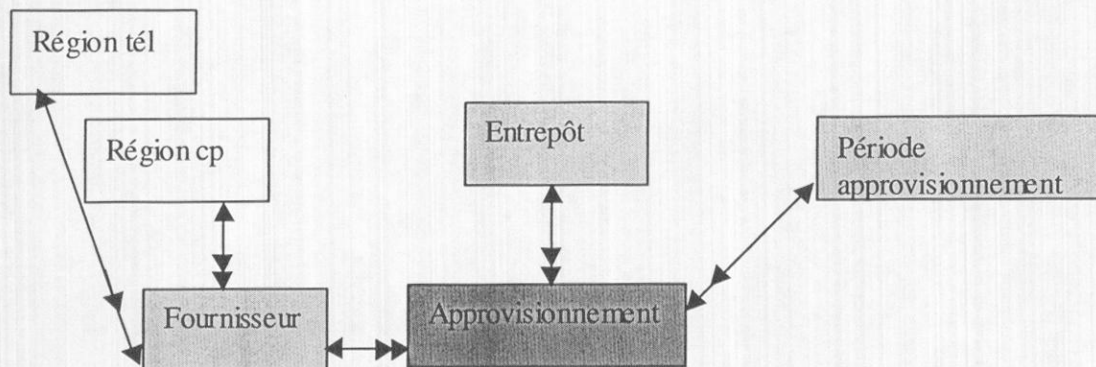
Étoile Approvisionnement

Figure 5.11 : Modèle conceptuel de l'étoile Approvisionnement selon Moody et Kortink.

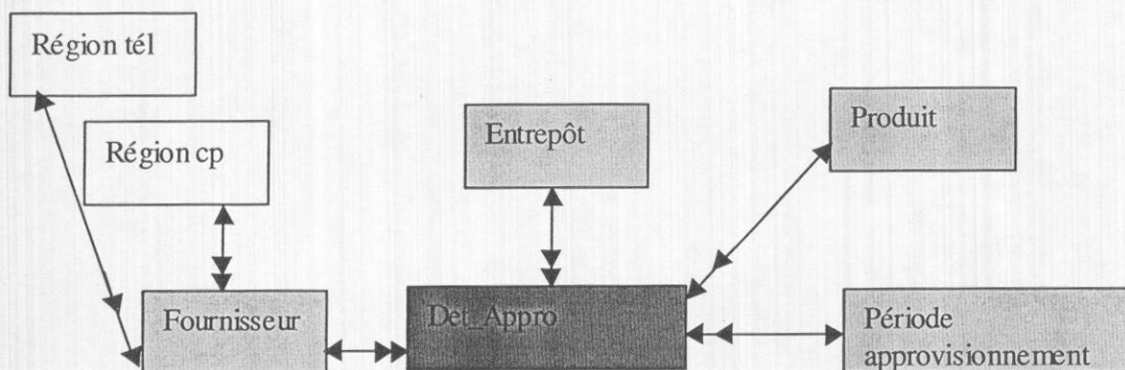
Étoile Det Appro

Figure 5.12 : Modèle conceptuel de l'étoile Det Appro selon Moody et Kortink.

Dimensions

Les dimensions des modèles étoiles sont créées à partir des entités composantes qui ont été inventoriées sur le schéma entité-relation. Dans le cas de Tarigomi, nous en avons recensé sept dont six ont été construites à partir des tables d'origine. Les tables correspondant aux entités Entrepôt et Fournisseur n'ont pas pu être créées dans la mesure où les données de Tarigomi ne contiennent pas les informations qui sont nécessaires.

Les tables de fait du modèle multidimensionnel avec leurs attributs sont représentées sur la Figure 5.13 et celles des dimensions sur la Figure 5.14.

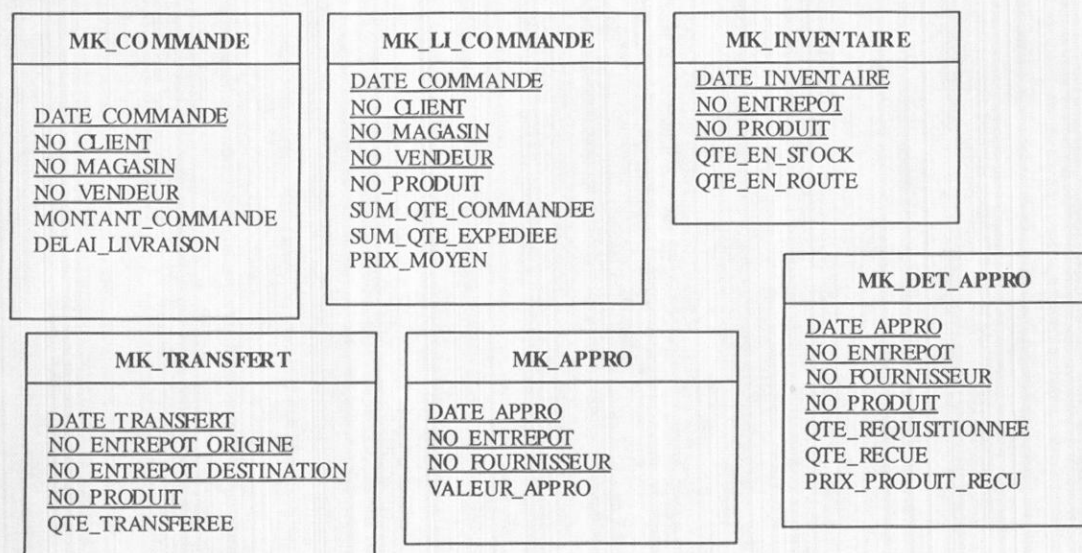


Figure 5.13 : Tables de fait du modèle multidimensionnel selon Moody et Kortink.

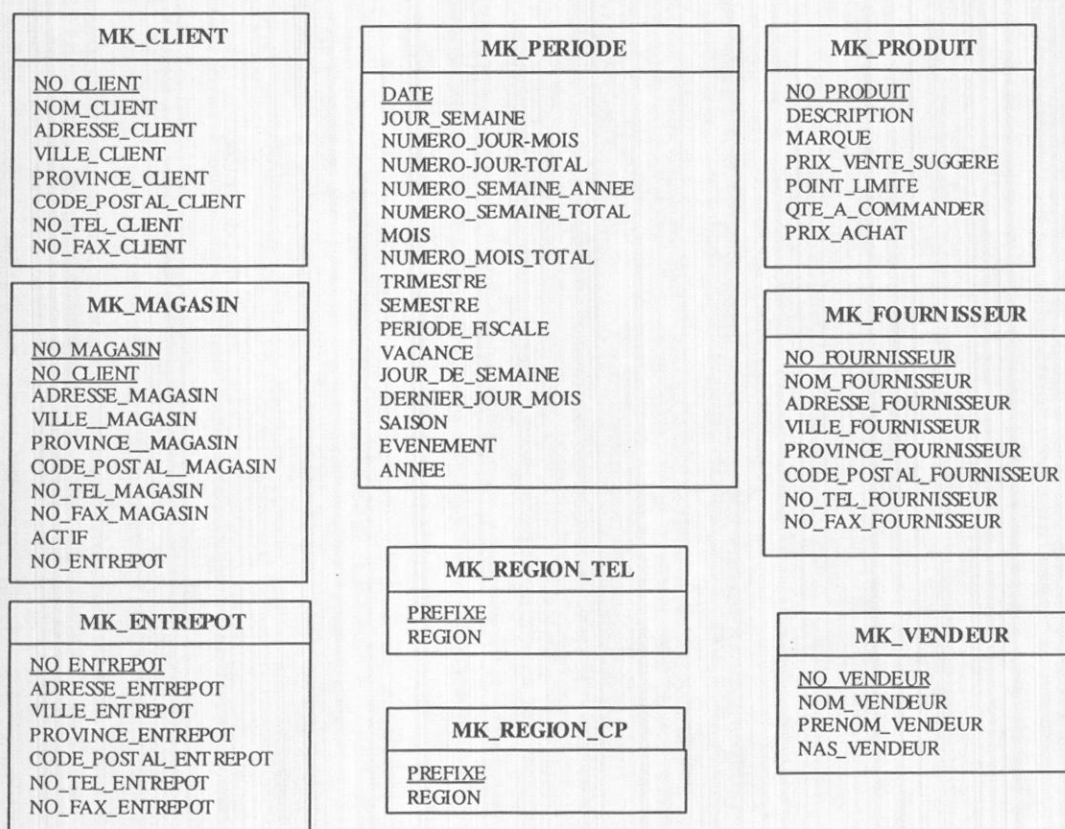


Figure 5.14 : Dimensions du modèle multidimensionnel construit selon Moody et Kortink.

5.1.5. Évaluation et raffinement

La création d'un modèle multidimensionnel est un processus itératif et la procédure présentée plus haut ne permet que d'obtenir une première version du schéma. Les raffinements et les modifications ultérieures permettront surtout de simplifier le modèle et de gérer les aspects non hiérarchiques.

Combinaison des tables de fait

Les tables de fait qui ont les mêmes dimensions, et donc les mêmes clés, seront combinées pour réduire le nombre de schéma étoile et pour faciliter la comparaison entre les mesures.

Combinaison des tables de dimension

La création d'une table de dimension pour chaque entité composante du modèle entité-relation engendre un grand nombre de tables. Il sera nécessaire de simplifier le modèle de mini-entrepôt et de combiner ensemble plusieurs dimensions.

Relation N à M

La plupart des difficultés lorsque l'on convertit un modèle entité-relation en un modèle multidimensionnel, proviennent de la présence de relation N à M entre les entités (présence d'entité intersection). Ces relations posent un problème parce qu'elles entraînent une rupture de la chaîne hiérarchique et les entités de niveau supérieur ne peuvent pas être rabattues. Pour résoudre ce problème, on peut :

- Ignorer les relations N à M,
- Convertir les relations N à M en relation 1 à N en définissant une relation dite primaire,
- Inclure les relations N à M dans le mini-entrepôt, ce qui pourra être utile pour les analystes expérimentés mais qui ne pourra pas être utilisé dans les outils OLAP.

Gestion des sous-types

Les super entités et les sous-types peuvent être facilement convertis en structure hiérarchique en créant une entité de classification pour distinguer les éléments du sous-type.

5.1.6. Analyse par requêtes SQL

On trouvera à l'annexe 9 le texte complet des requêtes SQL qui ont été utilisées pour effectuer l'analyse de Tarigomi. Il faut remarquer que les requêtes 2, 3 et 4 sont impossibles parce que le numéro de commande ne fait pas partie des dimensions et que la granularité de MK_COMMANDE et de

MK_LI_COMMANDE n'est pas assez grande pour fournir des réponses à ce niveau de détail. Le Tableau 5.1 présente une synthèse des regroupements, des mesures analysées et des vues qui ont dû être créées pour effectuer les analyses.

Requête	Regroupement	Mesure	Création de vues
Requête 1	NO_CLIENT, MOIS, ANNEE	Moyenne annuelle des montants mensuels	Oui car moyenne n'est pas calculée sur le regroupement
Requête 2	NO_VENDEUR, NO_CLIENT, MOIS, ANNEE	Nombre de commandes, Montant total	Requête impossible
Requête 3	NO_CLIENT, MOIS, ANNEE	Délai moyen	Requête impossible
Requête 4	NO_VENDEUR, MOIS	Nombre moyen de lignes de commande par commande	Requête impossible
Requête 5	MARQUE, JOUR_DE_SEMAINE	Montant commandé / valeur totale	Non
Requête 6	NO_VENDEUR, MARQUE, MOIS, ANNEE	Rabais moyen / montant de la commande	Non
Requête 7	NO_CLIENT, MARQUE, ANNEE	Montant total	Non
Requête 8	NO_PRODUIT, MARQUE, MOIS, ANNEE	Prix de vente moyen	Non

Tableau 5.1 : Analyse du modèle multidimensionnel effectué au moyen de requêtes SQL.

5.1.7. Analyse par Discoverer

Les analyses par Discoverer nécessitent que l'administrateur de Discoverer crée des « complex folders », des champs calculés et effectue des jointures entre les différentes tables. Le Tableau 5.2 fournit la liste des éléments qu'il faut créer dans Discoverer Administrator.

Création de « complex folder »	Analyse commande MK Analyse commande mensuelle MK Analyse ligne commande MK	
Champ calculé	MK Li Commande	MK Montant Ligne
	Analyse commande MK	MK Nb Lignes
	Analyse commande mensuelle MK	MK Montant Mensuel
	Analyse ligne commande MK	MK Rabais
Jointures	MK Commande	MK Client MK Magasin MK Vendeur MK Periode

	MK Li Commande	MK Client MK Magasin MK Vendeur MK Periode MK Produit
	Analyse commande MK	MK Client MK Magasin MK Vendeur MK Periode
	Analyse ligne commande MK	MK Client MK Magasin MK Vendeur MK Periode MK Produit

Tableau 5.2 : « Complex folders », champs calculés et jointures à créer dans Discoverer Administrator pour le modèle multidimensionnel construit en suivant la démarche de Moody et Kortink.

5.2. Hüsemann

La démarche proposée par Hüsemann et al. (2000) part de la structure du schéma entité-relation et d'une description des attributs en fonction des analyses qui seront effectuée pour déterminer les mesures. Les dimensions sont ensuite déterminées et finalement les faits sont constitués en créant des tables qui réunissent les mesures qui dépendent des mêmes dimensions. Hüsemann propose aussi une méthode graphique pour représenter le modèle multidimensionnel.

Dans la méthodologie de Hüsemann, l'ordre de détermination des éléments du modèle multidimensionnel est donc :

Mesure $\Rightarrow$ dimension $\Rightarrow$ fait (MDF)

Pour Hüsemann, il faut concevoir une méthodologie de conception qui prenne en compte toutes les étapes, pas uniquement celle de la modélisation des données. Selon eux, l'intérêt de leur démarche est d'établir une construction algorithmique du modèle dimensionnel à partir de la définition des mesures et a pour avantage d'assurer que le modèle proposé est conforme aux prescriptions imposées par les formes normales multidimensionnelles.

Les bases de données opérationnelles sont le point de départ du design conceptuel et Hüsemann considère qu'un schéma entité-relation existe.

5.2.1. Terminologie et notation

Hüsemann définit une terminologie et une notation graphique pour la modélisation multidimensionnelle conceptuelle qui est présentée par la suite.

5.2.1.1. _____ Structures multidimensionnelles de base

Un fait représente un élément atomique de la base de donnée multidimensionnelle. Il est constitué de valeurs quantitatives appelées mesures et d'un contexte qui les qualifie et qui est déterminé par des niveaux de dimension. Chaque niveau de dimension est composé d'un certain nombre d'éléments. Un chemin de regroupement est formé par une suite de niveaux de dimension qui débute par un niveau de base et se termine par le niveau implicite composé du seul élément « Tout ». Une dimension est composée d'un ou plusieurs chemins de regroupement qui partagent le même niveau de dimension terminal. Par exemple, la dimension Temps pourrait contenir deux chemins de regroupement qui débutent tous par Jour et qui se terminent par Année. Un des chemins sera composé des niveaux suivants : Jour $\Rightarrow$ Mois $\Rightarrow$ Semestre $\Rightarrow$ Année, l'autre des niveaux : Jour $\Rightarrow$ Semaine $\Rightarrow$ Année.

Un attribut de propriété décrit une information supplémentaire reliée à un niveau de dimension. Par exemple, l'attribut de propriété Nom Client est associé au niveau de dimension ID Client. L'attribut de propriété pourra être optionnel ou obligatoire. Il pourra être utilisé pour délimiter des catégories mais ne définira pas de regroupements.

Un schéma de fait représente le contexte dimensionnel d'une série de faits qui partagent les mêmes niveaux terminaux de dimension.

5.2.1.2. _____ Hiérarchies de dimension

Les hiérarchies de dimension peuvent être soit simples et n'être formées que d'un seul chemin de regroupement, soit multiple et être formées d'au moins deux chemins différents. Un groupe de chemins est dit alternatif si tous les éléments d'un niveau de dimension inférieur n'appartiennent qu'à un seul chemin et que les chemins de regroupement sont mutuellement exclusifs. Par contre, un groupe de chemins est dit optionnel si un élément peut faire partie de plusieurs chemins.

5.2.1.3. _____ Notation graphique

Les hiérarchies simples et les groupes de chemins alternatifs sont représentés graphiquement par une flèche simple alors que les chemins optionnels sont dessinés avec une flèche double. Les flèches pointent

du niveau d'agrégation inférieur au niveau supérieur. Un attribut de propriété obligatoire est connecté au niveau de dimension par un losange, ceux qui sont optionnels sont connectés par une ligne simple.

5.2.2. Démarche de conception

Hüsemann propose une méthode de conception d'un entrepôt de données organisée de manière proche de celle qui est utilisée pour les bases de données opérationnelles. Elle est composée des quatre phases suivantes :

- Analyse des besoins,
- Design conceptuel,
- Design logique,
- Design physique.

La partie qui nous intéresse est celle qui concerne le design conceptuel. Il est intéressant toutefois de noter que cette partie s'inscrit dans une démarche globale qui ne concerne pas seulement la partie conceptuelle mais aussi les parties logique et physique.

Avant d'effectuer le design conceptuel, les schémas des bases de données opérationnelles sont analysés et un groupe d'experts du domaine des affaires choisit les attributs pertinents et spécifie si ces attributs sont utilisés comme mesure ou comme dimension. Cette analyse aboutit à l'établissement d'un tableau des attributs qui fait la synthèse de toutes les propriétés des attributs pertinents pour l'analyse. Au cours de cette analyse, on décidera si les attributs sont optionnels ou non. De plus, l'analyse pourra proposer de rajouter des attributs pour répondre à des requêtes complexes.

Dans le cas de Tarigomi, le rôle des attributs du schéma entité-relation est représenté sur le Tableau 5.3. Les attributs qui ne sont pas des mesures sont forcément des dimensions. Dans ce tableau, on a aussi indiqué pour chaque attribut s'il est optionnel ou non.

Attribut	Mesure	Optionnel
NO_COMMANDE	non	oui
DATE_COMMANDE	non	oui
NOM_RECEVEUR	non	non
DELAI_LIVRAISON	oui	oui
NO_VENDEUR	non	oui
NOM	non	oui
PRENOM	non	non
NAS	non	non

QTE_COMMANDEE	oui	oui
QTE_EXPEDIEE	oui	oui
PRIX_VENTE_EFFECTIF	oui	oui
NO_MAGASIN	non	oui
NO_CLIENT	non	oui
ADRESSE	non	oui
VILLE	non	non
PROVINCE	non	non
CODE_POSTAL	non	oui
NO_TEL	non	oui
NO_FAX	non	oui
ACTIF	non	non
PREFIXE_CODE_POSTAL	non	oui
REGION	non	oui
PREFIXE_NO_TEL	non	oui
REGION	non	oui
NO_PRODUIT	non	oui
DESCRIPTION	non	oui
MARQUE	non	oui
PRIX_VENTE_SUGGERE	non	non
POINT_LIMITE	non	non
QTE_A_COMMANDER	non	non
PRIX_ACHAT	non	non
NO_CLIENT	non	oui
NOM	non	oui
ADRESSE	non	non
VILLE	non	non
PROVINCE	non	non
CODE_POSTAL	non	non
NO_TEL	non	non
NO_FAX	non	non
NO_ENTREPOT	non	oui
ADRESSE	non	oui
VILLE	non	non
PROVINCE	non	non
CODE_POSTAL	non	non
NO_TEL	non	non
NO_FAX	non	non

QTE_EN_STOCK	oui	oui
QTE_EN_ROUTE	oui	oui
NO_CASIER	non	non
DATE_INVENTAIRE	non	oui
NO_TRANSFERT	non	oui
NO_ENTREPOT_ORIGINE	non	oui
NO_ENTREPOT_DESTINATION	non	oui
QTE_TRANSFEE	oui	oui
DATE_TRANSFERT	non	non
NO_APPRO	non	oui
NO_FOURNISSEUR	non	non
NO_ENTREPOT	non	
DATE_REQUISITION	non	oui
DELAI_APPRO	oui	oui
NO_APPRO	non	oui
NO_PRODUIT	non	oui
QTE_REQUISITIONNEE	oui	oui
QTE_RECUE	oui	oui
PRIX_APPRO	oui	oui
NO_FOURNISSEUR	non	oui
NOM	non	oui
ADRESSE_FOURNISSEUR	non	non
VILLE	non	non
PROVINCE	non	non
CODE_POSTAL	non	non
NO_TEL	non	non
NO_FAX	non	non

Tableau 5.3 : Spécification du rôle des attributs du schéma entité-relation.

À partir des schémas des bases de données opérationnelles et du tableau d'analyse du rôle des attributs sélectionnés par une équipe d'experts du business, la démarche du design conceptuel est la suivante :

- Définition du contexte des mesures : détermination des dépendances fonctionnelles de chaque mesure,
- Définition des dimensions terminales qui sont les racines des dimensions,
- Détermination d'une clé minimale pour chaque mesure,
- Détermination de l'ensemble des dépendances fonctionnelles pour les mesures

- Regroupement des mesures qui ont les mêmes dépendances fonctionnelles.
- Design de la hiérarchie dimensionnelle,
- Détermination de toutes les dépendances fonctionnelles entre les niveaux qui appartiennent à une dimension,
- Distinction entre les attributs de propriétés et les niveaux des dimensions
- Définition des contraintes de récapitulation.

5.2.2.1. _____ Définition du contexte des mesures

La première phase du design conceptuel consiste à déterminer les dépendances fonctionnelles entre les dimensions et les mesures. Tout d'abord, on détermine un ensemble minimal de dimensions pour chaque mesure. Ensuite on établit l'ensemble des dépendances fonctionnelles entre les mesures et les dimensions de manière à ce que chaque dimension détermine une ou des mesures mais n'est pas dépendante fonctionnellement d'autres dimensions. Ces dimensions qualifient ainsi les niveaux terminaux qui sont utilisés comme élément racine des hiérarchies de dimension. À chaque élément racine correspond une dimension.

De plus, toutes les mesures qui dépendent fonctionnellement des mêmes dimensions sont regroupées dans le même fait puisqu'ils partagent les mêmes dimensions.

Application à Tarigomi

Les dépendances fonctionnelles des mesures sont les suivantes:

DF1 : (NO_COMMANDE, DATE_COMMANDE) $\Rightarrow$ DELAI_LIVRAISON

DF2 : (NO_COMMANDE, NO_PRODUIT, DATE_COMMANDE) $\Rightarrow$ QTE_COMMANDEE

DF3 : (NO_COMMANDE, NO_PRODUIT, DATE_COMMANDE) $\Rightarrow$ QTE_EXPEDIEE

DF4 : (NO_COMMANDE, NO_PRODUIT, DATE_COMMANDE) $\Rightarrow$ PRIX_VENTE_EFFECTIF

DF5 : (NO_PRODUIT, NO_ENTREPOT, DATE_INVENTAIRE) $\Rightarrow$ QTE_EN_STOCK

DF6 : (NO_PRODUIT, NO_ENTREPOT, DATE_INVENTAIRE) $\Rightarrow$ QTE_EN_ROUTE

DF7 : (NO_TRANSFERT, DATE_TRANSFERT) $\Rightarrow$ QTE_TRANSFEEE

DF8 : (NO_APPRO, DATE_REQUISITION) $\Rightarrow$ DELAI_APPRO

DF9 : (NO_APPRO, DATE_REQUISITION, NO_PRODUIT) $\Rightarrow$ QTE_REQUISITIONNEE

DF10 : (NO_APPRO, DATE_REQUISITION, NO_PRODUIT) $\Rightarrow$ QTE_RECUE

DF11 : (NO_APPRO, DATE_REQUISITION, NO_PRODUIT) $\Rightarrow$ PRIX_APPRO

Regroupement de mesures

Appelons HU_COMMANDE le fait correspondant à la mesure de délai de livraison dont le contexte est décrit par la dépendance fonctionnelle DF1. Les dépendances fonctionnelles DF2, DF3 et DF4 sont identiques. Il en résulte que les trois mesures qui concernent les lignes de commande sont regroupées dans le même fait « HU_LI_COMMANDE ». Les deux mesures qui concernent la gestion de l'inventaire ont des dépendances fonctionnelles identiques (DF5 et DF6) et sont donc regroupées dans le même fait « HU_INVENTAIRE ». La mesure qui concerne la quantité transférée (DF7) fait partie de la dimension « HU_TRANSFERT » et celle qui concerne le délai de l'approvisionnement (DF8) constitue la mesure du fait « HU_APPRO ». Finalement, les trois mesures qui concernent le détail des approvisionnements ont des dépendances fonctionnelles identiques (DF9, DF10 et DF11) et sont regroupées dans le fait « HU_DET_APPRO ».

La représentation graphique des modèles multidimensionnels utilise la notation graphique du paragraphe 5.2.1.3.

Fait HU_COMMANDE

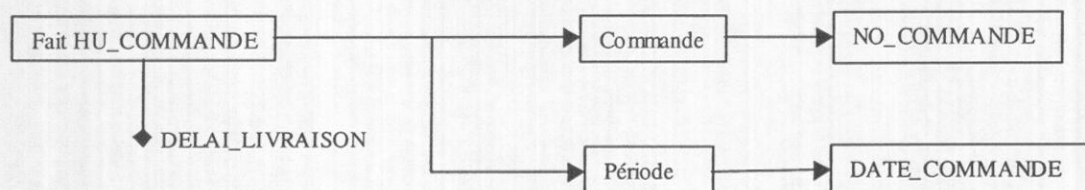


Figure 5.15 : Modèle multidimensionnel du fait HU_COMMANDE selon Hüsemann.

Fait HU_LI_COMMANDE

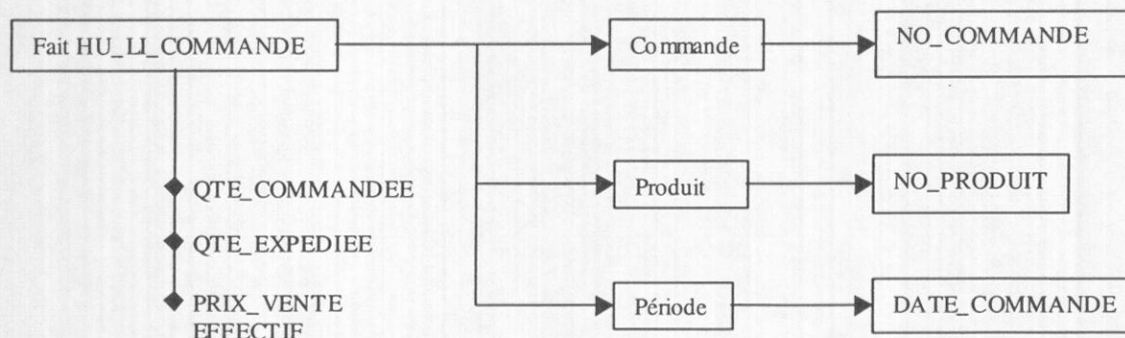


Figure 5.16 : Modèle multidimensionnel du fait HU_LI_COMMANDE selon Hüsemann.

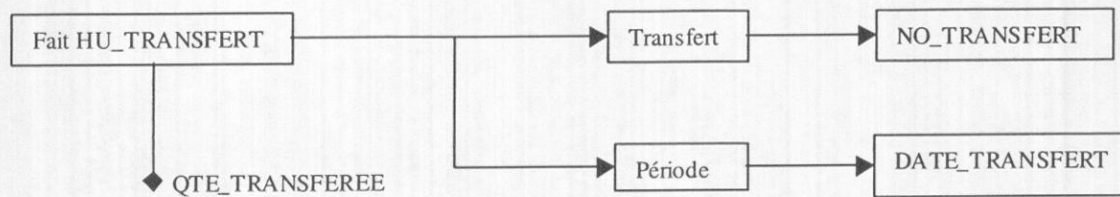
Fait HU_TRANSFERT

Figure 5.17 : Modèle multidimensionnel du fait HU_TRANSFERT selon Hüsemann.

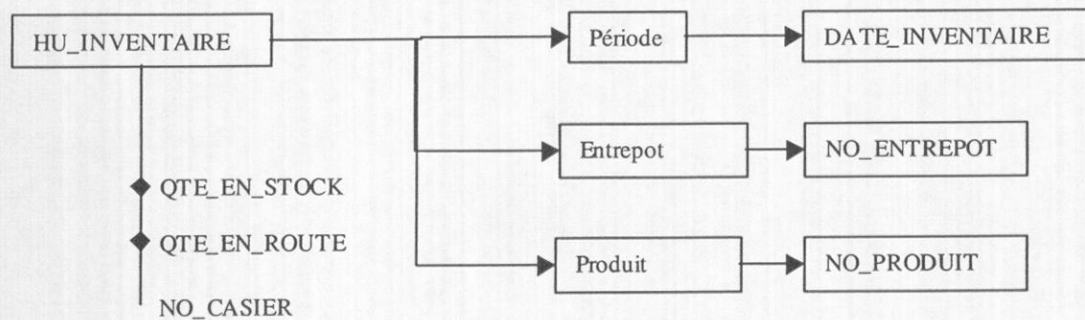
Fait HU_INVENTAIRE

Figure 5.18 : Modèle multidimensionnel du fait HU_INVENTAIRE selon Hüsemann.

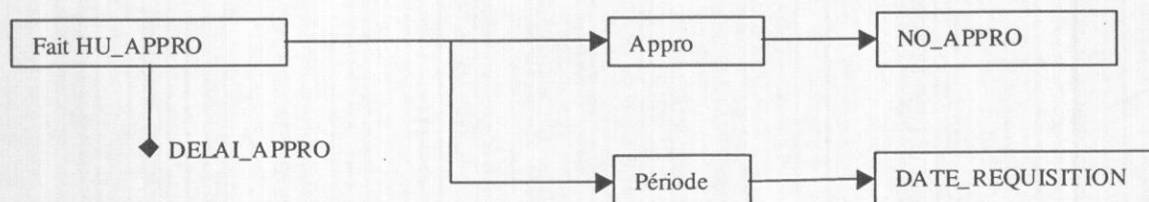
Fait HU_APPRO

Figure 5.19 : Modèle multidimensionnel du fait HU_APPRO selon Hüsemann.

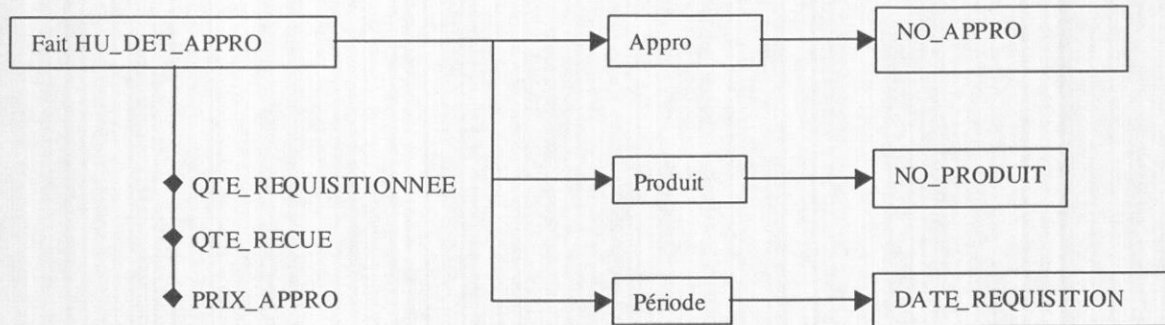
Fait HU_DET_APPRO

Figure 5.20 : Modèle multidimensionnel du fait HU_DET_APPRO selon Hüsemann.

5.2.2.2. ____ Design des hiérarchies dimensionnelles

Au cours de l'étape suivante, on détermine les hiérarchies de chacune des dimensions. Pour ce faire, on détermine toutes les dépendances fonctionnelles entre les niveaux qui appartiennent à la même dimension.

Tout d'abord, il convient de distinguer entre les attributs de propriété et les niveaux de dimension. Les attributs de propriété peuvent seulement être utilisés pour faire des sélections mais ne peuvent pas être utilisés pour des regroupements. Par la suite, la seconde étape consiste à obtenir une approximation grossière de la hiérarchie en construisant un graphe dirigé dont les nœuds sont des niveaux de dimension et dont les arêtes représentent des dépendances fonctionnelles.

Le graphe est ensuite complété avec les attributs de propriété qui sont rattachés au niveau de dimension dont ils dépendent fonctionnellement. De plus, on indique sur le graphe si les attributs de propriété sont optionnels ou obligatoires.

Finalement, les hiérarchies doivent être examinées pour vérifier s'il existe des groupes de regroupement qui sont optionnels ou non. Si tous les niveaux de dimension doivent être obligatoirement définis, alors tous les chemins de regroupement sont alternatifs. Par contre, si certains niveaux de dimension sont optionnels les chemins de regroupement sont optionnels.

Application à Tarigomi

La hiérarchie dimensionnelle est développée tout d'abord en déterminant les dépendances fonctionnelles des attributs du Tableau 5.3.

NO_COMMANDE $\Rightarrow$ NO_VENDEUR

NO_COMMANDE $\Rightarrow$ NO_CLIENT
NO_COMMANDE $\Rightarrow$ NO_MAGASIN
NO_CLIENT $\Rightarrow$ CODE_POSTAL,
NO_TEL,
NO_FAX
NO_MAGASIN $\Rightarrow$ CODE_POSTAL,
NO_TEL,
NO_FAX,
NO_ENTREPOT
CODE_POSTAL $\Rightarrow$ REGION_CP,
CODE_POSTAL $\Rightarrow$ VILLE,
VILLE $\Rightarrow$ PROVINCE,
NO_TEL $\Rightarrow$ REGION_TEL
NO_FAX $\Rightarrow$ REGION_TEL
NO_PRODUIT $\Rightarrow$ MARQUE
NO_TRANSFERT $\Rightarrow$ NO_ENTREPOT_ORIGINE
NO_TRANSFERT $\Rightarrow$ NO_ENTREPOT_DESTINATION
NO_TRANSFERT $\Rightarrow$ NO_PRODUIT

Le premier graphe des dimensions est obtenu en faisant correspondre un niveau de dimension à chaque attribut. Nous avons défini une dimension HU_DIM_COMMANDE qui contient le numéro de la commande, ce qui nous permettra de faire des analyses au niveau de chaque commande.

Ensuite, le graphe est complété avec les attributs de propriété qui décrivent des informations additionnelles reliées à un niveau de dimension. Ainsi par exemple, l'adresse d'un client est un attribut de propriété de la dimension client. Pour chaque dimension, on a les attributs de propriété suivants :

Pour la dimension HU_DIM_COMMANDE :

- NO_REQUISITION,
- NOM_RECEVEUR.

Pour la dimension HU_CLIENT :

- NOM,
- ADRESSE,

Pour la dimension HU_MAGASIN :

- ADRESSE,
- ACTIF.

Pour la dimension HU_PRODUIT :

- DESCRIPTION,

- PRIX_VENTE_SUGGERE,
- POINT_LIMITE,
- QTE_A_COMMANDER,
- PRIX_ACHAT,

Pour la dimension HU_VENDEUR :

- NOM_VENDEUR,
- PRENOM_VENDEUR,
- NAS,

Pour la dimension HU_ENTREPOT :

- ADRESSE
- CODE_POSTAL,
- VILLE,
- PROVINCE,
- NO_TEL,
- NO_FAX.

Représentation graphique

Les dimensions peuvent être représentées graphiquement de la manière suivante :

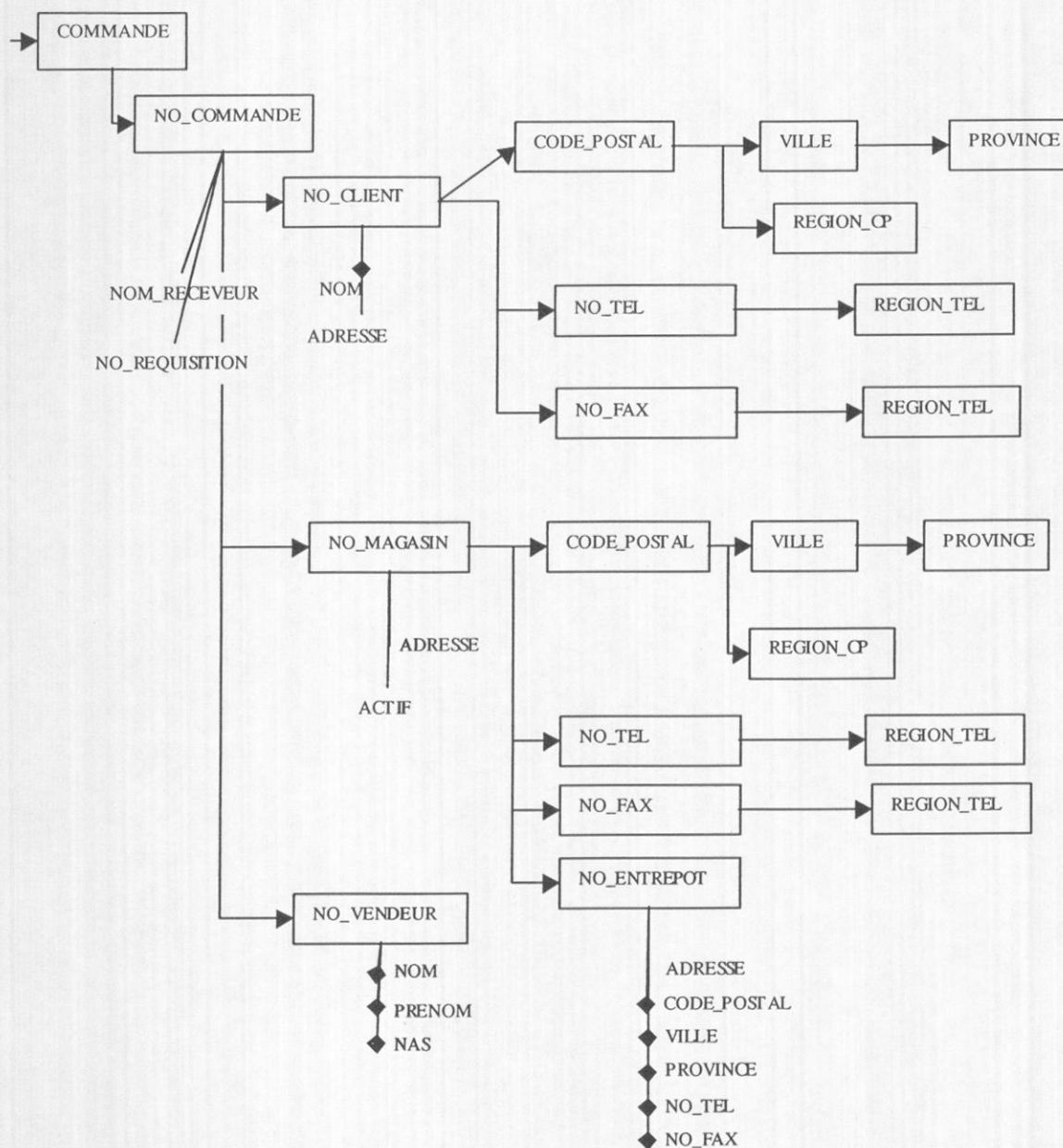


Figure 5.21 : Représentation graphique de la dimension *COMMANDE* construite selon Hüsemann. La notation graphique a été décrite dans le paragraphe 5.2.1.3.

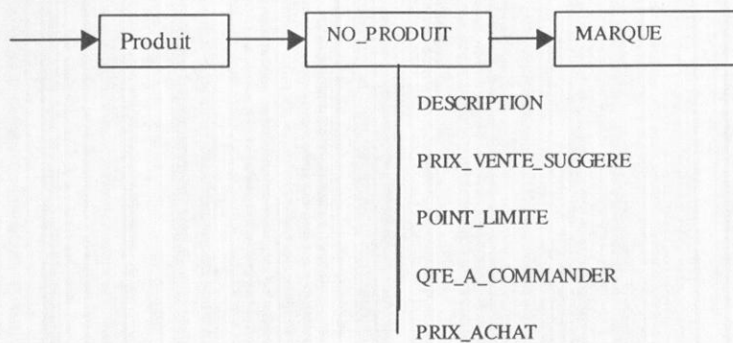
Dimension Produit

Figure 5.22 : Représentation graphique de la dimension *PRODUIT* construite selon Hüsemann.

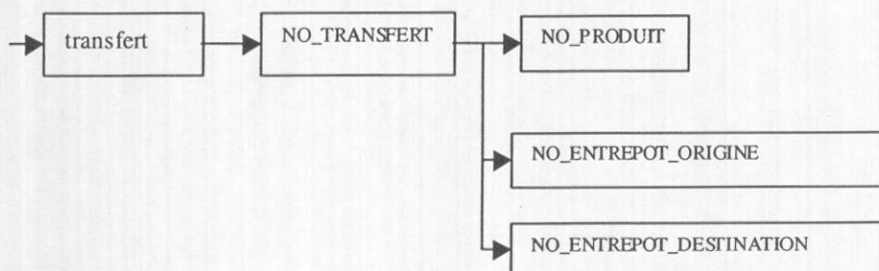
Dimension transfert

Figure 5.23 : Représentation graphique de la dimension *TRANSFERT* construite selon Hüsemann.

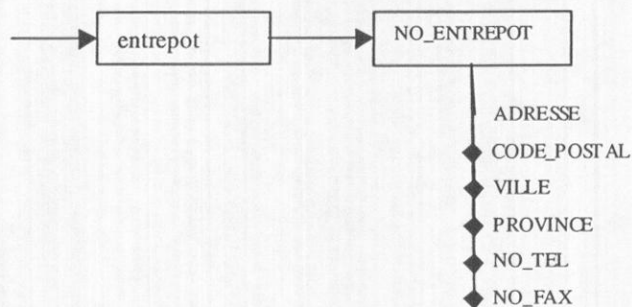
Dimension Entrepôt

Figure 5.24 : Représentation graphique de la dimension *ENTREPOT* construite selon Hüsemann.

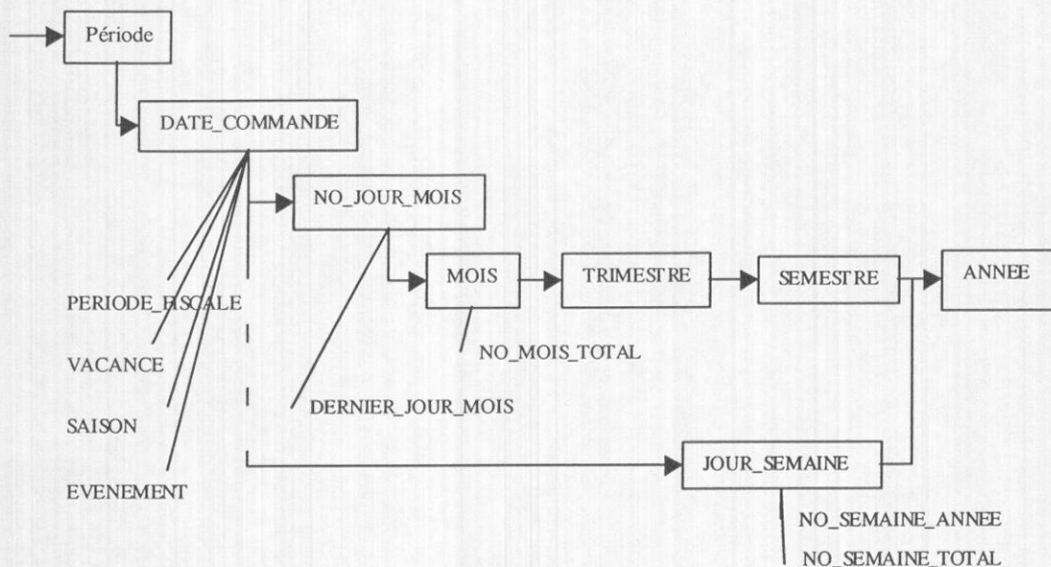
Dimension Période

Figure 5.25 : Représentation graphique de la dimension PERIODE construite selon Hüsemann.

5.2.2.3. ____ Définition des contraintes de récapitulation (summarizability)

Tous les regroupements des mesures ne sont pas possibles pour toutes les dimensions. Par exemple, la somme du solde des comptes d'un groupe de clients de même que le solde moyen a un sens. De même, il est raisonnable de calculer un âge moyen des clients. Par contre, il n'y a pas de sens à calculer la somme des âges d'un groupe de clients. En conséquence, le modèle conceptuel doit fournir une information sur les regroupements possibles des mesures en fonction de chaque dimension. Cette information pourrait être représentée sur le graphique du modèle mais elle alourdirait la présentation qui finirait par ne plus être lisible.

Hüsemann préconise plutôt de représenter ces contraintes de récapitulation sur un tableau à part. Ils définissent quatre degrés de récapitulation possibles qui sont représentés sur le Tableau 5.4. Le niveau 1 correspond au cas où toutes les fonctions de groupement pourraient être utilisées. Le niveau 2 est associé à toutes les fonctions sauf la somme des mesures qui n'a pas de sens (pourcentage par exemple). Le niveau 3 correspond au cas où seul le décompte des mesures ne serait possible. Le niveau 4 est le dernier où aucun regroupement n'est possible proposé par Hüsemann. Il nous semble toutefois que ce niveau ne correspond pas à une situation réelle dans la mesure où il est toujours possible de faire un décompte.

Niveau de restriction	Fonctions de regroupement utilisables
1	{SUM, AVG, MIN, MAX, STDEV, VAR, COUNT}
2	{AVG, MIN, MAX, STDEV, VAR, COUNT}
3	{ COUNT}
4	{ }

Tableau 5.4 : Niveaux de restriction et fonctions de regroupement.

Après avoir construit le graphique du schéma conceptuel, l'étape suivante sera de définir les niveaux de restriction pour toutes les paires {mesure – niveau de dimension}. Les niveaux de restriction ne peuvent pas croître le long d'une hiérarchie et sont définis par défaut de manière identique pour tous les niveaux supérieurs. Ils pourront par contre être plus restrictifs.

Application à Tarigomi

Le Tableau 5.5 contient le niveau de restriction de chaque mesure pour chaque dimension terminale.

Schéma de fait	Mesure	Niveau de dimension	Niveau de restriction
HU_COMMANDE	DELAI_BON	NO_COMMANDE	2
		DATE_COMMANDE	2
	DELAI_LIVRAISON	NO_COMMANDE	2
		DATE_COMMANDE	2
	DELAI_FACTURE	NO_COMMANDE	2
		DATE_COMMANDE	2
	DELAI_PAIEMENT	NO_COMMANDE	2
		DATE_COMMANDE	2
HU_LI_COMMANDE	QTE_COMMANDEE	NO_COMMANDE	1
		NO_PRODUIT	1
		DATE_COMMANDE	2
	QTE_EXPEDIEE	NO_COMMANDE	1
		NO_PRODUIT	1
		DATE_COMMANDE	2
HU_TRANSFERT	QTE_TRANSFEREE	NO_TRANSFERT	1
		DATE_COMMANDE	2
HU_INVENTAIRE	QTE_EN_STOCK	NO_ENTREPOT	1
		NO_PRODUIT	1
		DATE_COMMANDE	2
	QTE_EN_ROUTE	NO_ENTREPOT	1
		NO_PRODUIT	1
		DATE_COMMANDE	2

Tableau 5.5 : Contraintes de récapitulation de chaque mesure en fonction des dimensions.

5.2.3. Production du modèle multidimensionnel

Les Figure 5.26 et Figure 5.27 représentent les tables qui ont été construites à partir des modèles conceptuels. Il faut remarquer que les tables des dimensions HU_CLIENT et HU_MAGASIN contiennent les champs correspondant aux niveaux de hiérarchies qui ont été décrits explicitement dans le paragraphe 5.2.2.2. Nous avons regroupé les champs VILLE et PROVINCE plutôt que de créer des tables dans la mesure où aucune information à part le nom de la ville ou celui de la province n'apparaîtrait dans les tables.

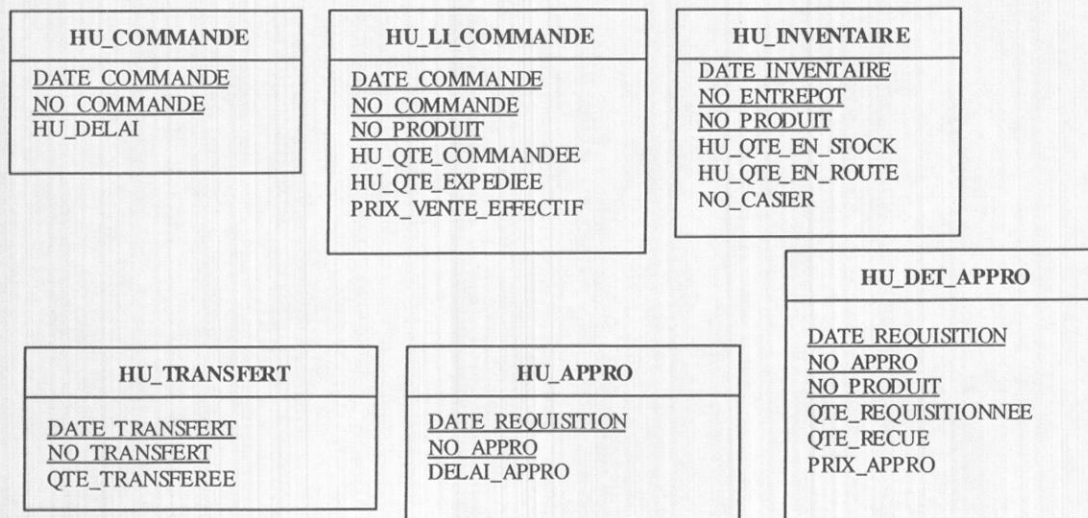


Figure 5.26 : Tables des faits du modèle multidimensionnel selon Hüsemann.

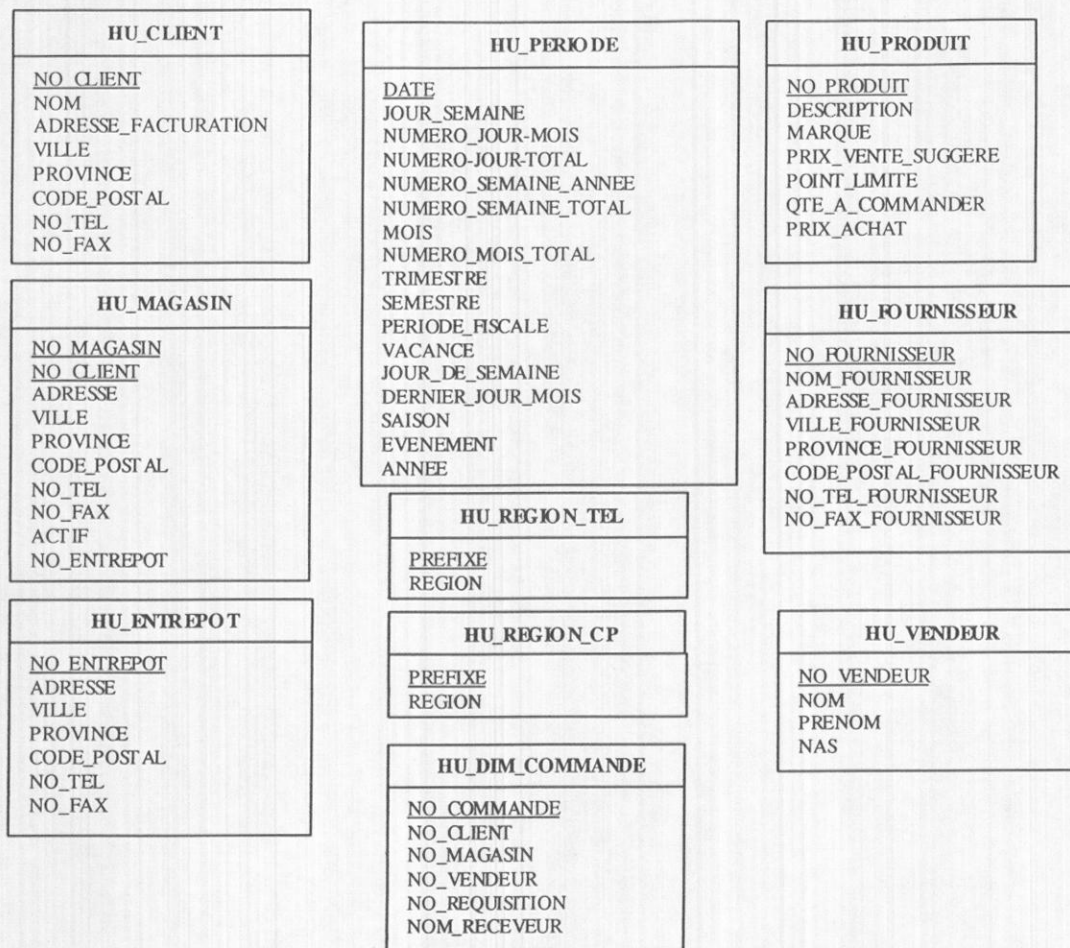


Figure 5.27 : Tables des dimensions du modèle multidimensionnel selon Hüsemann.

5.2.4. Analyse par requêtes SQL

Toutes les requêtes de la liste du tableau 4.1 peuvent être effectuées sur les tables du modèle multidimensionnel construites en suivant la démarche de Hüsemann. On trouvera en annexe 9 le code SQL qui a été créé pour effectuer les requêtes.

Requête	Regroupement	Analyse	Création de vues
Requête 1	NO_CLIENT, MOIS, ANNEE	Moyenne annuelle des montants mensuels	Oui car moyenne n'est pas calculée sur le regroupement
Requête 2	NO_VENDEUR, NO_CLIENT, MOIS, ANNEE	Nombre de commandes, Montant total	Oui car la mesure du montant de la commande n'existe pas
Requête 3	NO_CLIENT, MOIS, ANNEE	Délai moyen	Non
Requête 4	NO_VENDEUR,	Nombre moyen de lignes de	Oui car la mesure du nombre

	MOIS	commande par commande	de lignes par commande n'existe pas
Requête 5	MARQUE, JOUR_DE_SEMAINE	Montant commandé / valeur totale	Non
Requête 6	NO_VENDEUR, MARQUE, MOIS, ANNEE	Rabais moyen / montant de la commande	Non
Requête 7	NO_CLIENT, MARQUE, ANNEE	Montant total	Non
Requête 8	NO_PRODUIT, MARQUE, MOIS, ANNEE	Prix de vente moyen	Non

Tableau 5.6 : Analyse du modèle multidimensionnel effectué au moyen de requêtes SQL.

5.2.5. Analyse par Discoverer

Les analyses par Discoverer nécessitent que l'administrateur de Discoverer crée des « complex folders », des champs calculés et effectue des jointures entre les différentes tables. Le Tableau 5.2 fournit la liste des éléments qu'il faut créer dans Discoverer Administrator.

Création de « complex folder »	Analyse commande HU Analyse commande mensuelle HU Analyse ligne commande HU	
Champ calculé	HU Li Commande	HU Montant Ligne
	Analyse commande HU	HU Nb Lignes HU Montant Commande
	Analyse commande mensuelle HU	HU Montant Mensuel
	Analyse ligne commande HU	HU Rabais
Jointures	HU Commande	HU Dim Commande HU Periode
	HU Dim Commande	HU Client HU Magasin HU Vendeur
	HU Li Commande	HU Dim Commande HU Periode HU Produit
	HU Magasin	HU Entrepot
	Analyse commande HU	HU Dim Commande HU Periode
	Analyse ligne commande HU	HU Dim Commande HU Periode HU Produit

Tableau 5.7 : « Complex folders », champs calculés et jointures à créer dans Discoverer Administrator pour le modèle multidimensionnel construit en suivant la démarche de Hüsemann.

5.3. Kimball

Kimball (1996) propose une méthodologie de construction d'un entrepôt de données en neuf points qui part de la description des analyses par les utilisateurs et de la disponibilité des données. Par la suite, nous ne considérerons que les cinq premiers points qui concernent le design conceptuel et logique du modèle multidimensionnel. En effet, les quatre derniers points concernent le design physique et ne seront pas abordés.

Les cinq points de la méthodologie pour construire un entrepôt de données, que nous allons aborder, sont les suivants :

- Analyse du processus d'affaires, identification des tables de fait,
- Détermination de la granularité de chaque table de fait,
- Détermination des dimensions et de leur granularité,
- Détermination des mesures des tables de fait,
- Attributs de chaque dimension,

La construction d'un entrepôt de données est réalisée à partir des informations recueillies auprès des utilisateurs (User Driven) et l'ordre des éléments du modèle multidimensionnel est le suivant :

Fait $\Rightarrow$ dimension $\Rightarrow$ mesure $\Rightarrow$ attributs des dimensions (FDM)

5.3.1. Analyse du processus d'affaire et identification des tables de fait

L'analyse du processus d'affaire est effectuée par un ensemble de personnes impliquées dans le développement, l'administration et l'utilisation de l'entrepôt de données. Les besoins des utilisateurs sont extraits d'une liste de requêtes. Les données disponibles sont analysées par des entrevues avec les DBA et en étudiant les schémas entité-relation. À partir des interviews des utilisateurs et des DBA, l'ensemble du groupe qui est impliqué dans le développement de l'entrepôt de données identifie les processus d'affaire qui seront utiles à l'analyse et à la prise de décision.

Dans le cas de Tarigomi, les processus d'affaire les plus importants sont:

- Le suivi des commandes,
- Le suivi des lignes de commandes,
- Le suivi du stock
- Le suivi des approvisionnements
- Le suivi des lignes des approvisionnements
- Le suivi des transferts entre les entrepôts

Chaque processus d'affaire donne lieu à une étoile dont le centre est un fait qui relie les dimensions nécessaires. Les tables de fait de Tarigomi sont donc:

- COMMANDE,
- DET_COMMANDE,
- INVENTAIRE,
- APPROVISIONNEMENT,
- DET_APPRO,
- TRANSFERT.

5.3.2. Détermination de la granularité de chaque table de fait

Il va falloir déterminer la granularité de chaque table et prendre une décision sur ce que signifie chaque élément de niveau le plus bas de la table de fait. Les mesures de la table de fait pourront être mémorisées pour chaque transaction, chaque ligne de produit, chaque jour ou un chaque mois. Dans le cas de Tarigomi, les analyses devront pouvoir être conduites au niveau de chaque transaction ce qui implique que la granularité des tables de fait sera le numéro de commande.

5.3.3. Détermination des dimensions et de leur granularité

Lorsque le grain de la table de fait est connu, les dimensions et leur granularité respective pourront être définies. On pourra construire des dimensions qui ne sont pas absolument nécessaires pour la table de fait mais qui seront utiles pour l'analyse. Par exemple, dans le cas de la table de fait qui est définie en fonction de l'état cumulatif des lignes de commande, on pourrait définir des dimensions sur la réception et la livraison des produits. Ces dimensions se rajouteront aux dimensions nécessaires à la table de fait. Le choix des dimensions est crucial. Il est effectué avant de déterminer les mesures qui seront analysées,

Dans le cas de Tarigomi, nous n'avons pas rajouté de dimensions à celle qui ont été établies à partir de l'analyse des processus d'affaire.

Les dimensions du fait KI_COMMANDE sont les suivantes :

- NO_COMMANDE,
- CLIENT,
- MAGASIN,
- VENDEUR,
- PERIODE.

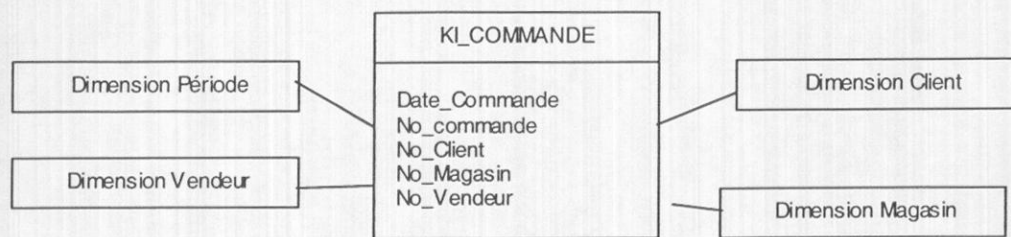


Figure 5.28: Modèle multidimensionnel de la gestion des commandes de Tarigomi selon Kimball.

Les dimensions du fait KI_DET_COMMANDE :

- NO_COMMANDE,
- CLIENT,
- MAGASIN,
- PRODUIT,
- VENDEUR,
- PERIODE.

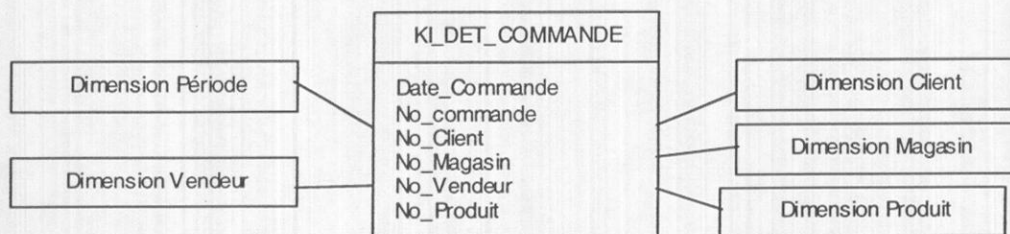


Figure 5.29 : Modèle multidimensionnel de la gestion des lignes de commande de Tarigomi.

Les dimensions du fait KI_INVENTAIRE :

- Produit,
- PERIODE,
- ENTREPOT.



Figure 5.30 : Modèle multidimensionnel de la gestion de l'inventaire de Tarigomi selon Kimball.

Les dimensions du fait KI_APPROVISIONNEMENT :

- FOURNISSEUR,
- PERIODE,
- ENTREPOT.

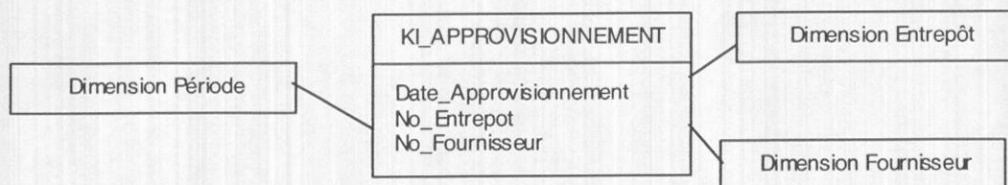


Figure 5.31 : Modèle multidimensionnel de la gestion des approvisionnements de Tarigomi selon Kimball.

Les dimensions du fait KI_DET_APPRO:

- FOURNISSEUR,
- PERIODE,
- ENTREPOT.
- PRODUIT

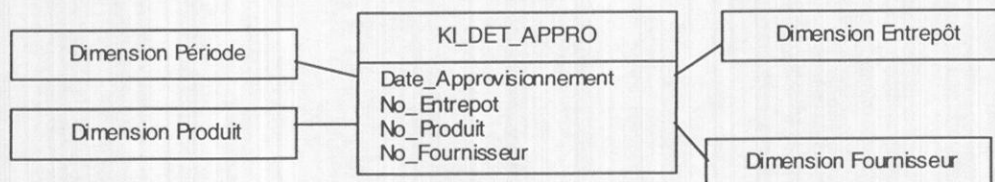


Figure 5.32 : Modèle multidimensionnel de la gestion des détails des approvisionnements de Tarigomi selon Kimball.

Les dimensions du fait KI_TRANSFERT:

- PERIODE,
- ENTREPOT.
- PRODUIT.

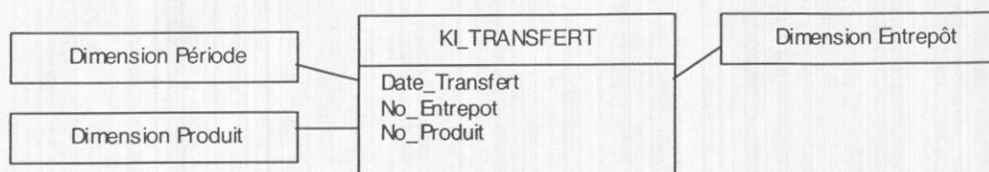


Figure 5.33 : Modèle multidimensionnel de la gestion des transferts de Tarigomi selon Kimball.

5.3.4. Détermination des mesures des tables de fait

Les mesures ont été déterminées au cours de l'analyse des besoins en information. Elles sont représentées sur la Figure 5.34.

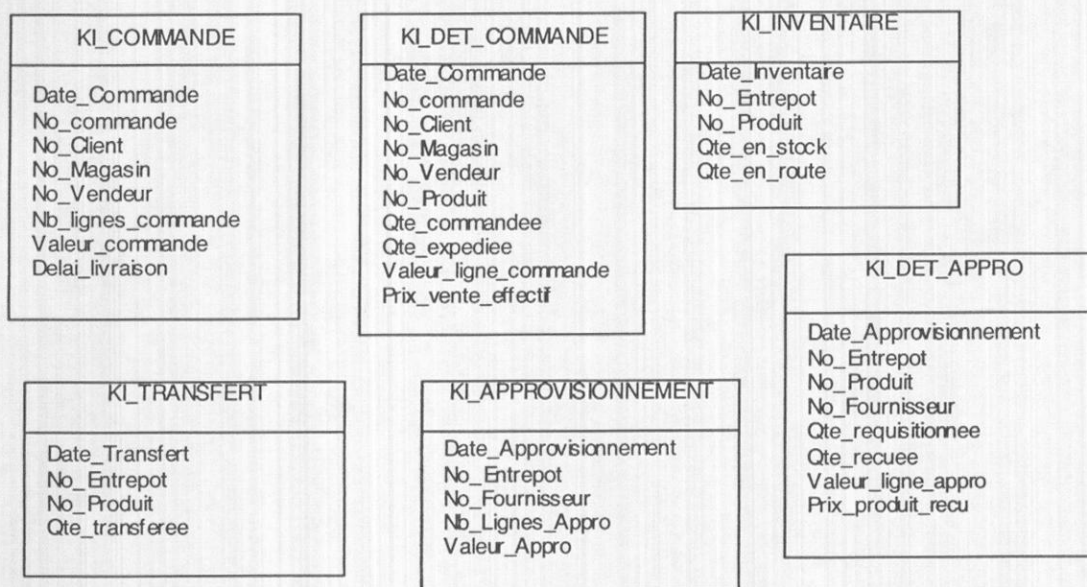


Figure 5.34 : Les tables de fait de Tarigomi selon Kimball.

5.3.5. Attributs de chaque dimension

Finalement, les attributs des dimensions seront définis. Il faut remarquer que Kimball ne prévoit pas de mécanisme pour définir les hiérarchies des dimensions. Les dimensions sont des tables qui contiennent tous les éléments et le modèle multidimensionnel ne fournit aucune information sur la structure interne des hiérarchies. La Figure 5.35 contient les attributs des dimensions.

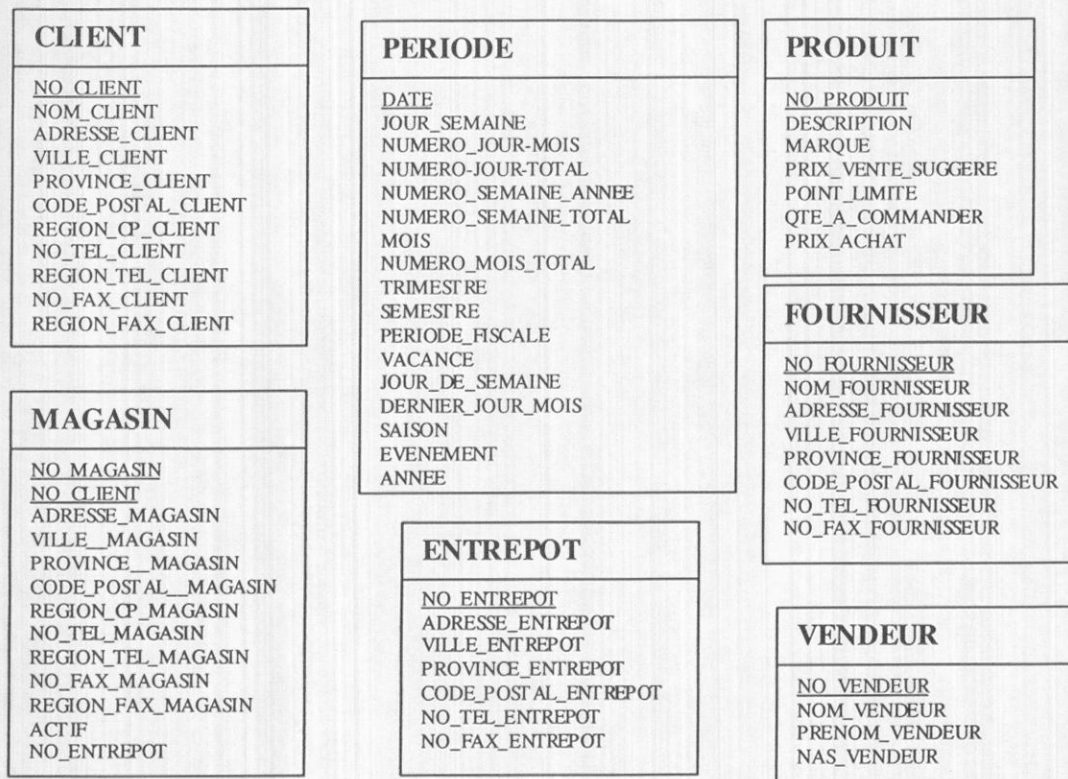


Figure 5.35 : Les tables des dimensions et leurs attributs pour les modèles multidimensionnels de Tarigomi selon Kimball.

5.3.6. Analyse par requêtes SQL

Toutes les requêtes de la liste du tableau 4.1 peuvent être effectuées sur les tables du modèle multidimensionnel construites en suivant la démarche de Kimball. On trouvera en annexe 9 le code SQL qui a été créé pour effectuer les requêtes.

Requête	Regroupement	Analyse	Création de vues
Requête 1	NO_CLIENT, MOIS, ANNEE	Moyenne annuelle des montants mensuels	Oui car moyenne n'est pas calculée sur le regroupement
Requête 2	NO_VENDEUR, NO_CLIENT, MOIS, ANNEE	Nombre de commandes, Montant total	Non
Requête 3	NO_CLIENT, MOIS, ANNEE	Délai moyen	Non
Requête 4	NO_VENDEUR, MOIS	Nombre moyen de lignes de commande par commande	Non
Requête 5	MARQUE,	Montant commandé / valeur	Non

	JOUR_DE_SEMAINE	totale	
Requête 6	NO_VENDEUR, MARQUE, MOIS, ANNEE	Rabais moyen / montant de la commande	Non
Requête 7	NO_CLIENT, MARQUE, ANNEE	Montant total	Non
Requête 8	NO_PRODUIT, MARQUE, MOIS, ANNEE	Prix de vente moyen	Non

Tableau 5.8 : Analyse du modèle multidimensionnel effectué au moyen de requêtes SQL.

5.3.7. Analyse par Discoverer

Les analyses par Discoverer nécessitent que l'administrateur de Discoverer crée des « complex folders », des champs calculés et effectue des jointures entre les différentes tables. Le Tableau 5.9 fournit la liste des éléments qu'il faut créer dans Discoverer Administrator.

Création de « complex folder »	Analyse commande mensuelle KI Analyse ligne commande KI	
Champ calculé	Analyse commande mensuelle KI	KI Montant Mensuel
	Analyse ligne commande KI	KI Rabais
Jointures	KI Commande	KI Periode KI Client KI Magasin KI Vendeur
	KI Li Commande	KI Periode KI Client KI Magasin KI Vendeur KI Produit
	KI Magasin	KI Entrepot
	Analyse ligne commande KI	KI Client KI Magasin KI Vendeur KI Periode KI Produit

Tableau 5.9 : « Complex folders », champs calculés et jointures à créer dans Discoverer Administrator pour le modèle multidimensionnel construit en suivant la démarche de Kimball.

5.4. Ballard

La démarche que Ballard propose (Ballard et al., 1998) consiste à déterminer les dimensions, les mesures et les faits à partir d'une liste de requêtes fournies par les utilisateurs. L'ordre de détermination des éléments est la suivant :

Dimension $\Rightarrow$ mesures $\Rightarrow$ fait (DMF)

Les auteurs font la remarque que les éléments déterminés au début de la démarche ne sont que des candidats qui pourront être transformés par la suite, certains d'entre eux pouvant disparaître du modèle, d'autres pouvant être fusionnés ou remplacés, d'autres encore pouvant être décomposés en plusieurs composants.

Les candidats des mesures sont déterminés en analysant les requêtes d'affaires, elles correspondent essentiellement à des données qui sont utilisées dans les requêtes pour mesurer la performance ou le comportement d'un processus ou d'un objet d'affaire.

5.4.1. Granularité des mesures

Rappelons que la granularité peut être définie comme le plus grand niveau de détail mémorisé dans l'entrepôt de données. Les mesures sont en général associées à plusieurs dimensions. La granularité des mesures est déterminée par la combinaison des détails d'enregistrement de toutes les dimensions. Plusieurs mesures différentes, qui ont la même granularité, peuvent être regroupées dans un même fait.

5.4.2. Détermination des dimensions

Ballard définissent les dimensions en transformant le modèle entité-relation des données qui seront analysées en enlevant toutes les entités associatives et les sous entités. En faisant cette opération, il faut bien prendre soin de créer toutes les relations de N à M qui remplacent ces entités (voir Figure 5.36).

L'étape suivante consiste à regrouper toutes les entités qui sont à l'extrémité des relations de un à plusieurs (N à M) pour obtenir un modèle simplifié où n'apparaissent plus que les entités terminales des dimensions. Au cours de cette opération, les hiérarchies des dimensions qui sont explicitées dans le modèle entité-relation disparaissent et les dimensions ne sont plus représentées que par une seule entité (voir Figure 5.37). Dans le cas de Tarigomi, l'entité CLIENT a été rabattue dans l'entité MAGASIN et tous les attributs du client ont été rajoutés au magasin. L'entité ENTREPOT a été conservée dans la mesure où il existe une relation de N à M entre PRODUIT et ENTREPOT.

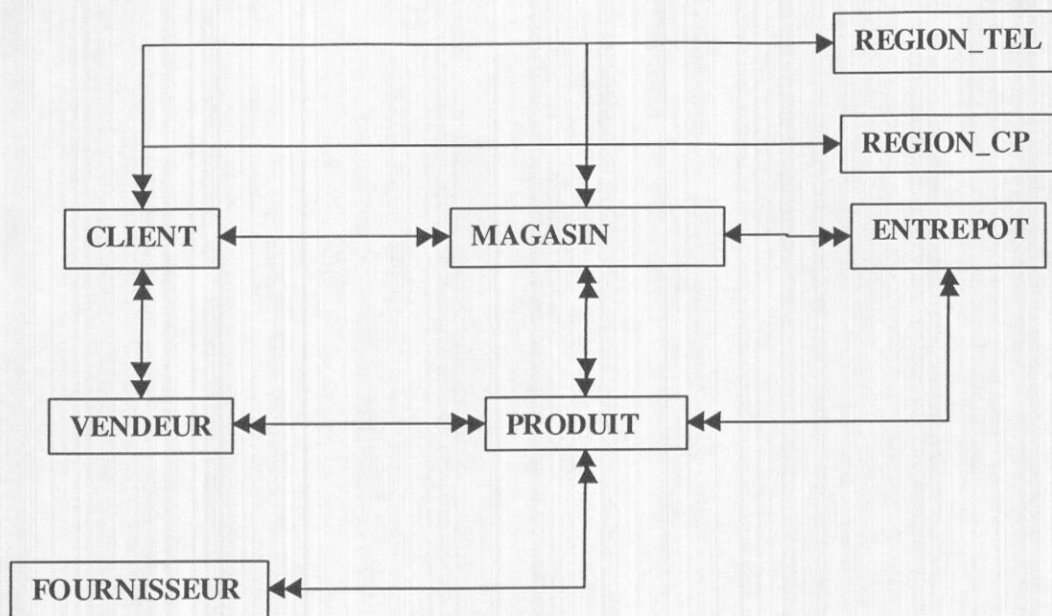


Figure 5.36 : Modèle entité-relation de Tarigomi dans lequel on a enlevé toutes les entités associatives et on les a remplacées par des relations de N à M.

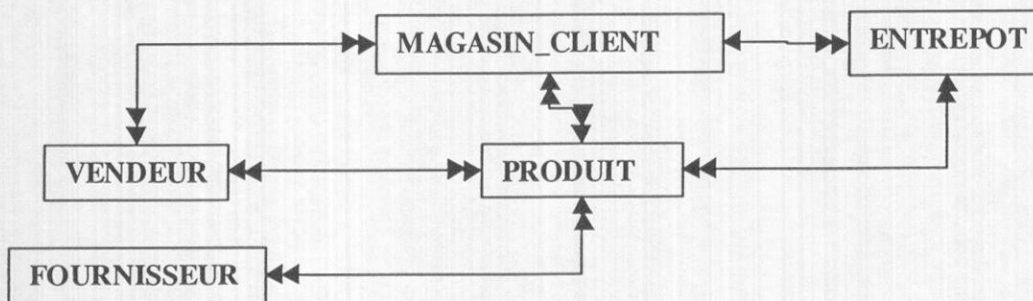


Figure 5.37 : Regroupement des entités qui sont aux extrémités des relations N à M.

5.4.3. Détermination des mesures

Les mesures sont déterminées en analysant les requêtes et en extrayant les objets sur lesquels portent les analyses. Dans le cas de Tarigomi, nous avons choisi dix requêtes qui ont été présentées dans le tableau 4.1 et qui ont été détaillées dans l'annexe 9. Nous reprenons ici la liste des requêtes pour simplifier la lecture.

No	Requêtes
Q1	Valeur moyenne des montants mensuels totaux par client
Q2	Nombre de commandes et ventes totales par vendeur, par client, par mois, par année
Q3	Délai moyen de livraison, par client, par mois, par année
Q4	Nombre moyen de lignes de commande par commande pour chaque vendeur, pour chaque mois de toutes les années confondues
Q5	Valeur commandée par marque par jour de la semaine en pour cent de la valeur totale des commandes
Q6	Rabais en pour-cent par vendeur, par marque, par mois et par année
Q7	Ventes totales par année, par client, par magasin, par marque
Q8	Quantité moyenne quotidienne en stock par produit, par année, par mois
Q9	Nombre de transferts entre entrepôt par entrepôt, par année, par mois
Q10	Évolution du prix moyen de vente effectif par produit par année, par mois

Tableau 4.1 : Liste des requêtes.

5.4.4. Détermination des relations entre les requêtes et les dimensions ou les mesures

Les relations entre les requêtes et les dimensions ou les mesures sont déterminées à partir d'un tableau qui décrit pour chaque dimension et pour chaque mesure si elle intervient dans la requête. Le tableau suivant a été construit à partir de la liste des requêtes du Tableau 5.10.

Dimension	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
Magasin_Client	X	X	X				X			
Vendeur		X		X		X				
Produit					X	X	X	X		X
Entrepôt								X	X	
Fournisseur										

Mesures	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
Montant mensuel moyen	X									
Montant total		X					X			
Nombre de commandes		X								
Délai moyen			X							
Nombre de lignes de commande				X						
Valeur commandée en pour cent					X					
Rabais en pour cent						X				
Quantité moyenne en stock								X		
Nombre de transfert									X	
Prix moyen de vente effectif										X

Tableau 5.10 : Relations entre les requêtes et les mesures ou les dimensions.

5.4.5. Ajout d'une dimension temporelle et construction des attributs des dimensions

Pour analyser les données, il faut toujours incorporer une dimension temporelle. Ballard et al.(1998) recommandent d'en rajouter une aux autres dimensions qui ont été déterminées dans le Tableau 5.10 d'analyse.

Les analyses effectuées sur l'entrepôt de données font intervenir systématiquement la dimension temporelle. Comme cette dimension n'apparaît pas dans le diagramme entité-relation qui a été obtenu au cours de la première étape, il est nécessaire de la rajouter dans tous les modèles multidimensionnels.

Les attributs des dimensions sont construits à partir du schéma de structure de données de Tarigomi. La Figure 5.38 présente les dimensions et leurs attributs qui ont été construits pour le cas de Tarigomi.

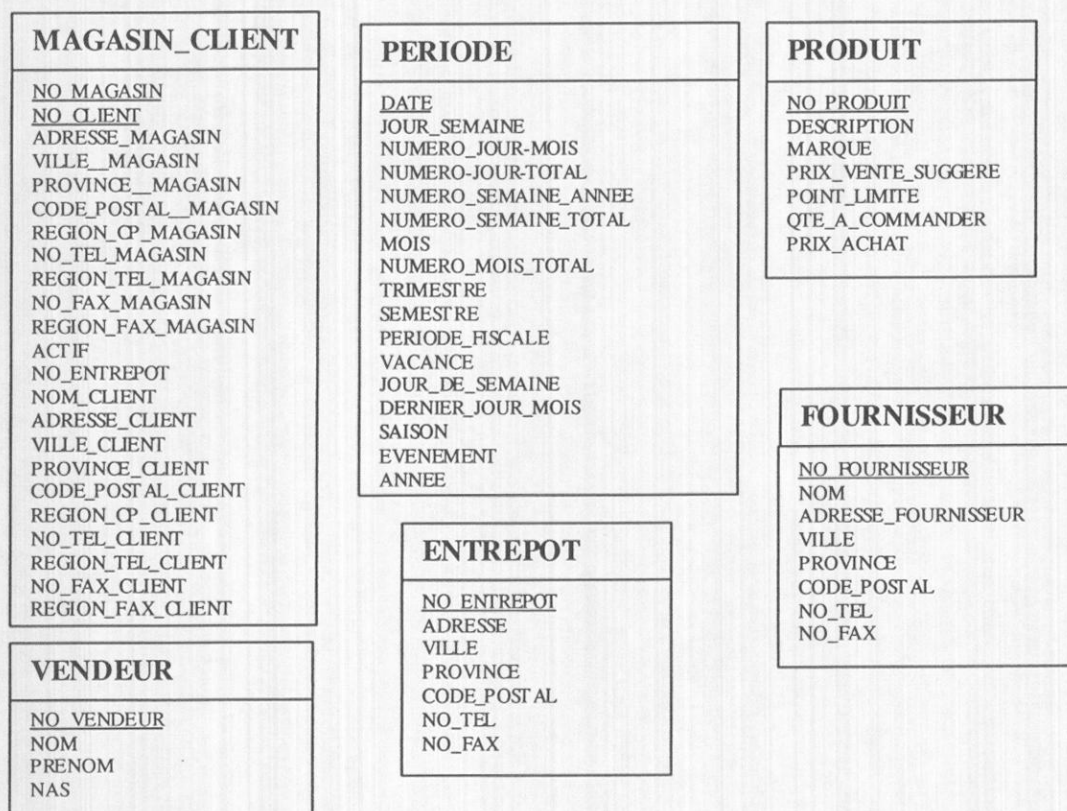


Figure 5.38 : Dimensions de Tarigomi.

5.4.6. Création des faits

Les faits contiennent les mesures du Tableau 5.10 qui sont associées aux dimensions. Dans un premier temps, on crée un fait pour chaque mesure sauf lorsque plusieurs mesures font référence à des dimensions identiques. La Figure 5.39 présente la construction initiale des faits pour le cas de Tarigomi. On remarque que les attributs des tables de fait sont très exactement les mesures qui ont été extraites des requêtes du Tableau 5.10.

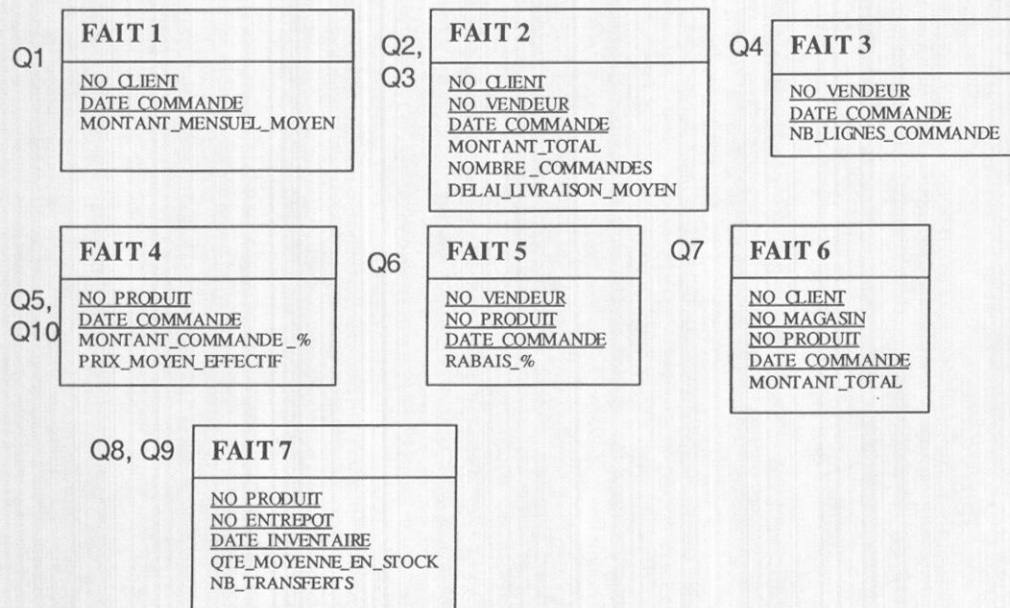


Figure 5.39 : Version initiale des faits pour les requêtes Q1 à Q10. Les attributs des tables de fait sont les mesures du Tableau 5.10.

5.4.7. Modification des faits en fonction de la granularité et de l'additivité des mesures

L'étape suivante de la démarche consiste à modifier la définition des faits pour simplifier le modèle et l'étendre afin qu'il réponde au plus grand nombre de requêtes.

Granularité

La granularité des données doit être la plus élevée possible, c'est-à-dire que les données doivent être mémorisées avec le plus haut niveau de détail possible. Pour réduire la granularité lorsque les analyses le nécessitent, il sera possible de regrouper les données.

Additivité

L'additivité est la capacité d'une mesure à être récapitulée. Rappelons que les mesures peuvent être additives, semi-additives ou non additives. Les pourcentages ne sont pas additifs. L'additivité est importante lorsqu'on fait des regroupements sur des tables de fait. En général, on cherche à avoir des mesures qui sont additives.

Fusion des faits

Pour étendre le domaine des analyses possibles, il est utile de fusionner les faits. La granularité de certains faits devra être modifiée lors de la fusion. Il faudra parfois aussi rajouter des dimensions. En fusionnant les faits, l'utilisateur a plus de facilité à retrouver les données nécessaires à son analyse étant donné qu'il a moins d'endroits à étudier. D'autre part, les possibilités d'analyse sont plus grandes en fusionnant les faits puisqu'un plus grand nombre de mesures est associé à un plus grand nombre de dimensions. Finalement, la fusion des faits entraîne moins d'administration et de maintenance.

Cette étape va consister à transformer les faits qui ont été obtenus au cours de l'étape précédente pour harmoniser la granularité des mesures et pour ne conserver que des mesures additives. La

Figure 5.40 présente les faits de la version initiale avec des attributs qui ont été transformés.

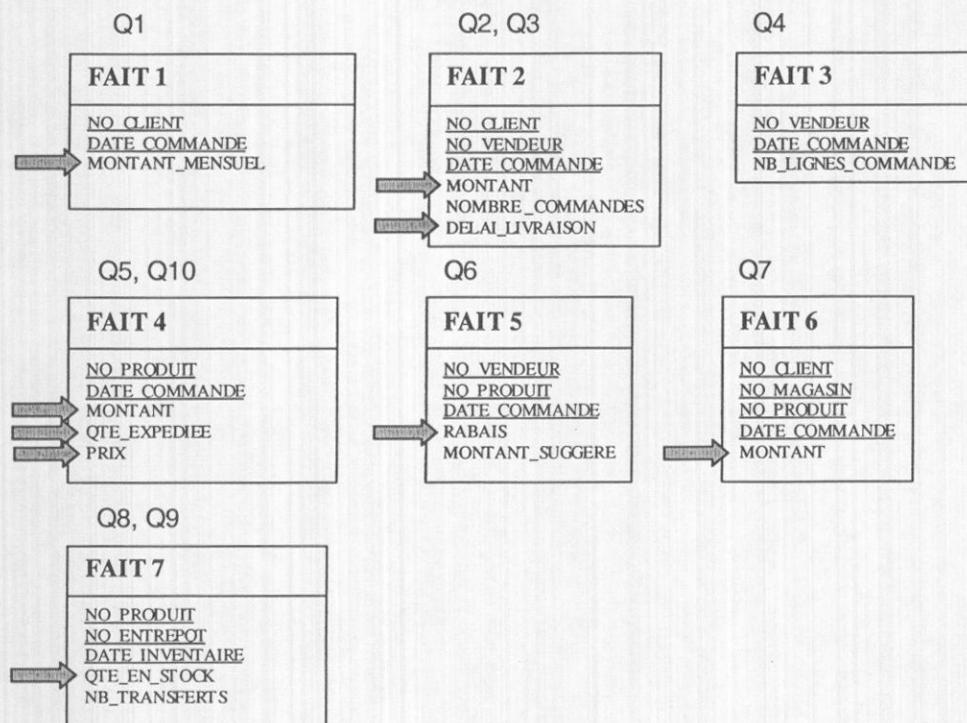


Figure 5.40 : Les attributs des faits indiqués dans la Figure 5.39 sont modifiés pour harmoniser la granularité et ne garder que des mesures additives.

5.4.8. Fusion des faits

Les faits obtenus au cours de l'étape précédente sont regroupés pour améliorer la compréhension du modèle multidimensionnel, pour étendre les possibilités d'analyse et pour éviter une administration trop importante de la base de données. Dans le cas de Tarigomi, on obtient quatre entités après avoir fusionné les faits (

Figure 5.41).

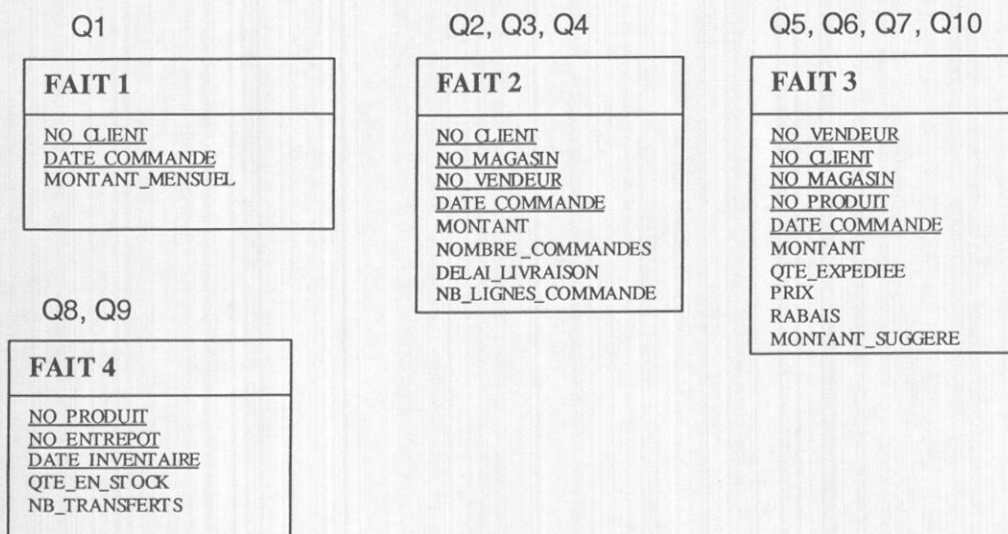


Figure 5.41 : Les faits de la Figure 5.40 ont été fusionnés

Finalement, les faits ont été nommés pour faciliter la lecture. De plus, les faits 2 et 3 ont été améliorés en rajoutant la dimension du numéro de commande. Dans le cas de Tarigomi, on n'introduit que la clé NO_COMMANDE dans FAIT COMMANDE et FAIT DET_COMMANDE dans la mesure où il n'y a pas d'informations associées à ces faits qui n'existe pas déjà dans les tables existantes.

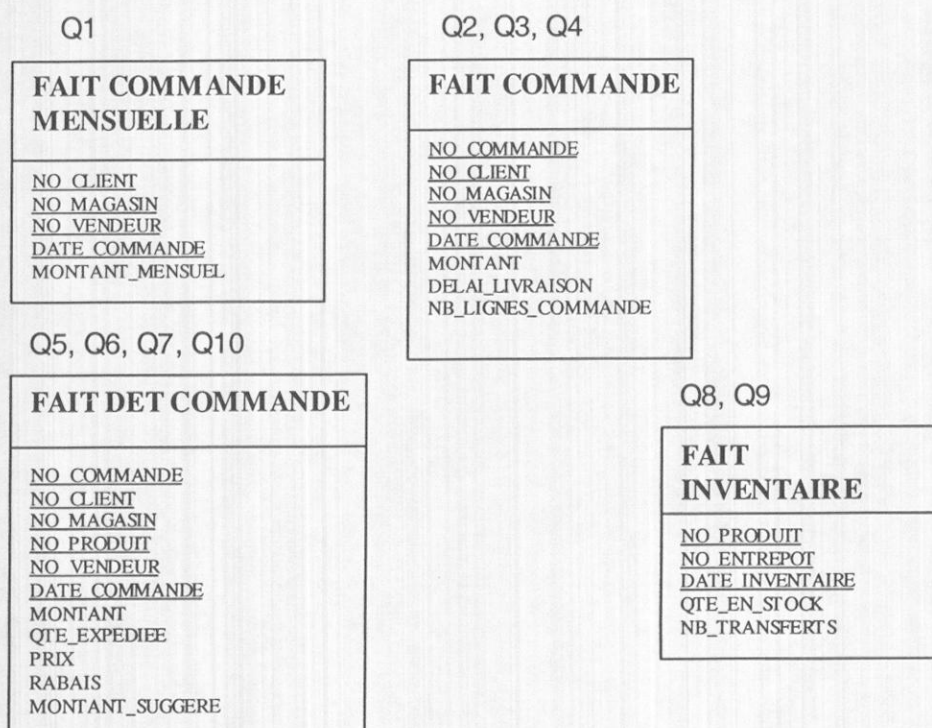


Figure 5.42 : Version finale des faits pour le cas de Tarigomi.

5.4.9. Production du modèle multidimensionnel

La dernière étape consiste à mettre en relation les faits qui ont été obtenus avec les dimensions qui les caractérisent. Le fait COMMANDE MENSUELLE est associé aux dimensions CLIENT, VENDEUR et PERIODE. Les Figure 5.43, Figure 5.44, Figure 5.45 et Figure 5.46 montrent les modèles multidimensionnels obtenus en rattachant les faits aux dimensions.

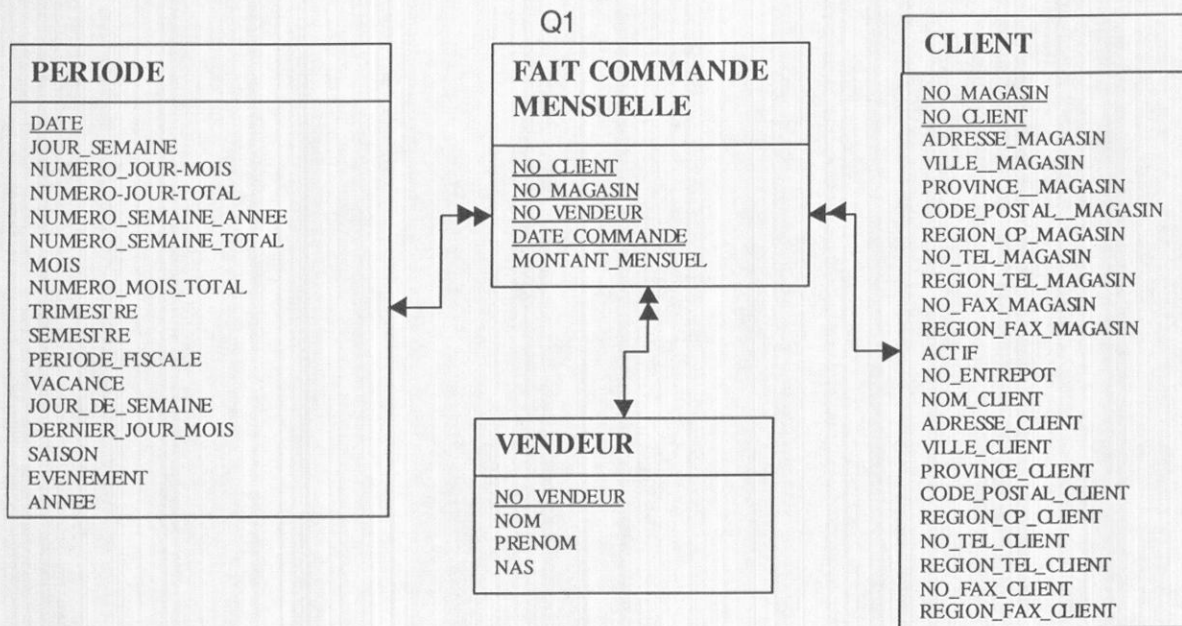


Figure 5.43 : Modèle multidimensionnel des commandes mensuelles.

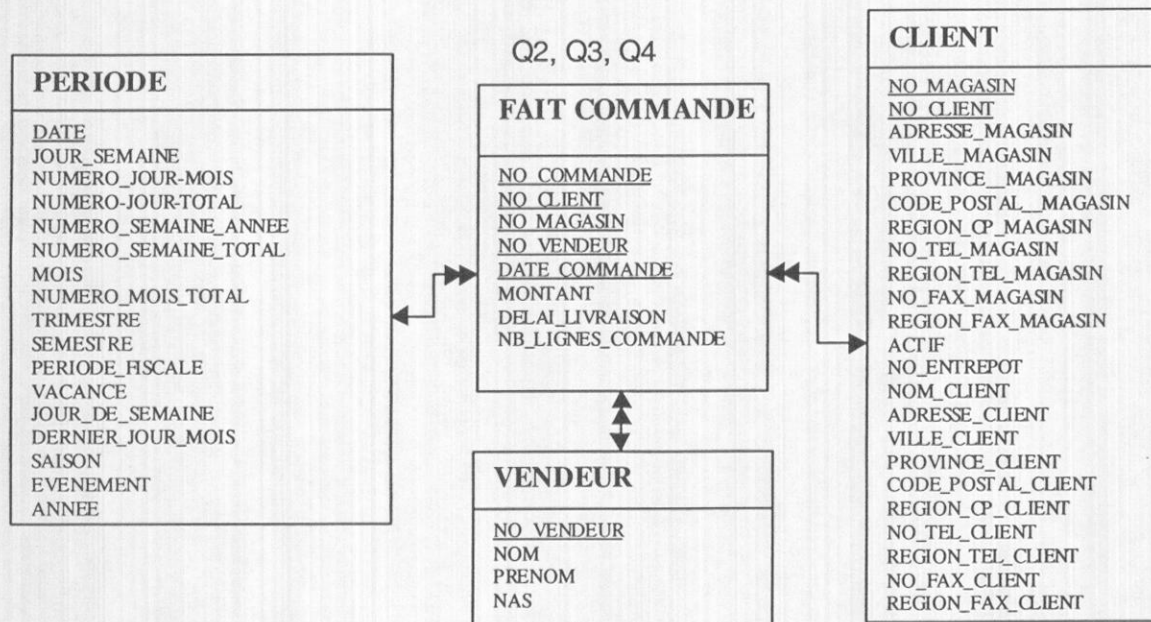


Figure 5.44 : Modèle multidimensionnel des commandes.

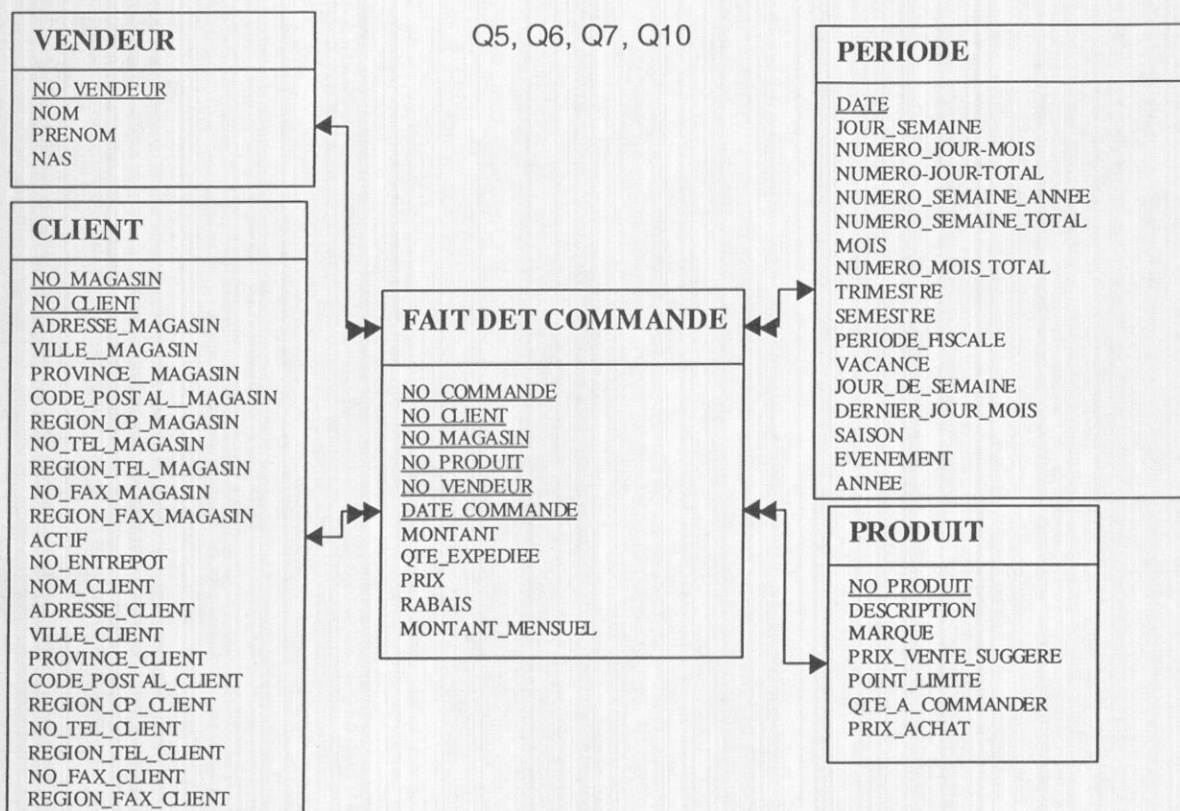


Figure 5.45 : Modèle multidimensionnel des lignes de commande.

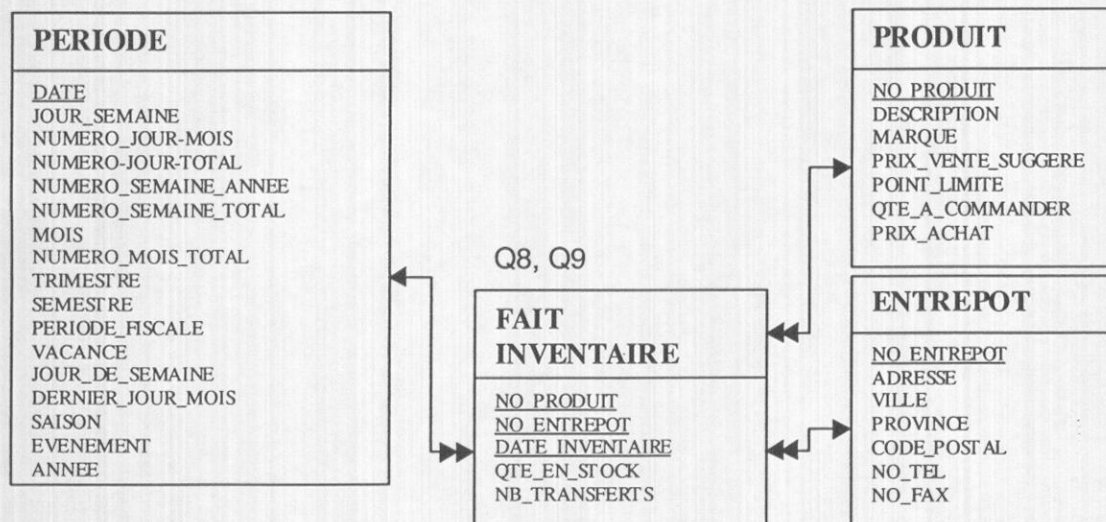


Figure 5.46 : Modèle multidimensionnel de l'inventaire.

5.4.10. Analyse par requêtes SQL

Toutes les requêtes de la liste du tableau 4.1 peuvent être effectuées sur les tables du modèle multidimensionnel construites en suivant la démarche de Ballard. On trouvera en annexe 9 le code SQL qui a été créé pour effectuer les requêtes.

Requête	Regroupement	Analyse	Création de vues
Requête 1	NO_CLIENT, MOIS, ANNEE	Moyenne annuelle des montants mensuels	Oui car moyenne n'est pas calculée sur le regroupement
Requête 2	NO_VENDEUR, NO_CLIENT, MOIS, ANNEE	Nombre de commandes, Montant total	Non
Requête 3	NO_CLIENT, MOIS, ANNEE	Délai moyen	Non
Requête 4	NO_VENDEUR, MOIS	Nombre moyen de lignes de commande par commande	Non
Requête 5	MARQUE, JOUR_DE_SEMAINE	Montant commandé / valeur totale	Non
Requête 6	NO_VENDEUR, MARQUE, MOIS, ANNEE	Rabais moyen / montant de la commande	Non
Requête 7	NO_CLIENT, MARQUE, ANNEE	Montant total	Non
Requête 8	NO_PRODUIIT, MARQUE, MOIS, ANNEE	Prix de vente moyen	Non

Tableau 5.11 : Analyse du modèle multidimensionnel effectué au moyen de requêtes SQL.

5.4.11. Analyse par Discoverer

Les analyses par Discoverer nécessitent que l'administrateur de Discoverer crée des « complex folders », des champs calculés et effectue des jointures entre les différentes tables (cf Annexe pour la présentation de Discoverer). Le Tableau 5.12 fournit la liste des éléments qu'il faut créer dans Discoverer Administrator.

Jointures	BA Commande	BA Periode BA Client BA Magasin BA Vendeur
	BA Li Commande	BA Periode BA Client BA Magasin BA Vendeur BA Produit
	BA Magasin	BA Entrepot

Tableau 5.12 : « Complex folders », champs calculés et jointures à créer dans Discoverer Administrator pour le modèle multidimensionnel construit en suivant la démarche de Ballard.

6. Comparaison et critique des démarches

Au cours de ce chapitre, nous allons comparer les démarches qui ont été décrites au chapitre 5. Nous allons tout d'abord les comparer par rapport à l'origine des éléments du modèle multidimensionnel. Nous allons ainsi dresser une liste des objets qui sont nécessaires pour débiter la construction d'un entrepôt de données. Par la suite, nous allons comparer la mise en place de la démarche, en particulier l'implication des utilisateurs, l'importance de la qualité du modèle entité-relation et la manière de déterminer les mesures, les faits, les dimensions, les hiérarchies des dimensions et les contraintes d'agrégation. De plus, nous comparerons les résultats de l'application des démarches dans le cas de Tarigomi et nous analyserons les résultats de l'application de Discoverer.

6.1. _____ Pré-requis à la démarche

Les éléments du modèle multi dimensionnel qui vont être définis proviennent soit des modèles de données de l'entreprise, soit de l'étude des processus d'affaire qui vont être analysés soit encore des deux.

Moody et Kortink

La démarche de Moody et Kortink est basée uniquement sur le modèle des données de l'entreprise. Les auteurs, en ne faisant jamais référence aux utilisateurs, estiment que leur démarche est mieux structurée que celle qui repose sur l'analyse des besoins des usagers. De plus, la structure de l'entrepôt de données qui aura été construit reflète mieux à leurs yeux la structure sémantique des données. Finalement, leur démarche conduit à un design de l'entrepôt de données qui est plus flexible et moins sensible aux changements des analyses au cours du temps.

Hüsemann

La démarche de Hüsemann repose, elle aussi, sur le modèle entité-relation des données de l'entreprise. Ce modèle est indispensable et Hüsemann et al. (2000) suggèrent de faire du « reverse engineering » s'il n'est pas disponible. Mais la démarche pour construire le modèle multidimensionnel s'appuie aussi sur un tableau des rôles des attributs qui a été défini par un groupe d'utilisateurs. Le contenu de ce tableau est déterminant parce qu'il va constituer un des points de départ de la démarche et que sa qualité est très importante.

Une fois le tableau établi, la construction du modèle multidimensionnel est automatique et suit une procédure algorithmique, ce qui constitue un avantage aux yeux de Hüsemann et al. (2000).

Kimball

Dans le cas de Kimball, la démarche est basée, au contraire de Moody et Kortink, sur l'analyse des besoins des utilisateurs par une équipe de développement de l'entrepôt de données qui détermine les processus d'affaire qui seront analysés.

Kimball insiste pour indiquer que cette méthodologie ne doit pas être suivie à la lettre et ne fournit pas une formule qui peut être appliquée sans discernement. Il faut faire appel à beaucoup de jugement et déterminer au cas par cas ce qui est nécessaire à l'analyse et à la prise de décision. Il faut aussi prendre soin de garder le modèle multi dimensionnel le plus simple possible.

Ballard

La démarche de Ballard repose, comme dans le cas de celle proposée par Hüsemann et al. (2000), sur le modèle entité-relation des données de l'entreprise et sur une liste de besoins d'information définie par un groupe d'utilisateur. Ces besoins sont décrits sous la forme d'une liste de requêtes qui permettront de déterminer les mesures. Les mesures correspondent donc exactement aux sujets des requêtes, ce qui pourra causer des problèmes si la liste des requêtes n'est pas complète ou qu'elle ne correspond qu'aux besoins d'un groupe particulier d'utilisateurs. On risque alors de ne pas construire un entrepôt de données qui conviennent à tous les utilisateurs de l'entreprise. La démarche convient donc bien à un mini-entrepôt qui est centré sur les besoins d'un département ou d'un groupe d'analystes.

6.2. _____ Mise en place de la démarche

6.2.1. _____ Implication des utilisateurs

Le choix de Moody et Kortink (2000) de ne pas tenir compte des besoins des utilisateurs semble une hérésie dans le développement de systèmes d'information mais il est partagé par McGuff (1998) qui estime que les utilisateurs vont proposer une liste de requêtes en fonction de ce qu'ils ont l'habitude d'analyser et que l'entrepôt de données ne permettra pas de répondre à des questions nouvelles. Cet auteur relève qu'un entrepôt de données est une structure organique qui croît de manière imprévisible en fonction des besoins et qu'il n'est pas possible de définir à priori et une fois pour toutes une structure optimale qui résiste aux changements.

Il faut tout de même relever que le modèle entité-relation doit être de grande qualité et qu'il doit contenir toutes les informations qui seront utiles à l'analyse. Ce dernier point risque de ne pas être satisfait étant donné qu'il est en contradiction avec le fait que les structures des bases de données opérationnelles sont différentes de celles des entrepôts de données et ne contiennent donc pas explicitement toutes les informations nécessaires à l'analyse.

La démarche proposée par Hüsemann nécessite l'implication d'une équipe d'experts pour définir le rôle des attributs. La qualité de l'entrepôt de données repose donc sur celle de la définition des rôles et sur celle du modèle entité-relation. De la même manière, la démarche de Ballard implique une participation des utilisateurs au début du développement de l'entrepôt de données pour définir la liste des requêtes. La poursuite de la construction de l'entrepôt de données se fait par contre de manière automatique et ne nécessite plus l'intervention des utilisateurs.

Dans le cas de Kimball, les utilisateurs sont impliqués dans le choix des processus à analyser et dans celui des dimensions et des mesures. Comme la démarche n'est pas automatique mais qu'une place importante est laissée au jugement de l'équipe de développement, il est très important que les utilisateurs participent aux décisions et puissent intervenir tout au long du processus de construction de l'entrepôt de données.

6.2.2. _____ Importance de la qualité du modèle entité-relation

La démarche de Moody et Kortink repose entièrement sur le modèle entité-relation des bases de données opérationnelles et la qualité de ce modèle est primordiale. Elle va déterminer dans quelle mesure l'entrepôt de données sera efficace et permettra des analyses utiles. Dans le cas de la démarche de Hüsemann, le modèle entité-relation devra aussi être de bonne qualité et comprendre les attributs qui seront nécessaires pour effectuer les analyses.

Kimball n'utilise pas le modèle entité-relation pour construire le modèle multi dimensionnel et la démarche qu'il propose n'est pas du tout affectée par la qualité ou l'existence d'un modèle entité-relation. La démarche de Ballard ne s'appuie sur le modèle entité-relation que pour déterminer les dimensions du modèle multi dimensionnel. Ce dernier pourra donc être de moindre qualité sans que le modèle multi dimensionnel ne soit affecté.

6.2.3. _____ Détermination des mesures

Pour Moody et Kortink, les mesures sont les attributs numériques des entités de transaction. Hüsemann s'appuie par contre sur la définition par un groupe d'experts du rôle des attributs. La détermination des mesures ne fait donc pas partie de la démarche de modélisation en tant que telle mais a été réalisée au préalable. Kimball n'est pas très clair sur la manière de déterminer les mesures. Elles doivent être établies au cours de l'analyse des processus d'affaire mais ne sont intégrées au modèle qu'à la fin. La démarche de Ballard a recours à la liste des requêtes des utilisateurs pour choisir les mesures.

6.2.4. _____ Détermination des dimensions

La démarche de Moody et Kortink détermine les dimensions à partir de la classification des entités du modèle entité-relation des données de l'entreprise. Les auteurs ne prévoient pas de mécanisme pour traiter le cas des dimensions qui ne contiennent qu'un identificateur sur lequel on pourra faire des opérations de comptage, comme par exemple la commande dans le cas de Tarigomi (dimension dégénérée). On a vu dans le paragraphe 5.1.5 que cette absence de dimension dégénérée a un impact direct sur la granularité de la table du fait MK_COMMANDE et qu'elle va entraîner l'incapacité du modèle à fournir une réponse aux requêtes 2, 3 et 4.

Dans le cas de la démarche de Hüsemann, les attributs terminaux des dimensions sont tout d'abord déterminés par les dépendances fonctionnelles des mesures. Les dimensions sont ensuite développées à partir des attributs terminaux en suivant les dépendances fonctionnelles entre les attributs des dimensions.

Pour Kimball, les dimensions sont choisies tout d'abord pour caractériser les faits qui ont été déterminés à partir des processus d'affaire. Ensuite, on peut rajouter des dimensions qui caractérisent les faits pour étendre les possibilités d'analyse. Dans le cas de Tarigomi, on n'a pas rajouté de dimension mais le cas échéant, on pourrait inclure dans le modèle des dimensions qui caractérisent le mode de livraison de la commande ou le type de négociation du prix avec les clients.

La démarche de Ballard prévoit de déterminer les dimensions à partir du modèle entité-relation dont on a retiré toutes les entités associatives.

6.2.5. _____ Détermination des faits

Dans le cas de la démarche de Moody et Kortink, les faits sont déterminés à partir de la classification des entités du modèle entité-relation et correspondent aux entités de transaction. Pour Hüsemann, les faits sont construits par regroupement des mesures qui sont associées au même ensemble de dimensions. La

démarche de Kimball établit les faits à partir de l'analyse du processus d'affaire. Dans le cas de Ballard, un fait est d'abord construit pour chaque mesure ou groupe de mesures associées à des dimensions identiques. Par la suite, les faits sont modifiés en fonction de la granularité et de l'additivité des mesures. Finalement, les tables de faits sont regroupées pour étendre les possibilités d'analyse, faciliter la compréhension du modèle par les utilisateurs et limiter l'administration de l'entrepôt de données.

6.2.6. _____ Description explicite des hiérarchies

Les hiérarchies des dimensions sont très importantes lors de la modélisation multi dimensionnelles dans la mesure où elles guident la navigation le long des différents niveaux et permettent de détailler ou de regrouper les analyses. Il est donc très utile de décrire explicitement les hiérarchies d'une part pour faciliter la communication entre les utilisateurs et les développeurs et d'autre part pour aider la navigation au cours de l'analyse.

Dans le cas de la démarche proposée par Moody et Kortink, les hiérarchies maximales sont déterminées afin de définir comment on va rabattre les entités les unes dans les autres. Les hiérarchies sont définies à partir des relations 1 à N entre les entités du modèle entité-relation de départ. Ainsi, les hiérarchies ne sont pas explicitement définies mais servent plutôt à faire les bons rabattements entre les entités qui existent dans le modèle entité-relation.

La démarche de Husemann prévoit une définition explicite des hiérarchies construites à partir des dépendances fonctionnelles entre les attributs de dimension. Les auteurs utilisent même une représentation graphique des hiérarchies ce qui facilite la compréhension de la navigation entre les différents niveaux.

Dans le cas de Kimball, le modèle multi dimensionnel ne contient pas de représentation explicite des hiérarchies. Kimball s'attache particulièrement aux performances de l'analyse des données et montre que la taille des tables de dimension est très faible par rapport à celle des faits. Il n'y a donc pas d'intérêt à garder des tables normalisées pour les dimensions et qu'on peut obtenir une perte d'efficacité de l'entrepôt de données si le schéma multidimensionnel est en constellation.

Pour Ballard, le modèle ne comprend pas de représentation explicite des hiérarchies et les dimensions sont regroupées le plus possible.

6.2.7. _____ Définition des contraintes d'agrégation

Pour Pedersen (2000), il est important que le modèle multidimensionnel reflète la sémantique d'agrégation des mesures et qu'il fournisse un « filet de sécurité » pour que les requêtes retournent toujours des résultats qui ont un sens. Il s'agit d'éviter de compter plusieurs fois des données identiques ou d'additionner des mesures qui ne sont pas additives.

Moody et Kortink (2000) ne parlent pas de cette difficulté et ne mentionnent pas de mécanisme pour assurer que les résultats des requêtes ont un sens. Hüsemann et al. (2000) pour leur part, prennent beaucoup de soin à définir les contraintes d'agrégation et insistent pour que le modèle assure que les calculs d'agrégation des mesures ont un sens pour tous les niveaux des dimensions. Ils prévoient une étape dans leur démarche pour définir de manière explicite les contraintes d'agrégation des mesures en fonction de tous les niveaux des dimensions.

Kimball (1996) mentionne l'importance des contraintes d'agrégation et précise (voir chapitre 2) qu'il existe des mesures additives, semi-additives ou non additive. Dans le cas de la démarche de Ballard et al. (1998), la notion d'additivité est utilisée pour modifier les mesures de manière à ne garder que celles qui sont additives.

6.2.8. _____ Prise en compte de la granularité

Dans le cas de Moody et Kortink, la granularité des faits n'est pas explicitement mentionnée mais elle est implicitement fixée par les dimensions. Comme il n'y a pas de dimension dégénérée de commande, la granularité du fait MK_COMMANDE correspond à celle des dimensions MK_PERIODE, MK_CLIENT, MK_MAGASIN et MK_VENDEUR. De la même manière, la granularité du fait MK_LI_COMMANDE correspond à celle des mêmes dimensions plus la dimension MK_PRODUIT. Ainsi, le modèle développé en suivant la démarche de Moody et Kortink ne permet pas de fournir une réponse à certaines requêtes. Pour leur part, Hüsemann et al. (2000) ne font pas mention de la granularité et supposent que le niveau de détail du modèle entité-relation correspond à la granularité nécessaire du modèle multidimensionnel. Dans leur cas, la structure du modèle est telle que toutes les requêtes peuvent être exécutées.

Pour Kimball en revanche, la définition de la granularité est très importante et doit être faite au cours de la démarche. Elle doit être déterminée relativement tôt, c'est-à-dire avant de choisir les dimensions. La granularité est utilisée par Ballard au cours de la démarche pour raffiner les faits. En effet, les possibilités d'analyse sont augmentées lorsqu'on augmente la granularité.

Le Tableau 6.1 fait la synthèse de la comparaison des différentes démarches.

	Moody et Kortink (2000)	Hüsemann et al. (2000)	Kimball (1996)	Ballard et al. (1998)
Pré-requis	Modèle entité-relation	Modèle entité-relation et tableau des rôles des attributs	Étude des processus d'affaire à analyser	Modèle entité-relation et liste des requêtes
Implication des utilisateurs	Non	Oui (tableau du rôle des attributs)	Oui	Oui (liste des requêtes)
Importance de la qualité du modèle entité-relation	Grande	Grande	Très faible	Faible
Détermination des faits	Entités de transaction	Regroupement des mesures associées à des dimensions identiques	Analyse des processus d'affaire	Regroupement des mesures associées à des dimensions identiques
Détermination des dimensions	Entités composantes	Attributs dont dépendent fonctionnellement les mesures	Caractéristiques des faits	Entités du modèle entité-relation qui ne sont pas associatives
Détermination des mesures	Attributs numériques des entités de transaction	Attributs dont le rôle est déterminé par un groupe d'experts	Par analyse des processus d'affaire	Par analyse des requêtes
Description explicite des hiérarchies	Non	Oui	Non	Non
Définition des contraintes d'agrégation	Non	Oui	Partiellement	Partiellement
Prise en compte de la granularité	Non	Non	Oui	Oui

Tableau 6.1 : Tableau récapitulatif de la comparaison des démarches.

6.3. Résultats des démarches

Remarquons tout d'abord que les modèles multi dimensionnels qui ont été construits en suivant les quatre démarches sont relativement proches les uns des autres. Le modèle de Tarigomi a été conçu pour l'analyse des commandes et des lignes de commande ce qui assure que le modèle entité-relation est proche de celui de l'entrepôt de données. D'ailleurs, il est possible de répondre à toutes les questions par des requêtes directes sur le modèle de Tarigomi. Ce résultat pourra être très différent si les modèles des bases de données opérationnelles ne sont pas prévus à l'origine pour permettre des analyses mais ont été optimisés pour la rapidité des transactions ou la qualité des mises à jour.

Le Tableau 6.2 fournit la liste des faits, des dimensions et des mesures qui ont été générés en construisant les tables prescrites par les modèles en suivant les différentes démarches.

	Moody et Kortink (2000)	Hüsemann et al. (2000)	Kimball (1996)	Ballard et al. (1998)
Faits	MK_COMMANDE MK_LI_COMMANDE	HU_COMMANDE HU_LI_COMMANDE	KI_COMMANDE KI_LI_COMMANDE	BA_COMMANDE BA_LI_COMMANDE
Dimensions	MK_CLIENT	HU_DIM_COMMANDE HU_CLIENT	KI_CLIENT	BA_CLIENT

	MK_MAGASIN MK_VENDEUR MK_PRODUT MK_PERIODE MK_REGION_TEL MK_REGION_CP	HU_MAGASIN HU_VENDEUR HU_PRODUT HU_PERIODE HU_ENTREPOT HU_REGION_TEL HU_REGION_CP	KI_MAGASIN KI_VENDEUR KI_PRODUT KI_PERIODE KI_ENTREPOT	BA_VENDEUR BA_PRODUT BA_PERIODE BA_ENTREPOT
Mesures calculées	MK_DELAI MK_MONTANT_COMMANDE	HU_DELAI	KI_DELAI KI_MONTANT_COMMANDE KI_MONTANT_LIGNE KI_NB_LIGNES	BA_DELAI BA_MONTANT_COMMANDE BA_MONTANT_LIGNE BA_NB_LIGNES BA_MONTANT_MENSUEL BA_RABAI BA_MONTANT_SUGGERE

Tableau 6.2 : Tableau récapitulatif de la comparaison des démarches.

6.4. _____ Utilisation des modèles par Discoverer

Le Tableau 6.3 représente la charge de travail de l'administrateur de Discoverer pour répondre aux questions formulées par la liste des requêtes. Nous avons représenté d'une part le nombre de dossiers supplémentaires (« complex folders ») qu'il faut créer, le nombre champs calculés qu'il faut rajouter aux dossiers et le nombre de jointures qu'il faut effectuer entre les différents dossiers. La description de la construction de chacune des zones de travail (« work area ») qui ont été construits avec Discoverer Administrator se trouve dans le chapitre 5.

La création de ces zones de travail par Discoverer diffère en fonction des modèles qui ont été construits. Ainsi, on constate que la démarche de Moody et Kortink, qui ne prend en compte que le modèle entité-relation des données, nécessite une plus grande charge de travail de l'administrateur de Discoverer. Ainsi, en appliquant cette démarche, il faudra créer trois « complex folders » et quatre champs calculés. Il faudra, de plus, effectuer 18 jointures entre les tables. Dans le cas de Hüsemann, toutes les requêtes pourront être effectuées mais la charge de travail est relativement la même que celle qui a été nécessaire avec les tables du modèle de Moody et Kortink. Le modèle développé en suivant la démarche de Kimball, qui se base rappelons-le sur l'analyse des processus d'affaire, nécessite la création de deux « complex folders », de deux champs calculés et de 15 jointures entre les différents dossiers. Finalement, le modèle construit selon Ballard est celui qui demande la plus faible charge de travail. En effet, les tables contiennent déjà tous les champs et toutes les étoiles nécessaires pour répondre aux requêtes ont été développées. Il ne faut effectuer que les 10 jointures entre les différents dossiers de Discoverer Administrator.

	Moody et Kortink	Hüsemann	Kimball	Ballard
Création de « complex folder »	3	3	2	0
Champ calculé	5	6	3	0
Jointures	18	14	15	10

Tableau 6.3 : Nombre de « complex folders », de champs calculés et de jointures que l'administrateur de Discoverer doit effectuer.

On constate que le modèle développé en suivant la démarche de Ballard se prête le mieux aux analyses prévues dans les requêtes mais il faut remarquer qu'il a justement été développé en tenant compte de la liste des requêtes. Il n'est donc pas surprenant qu'il soit particulièrement bien adapté.

7. Conclusions

Le présent travail de mémoire nous a permis de comparer différentes démarches de conception d'entrepôt de données et d'évaluer la capacité des modèles multi dimensionnels, développés au cours de l'utilisation de ces démarches, à répondre à une liste de requêtes.

Nous avons remarqué tout d'abord que les entrepôts de données ne peuvent pas être construits de la même manière que les bases de données opérationnelles. L'analyse multidimensionnelle des entrepôts de données au moyen des outils OLAP est facilitée s'ils sont bâtis selon un modèle multidimensionnel. L'étude de la littérature sur le domaine nous montre que les concepts de base de la modélisation multidimensionnelle sont bien reconnus et que les modèles multi dimensionnels sont toujours construits avec des éléments identiques qui sont les faits, les dimensions et les mesures.

Toutefois, la démarche de conception des modèles multidimensionnels n'est pas unanimement admise et plusieurs procédures sont proposées dans la littérature. Ces démarches diffèrent par la manière de déterminer les faits, les dimensions et les mesures et par l'ordre dans lequel ces éléments sont établis.

Dans le cadre de ce travail, nous avons choisi quatre démarches de conception en fonction de la manière et de l'ordre de détermination des éléments du modèle multi dimensionnel. Ces démarches ont été appliquées aux données simulées d'une entreprise de grossiste en produits électroniques qui distribue des articles à des magasins.

Nous avons obtenu quatre modèles multi dimensionnels sur lesquels nous avons appliqué une série d'analyses au moyen de requêtes SQL et du logiciel Discoverer de la compagnie Oracle. Nous avons pu ainsi comparer les modèles et les tâches que l'administrateur de Discoverer doit effectuer pour que les analyses soient possibles. La comparaison a été effectuée en fonction de la disponibilité de modèles entité-relation, de l'importance de leur qualité et de l'implication des utilisateurs au cours du processus de développement de l'entrepôt de données.

Tout d'abord, nous avons trouvé que les modèles multi dimensionnels qui ont été développés en suivant les quatre démarches sont relativement proches les uns des autres. Cette constatation n'est pas très surprenante dans la mesure où le modèle original des données a été conçu pour l'analyse des commandes et des lignes de commande ce qui assure que le modèle entité-relation est lui-même déjà proche de celui de l'entrepôt de données. De plus, Tarigomi, utilisé en exemple, ne contient qu'une seule source de données. Dans un cas réel de développement d'entrepôt de données, les données proviennent de plusieurs

sources qui ne sont pas forcément cohérentes entre elles et il n'existe souvent pas de modèle entité-relation de ces différentes sources.

Toutefois, un des modèles développés n'a pas une granularité suffisante pour répondre à l'ensemble des requêtes. En effet, la démarche qui a été utilisée pour construire ce modèle ne tient pas compte de la granularité nécessaire durant l'analyse et celle-ci est déterminée uniquement par celle des dimensions.

Le travail a aussi permis de déterminer que la charge de travail de l'administrateur de Discoverer est plus faible lorsque les modèles ont été développés en tenant compte des besoins des utilisateurs et de l'analyse des processus d'affaire.

En fonction des résultats de notre travail, le choix de la démarche doit être adapté à la situation. En effet, s'il n'est pas possible de compter sur la participation des utilisateurs à cause de leur éloignement ou encore à cause de la nature du domaine des affaires, il est possible d'utiliser la démarche proposée par Moody et Kortink. Il faut toutefois prêter une grande attention à la qualité des modèles entité-relation dans la mesure où le développement de l'entrepôt de données repose uniquement sur celui-ci. Il faut aussi avoir conscience que la granularité de l'entrepôt de données peut empêcher certaines analyses.

Lorsqu'il existe un modèle entité-relation des données de l'entreprise et qu'il est possible de faire appel aux utilisateurs pour dresser une liste de requêtes ou un tableau des rôles des attributs de ce modèle, il sera possible d'utiliser les démarches proposées par Husemann ou pas Ballard. Si la liste des requêtes est très fiable et qu'elle n'est pas amenée à changer, notre travail a montré que la démarche de Ballard conduit à un modèle qui nécessite une charge de travail plus faible de l'administrateur de Discoverer.

Dans les cas où il n'existe pas de modèle entité-relation des données de l'entreprise, il sera possible d'appliquer la démarche proposée par Kimball qui fait appel à l'analyse des processus d'affaire.

8. Références

- Ballard Chuck, Herreman Dirk, Schau Don, Bell Rhonda, Kim Eunsang, Data Modeling Techniques for Data Warehousing, IBM Corporation, International Technical Support Organization 216 p., 1998.
- Barquin R.C., A data Warehouse Manifesto, in R. Barquin & H. Edelstein (Eds) Planning and Designing the Data Warehouse, pp.3 - 16, 1997.
- Barquin R.C., Paller A., Edelstein H., Ten Mistakes to Avoid for Data Warehousing Managers, in R. Barquin & H. Edelstein (Eds) Planning and Designing the Data Warehouse, pp.145 - 156, 1997.
- Codd E.F., Codd S.B., Salley C.T., Providing OLAP to User-Analysts: An IT Mandate, Hyperion 24 p., 1993.
- Devlin B., Data warehouse, from architecture to implementation, Addison-Wesley Longman Inc. 0 p., 1997.
- Few Stephen C., Achieving Data Warehouse Success: How to Avoid a Data Warehouse Meltdown, Brio Technology, Inc. 0 p., 1999.
- Fosdick H., Real-World Data Warehousing: the Management Challenge, in J. Bischof & T. Alexander. Data Warehouse: Practical advice from the Experts, pp.71 - 81, 1997.
- Hüsemann B., Lechtenbörger J., Vossen G., Conceptual Data Warehouse Design, Proc. International Workshop on Design and Management of Data Warehouses (DMDW) no 6, pp.1 - 11, 2000.
- Inmon W. H., Building the Data Warehouse, QED Information Sciences, Welleseley (MA) 0 p., 1993.
- Inmon W. H., OLAP and Data Warehouse, William H. Inmon 24 p., 2000.
- Kimball R., The Data Warehouse Toolkit, John Wiley&Sons, Inc. 216 p., 1996.
- Kosar D., The Seven deadly Sins, in J. Bischof & T. Alexander. Data Warehouse: Practical advice from the Experts, pp.57 - 70, 1997.
- Ladley J., A Flexible approach to Developing a Data Warehouse, in J. Bischof & T. Alexander. Data Warehouse: Practical advice from the Experts, pp.100 - 119, 1997.

-
- McGuff Frank, Designing the Perfect Data Warehouse (Data Modeling for Data Warehouses), knowledgecenters 75 p., 1998.
- Meta Group , Data warehouse scorecard : cost of ownership and successes in application of data warehouse technology, META Group , 0 p., 1999.
- Mimno Pieter R., Build Your Data Warehouse Right The First Time, Brio Technology, Inc. 0 p., 1999.
- Moody D.L., Kortink M.A.R., From Enterprise Models to Dimensional Models: a Methodology for Data Warehouse and Data Mart Design, Proc. International Workshop on Design and Management of Data Warehouses (DMDW) no 5, pp.1 - 12, 2000.
- Moorman M., The Art of Designing HOLAP Databases, SAS 5 p., 2000.
- Pedersen T.B., Jensen C.S, Multidimensional Data Modeling for complex Data, Proc. ICDE no 15, pp.1 - 27, 1999.
- Raden N., Choosing the right OLAP technology, in R. Barquin & H. Edelstein (Eds) Planning and Designing the Data Warehouse, pp.199 - 224, 1997.
- Vassiliadis P., Modeling Multidimensional Databases, Cubes and Cube operations Formal Foundations for Multidimensional Databases, Proc. SSDBM no 10, pp.1 - 39, 1998.
- White C., Multidimensional OLAP (MOLAP) versus relational OLAP (ROLAP), in J. Bischof & T. Alexander. Data Warehouse: Practical advice from the Experts, pp.199 - 205, 1997.
- Wu Ming-Chuan, Buchmann A. P., Research Issues in Data Warehousing, Proc. BTW no 0, pp.61 - 82, 1997.

9. Annexes

9.1. Discoverer

La présente annexe fournit une brève description de Discoverer pour permettre au lecteur de saisir l'essentiel du logiciel nécessaire à la compréhension du texte. Il est évident que cette annexe ne prétend pas fournir une description précise et détaillée de toutes les fonctionnalités offertes par le logiciel. On se référera à la documentation du logiciel pour des informations plus complètes.

Discoverer est un logiciel Oracle d'aide à la prise de décision qui permet de faire des requêtes ad hoc sur une base de données, d'analyser et de mettre en forme les résultats des requêtes, de préparer la présentation des résultats et de gérer les données du domaine des affaires d'une manière pertinente et représentative. Ce logiciel sépare les tâches d'administration d'une base de données de celles qui concernent les requêtes des rapports. Il permet aux utilisateurs d'effectuer le travail d'analyse sans avoir besoin de connaissance sur les bases de données ou de SQL.

Discoverer est formé de deux parties :

La partie utilisateur (User Edition) permet de faire facilement des requêtes sur une base de données, d'analyser les résultats et de mettre en forme un rapport,

La partie d'administration (Administration Edition) qui permet de créer une couche de métadonnées qui cache la complexité de la base de données aux utilisateurs et reflète un domaine particulier des affaires (« business area »).

Un « business area » dans Discoverer est une vue de tables et/ou d'éléments que l'administrateur de Discoverer crée pour répondre aux besoins d'affaire des utilisateur. Lorsque l'administrateur crée le « business area », il sélectionne des tables et des objets d'une base de données qui apparaîtront, leur donne un nom qui soit simple à comprendre et donne aux utilisateurs un accès à ces éléments. Dans le « business area », l'administrateur contrôle quelles données sont accessibles, comment elles sont structurées, les privilèges des utilisateurs ainsi que plusieurs autres aspects des données vues dans la partie utilisateur.

Lorsque l'administrateur crée le « business area », il ne manipule pas les éléments de la base de données mais seulement la manière que ces éléments apparaissent aux utilisateurs. Cela permet d'avoir la

flexibilité de créer plusieurs vues pour satisfaire plusieurs utilisateurs différents. Cela permet aussi de présenter les données d'une manière plus représentative aux utilisateurs. Par exemple, il est possible de donner aux éléments d'un « business area » des noms que l'utilisateur peut comprendre à la place des noms relativement complexes utilisés dans la base de données. Il est possible aussi de créer des jointures qui permet aux utilisateurs d'accéder à des données qui ne sont pas reliées dans la base de données. Le « business area » peut aussi contenir des conditions, des calculs, du formatage et des hiérarchies pour améliorer les données accessibles par les utilisateurs.

Les tables de la base de données sont représentées sous la forme de dossier dans le « business area ». Le « business area » peut contenir des dossiers d'une ou plusieurs base de données, ce qui donne une plus grande flexibilité aux utilisateurs en leur donnant accès aux données dont ils ont besoin.

L'administrateur peut créer des jointures entre les dossiers afin de permettre de retrouver des informations provenant des différents dossiers dans la même requête. Dans une relation de jointure, une table ou une vue est définie comme l'élément maître et la seconde table ou vue est définie comme étant l'élément de détail. En général une relation de maître à détail est une relation de 1 à N. Les jointures doivent être définies par l'administrateur, l'utilisateur ne peut pas en rajouter.

L'administrateur peut aussi créer des dossiers complexes (« complex folder ») qui contiennent des champs de différents dossiers joints. Ces « custom folders » peuvent aussi contenir des instructions SQL pour fournir aux utilisateurs des données encore plus adaptées à leurs besoins.

Le « custom folder » fournit un mécanisme pour implanter n'importe quelle instruction SQL de type SELECT. Comme les autres dossiers, le « custom folder » doit être joint aux autres pour que ses données soit reliées aux autres dans le « business area ». Une fois que l'administrateur a créé le « complex folder », les instructions SQL et les champs nécessaires, le « complex folder » apparaît de la même manière à l'utilisateur qu'un dossier simple et l'utilisateur peut construire des requêtes sur les éléments du « complex folder » exactement de la même manière que sur ceux d'un dossier simple.

L'administrateur peut créer des conditions pour filtrer les données qui seront retrouvées par l'utilisateur en fonction de critère que l'administrateur définit. Ces conditions permettent de sauver du temps à l'utilisateur en ne récupérant que les données nécessaires à l'analyse.

L'administrateur peut encore créer des hiérarchies qui sont des relations définies entre les éléments du « business area ». Ces relations permettront de détailler (« drill down ») ou de regrouper (« roll up ») les données dans différents niveaux. Par exemple, il est possible de relier les champs pays, région, district et ville en créant une hiérarchie géographique. Chaque élément de niveau supérieur (comme pays par

exemple) est un ensemble des éléments qui lui sont inférieurs (région). Lorsque l'administrateur bâtit une hiérarchie géographique, l'utilisateur peut construire une feuille d'analyse qui représente les ventes par pays et ensuite détailler l'analyse par région et ainsi de suite pour aboutir au niveau du magasin. Les hiérarchies ne sont pas définies dans la base de données mais dans le « business area ».

Les tables d'agrégation contiennent des données qui sont pré-résumées et pré-jointes. Elles contiennent les résultats de requêtes qui sont effectuées fréquemment avec des agrégations qui peuvent prendre beaucoup de temps lors de leur première exécution. Une requête effectuée sur une table d'agrégation peut ne prendre que quelques secondes alors que la requête initiale peut nécessiter la jointure de plusieurs tables et la sélection de plusieurs milliers ou millions de lignes et durer considérablement plus longtemps. Le processus de requête est transparent pour l'utilisateur qui ne se rend pas compte si sa requête est effectuée sur une table d'agrégation ou sur une table de base.

9.2. Données simplifiées

Nous avons créé des données simplifiées pour 15 commandes passées par les clients 1001 et 1002 qui possèdent chacun deux magasins (Tableau 9.1).

Commande	Vendeur	Client	Magasin	Date commande	Date Livraison	No produit	Quantité	Prix	Prix suggéré	Marque
1	2	1001	1	12-janv-90	12-janv-90	1001	2	40\$	40,95\$	Candle
						1002	3	40\$	45,95\$	Sony
						1003	2	25\$	32,95\$	Candle
2	1	1001	1	25-janv-90	26-janv-90	1001	10	25\$	40,95\$	Candle
3	2	1001	1	8-janv-91	10-janv-91	1001	15	35\$	40,95\$	Candle
						1002	15	30\$	45,95\$	Sony
						1003	21	25\$	32,95\$	Candle
4	1	1001	1	23-janv-92	26-janv-92	1005	40	50\$	51,95\$	Candle
5	2	1001	1	27-janv-92	27-janv-92	1009	10	200\$	200,95\$	Candle
6	1	1001	2	12-janv-90	13-janv-90	1003	8	25\$	32,95\$	Candle
7	2	1001	2	15-janv-90	17-janv-90	1002	2	40\$	45,95\$	Sony
						1003	4	30\$	32,95\$	Candle
8	1	1001	2	17-janv-90	20-janv-90	1005	4	50\$	51,95\$	Candle
9	2	1001	2	22-janv-90	24-janv-90	1001	10	40\$	40,95\$	Candle
10	1	1001	2	29-janv-90	29-janv-90	1003	20	30\$	32,95\$	Candle
11	2	1001	2	4-janv-92	5-janv-92	1001	6	35\$	40,95\$	Candle
						1002	8	30\$	45,95\$	Sony
						1003	6	25\$	32,95\$	Candle
12	1	1001	2	18-janv-92	19-janv-92	1001	10	35\$	40,95\$	Candle
						1002	10	30\$	45,95\$	Sony
						1003	6	25\$	32,95\$	Candle
13	2	1002	1	12-févr-90	12-févr-90	1001	2	40\$	40,95\$	Candle
						1002	3	40\$	45,95\$	Sony
						1003	2	25\$	32,95\$	Candle
14	1	1002	1	25-févr-90	26-févr-90	1001	10	25\$	40,95\$	Candle
15	2	1002	1	8-févr-91	10-févr-91	1001	15	35\$	40,95\$	Candle
						1002	15	30\$	45,95\$	Sony
						1003	21	25\$	32,95\$	Candle
16	1	1002	1	23-févr-92	26-févr-92	1005	40	50\$	51,95\$	Candle
17	2	1002	1	27-févr-92	27-févr-92	1009	10	200\$	200,95\$	Candle
18	1	1002	2	12-févr-90	13-févr-90	1003	4	25\$	32,95\$	Candle
19	2	1002	2	15-févr-90	17-févr-90	1002	1	40\$	45,95\$	Sony
						1003	2	30\$	32,95\$	Candle

Tableau 9.1 : Liste des données simplifiées.

9.3. Requêtes

9.3.1. Valeur moyenne des montants mensuels totaux par client.

La première requête fait l'analyse des montants mensuels moyens pour chaque client. La valeur moyenne commandée est calculée entre les mois identiques d'une année à l'autre en prenant la valeur totale commandée pendant un mois de l'année par client. Le Tableau 9.2 représente le détail des commandes pour chaque client et la valeur moyenne pour chaque mois.

Client	Date commande	No produit	Quantité	Prix	Montant ligne	Montant commande	Montant pour le mois et l'année	Montant moyen pour le mois
1001	12-janv-90	1001	2	40 \$	80 \$			
		1002	3	40 \$	120 \$			
		1003	2	25 \$	50 \$	250 \$		
1001	25-janv-90	1001	10	25 \$	250 \$	250 \$		
1001	15-janv-90	1003	4	30 \$	120 \$			
		1002	2	40 \$	80 \$	200 \$		
1001	17-janv-90	1005	4	50 \$	200 \$	200 \$		
1001	12-janv-90	1003	8	25 \$	200 \$	200 \$		
	Total Jan 90						1 100 \$	
1001	08-janv-91	1001	15	35 \$	525 \$			
		1002	15	30 \$	450 \$			
		1003	21	25 \$	525 \$	1 500 \$		
1001	22-janv-91	1001	10	40 \$	400 \$	400 \$		
1001	29-janv-91	1003	20	30 \$	600 \$	600 \$		
	Total Jan 91						2 500 \$	
1001	23-janv-92	1005	40	50 \$	2 000 \$	2 000 \$		
1001	27-janv-92	1009	10	200 \$	2 000 \$	2 000 \$		
1001	04-janv-92	1001	6	35 \$	210 \$			
		1002	8	30 \$	240 \$			
		1003	6	25 \$	150 \$	600 \$		
1001	18-janv-92	1001	10	35 \$	350 \$			
		1002	10	30 \$	300 \$			
		1003	6	25 \$	150 \$	800 \$		
	Total Jan 92						5 400 \$	
	Moyenne de Janvier							3 000 \$
1002	12-févr-90	1001	2	40 \$	80 \$			
		1002	3	40 \$	120 \$			
		1003	2	25 \$	50 \$	250 \$		
1002	25-févr-90	1001	10	25 \$	250 \$	250 \$		
1002	12-Fév-90	1003	4	25 \$	100 \$	100 \$		
1002	15-Fév-90	1003	2	30 \$	60 \$			
		1002	1	40 \$	40 \$	100 \$		

Total F��v 90		700 \$				
1002	08-f��vr-91	1001	15	35 \$	525 \$	
		1002	15	30 \$	450 \$	
		1003	21	25 \$	525 \$	1 500 \$
Total F��v 91		1 500 \$				
1002	23-f��vr-92	1005	40	50 \$	2 000 \$	2 000 \$
1002	27-f��vr-92	1009	10	200 \$	2 000 \$	2 000 \$
Total F��v 92		4 000 \$				
Moyenne de F��vrier		2 066.67 \$				

Tableau 9.2 : Donn  es simples pour le calcul du mensuel moyen de chaque commande.

Le r  sultat de l'analyse donne le tableau suivant:

Client	Jan	Fev	Mars
1001	3 000,00 \$	- \$	
1002	- \$	2 066,67 \$	

Tableau 9.3 : R  sultat de la requ  te no1.

9.3.2. Nombre de commandes et ventes totales par vendeur, par client, par mois, par ann  e

La deuxi  me requ  te fournit le nombre de commandes et le montant total des ventes pour chaque client, pour chaque vendeur, pour tous les mois et pour toutes les ann  es().

Vendeur	Client	Date commande	Quantit��	Prix	Montant ligne	Montant commande	Nombre de commandes	Montant pour le mois
1	1001	25-janv-90	10	25 \$	250 \$	250 \$		
1	1001	12-janv-90	8	25 \$	200 \$	200 \$		
1	1001	17-janv-90	4	50 \$	200 \$	200 \$		
		Janvier 90					3	650 \$
1	1001	29-janv-91	20	30 \$	600 \$	600 \$		
		Janvier 91					1	600 \$
1	1001	18-janv-92	10	35 \$	350 \$			
			10	30 \$	300 \$			
			6	25 \$	150 \$	800 \$		
1	1001	23-janv-92	40	50 \$	2 000 \$	2 000 \$		
		Janvier 92					2	2800 \$
1	1002	25-f��vr-90	10	25 \$	250 \$	250 \$		
1	1002	12-F��v-90	4	25 \$	100 \$	100 \$		
		F��vrier 90					2	350 \$
1	1002	23-f��vr-92	40	50 \$	2 000 \$	2 000 \$		

Février 92						1	2 000\$
2	1001	15-janv-90	4	30 \$	120 \$	200 \$	
			2	40 \$	80 \$		
2	1001	12-janv-90	2	40 \$	80 \$		
			3	40 \$	120 \$		
			2	25 \$	50 \$	250 \$	
Janvier 90						2	450\$
2	1001	08-janv-91	15	35 \$	525 \$	1 500 \$	
			15	30 \$	450 \$		
			21	25 \$	525 \$		
2	1001	22-janv-91	10	40 \$	400 \$	400 \$	
Janvier 91						2	1 900 \$
2	1001	27-janv-92	10	200 \$	2 000 \$	2 000 \$	
2	1001	04-janv-92	6	35 \$	210 \$	600 \$	
			8	30 \$	240 \$		
			6	25 \$	150 \$		
Janvier 91						2	2 600 \$
2	1002	12-févr-90	2	40 \$	80 \$	250 \$	
			3	40 \$	120 \$		
			2	25 \$	50 \$		
2	1002	15-Fév-90	2	30 \$	60 \$	100 \$	
			1	40 \$	40 \$		
Février 90						2	350 \$
2	1002	08-févr-91	15	35 \$	525 \$	1 500 \$	
			15	30 \$	450 \$		
			21	25 \$	525 \$		
Février 91						1	1 500 \$
2	1002	27-févr-92	10	200 \$	2 000 \$	2 000 \$	
Février 92						1	2 000 \$

Tableau 9.4 : Données simplifiées organisées pour calculer le nombre de commandes et ventes totales par vendeur, par client, par mois, par année.

Le résultat de la requête fournit le tableau suivant:

Vendeur	Client	90		91		92	
		Janvier	Février	Janvier	Février	Janvier	Février
		Nb Valeur	Nb Valeur	Nb Valeur	Nb Valeur	Nb Valeur	Nb Valeur
1	1001	3 650	0 0	1 600	0 0	2 2800	0 0
	1002	0 0	2 350	0 0	0 0	0 0	1 2000
2	1001	2 450	0 0	2 1900	0 0	2 2600	0 0
	1002	0 0	2 350	0 0	1 1500	0 0	1 2000

Tableau 9.5 : Résultats de la requête.

9.3.3. Délai moyen de livraison, par client, par mois, par année

La troisième requête calcule le délai de livraison moyen par client, par mois et par année. Dans le cas des données simplifiées, les commandes ont été livrées 0, 1,2 et 3 jours après la date de prise de commande. Dans le cas des commandes passées par le client 1001 pendant le mois de janvier 1990 par exemple, le délai total est de 7 jours et il y a eu 5 commandes. Le délai moyen a donc été de 1.4 jours en janvier 1990.

Client	Date commande	Date livraison	Délai	Délai moyen
1001	12-janv-90	13-janv-90	1	1.4
1001	15-janv-90	17-janv-90	2	
1001	25-janv-90	26-janv-90	1	
1001	12-janv-90	12-janv-90	0	
1001	17-janv-90	20-janv-90	3	
	Janvier 90			
1001	08-janv-91	10-janv-91	2	1.33
1001	22-janv-91	24-janv-91	2	
1001	29-janv-91	29-janv-91	0	
	Janvier 91			
1001	18-janv-92	19-janv-92	1	
1001	04-janv-92	05-janv-92	1	1.25
1001	23-janv-92	26-janv-92	3	
1001	27-janv-92	27-janv-92	0	
	Janvier 92			
1002	12-févr-90	12-févr-90	0	1.0
1002	12-Fév-90	13-Fév-90	1	
1002	15-Fév-90	17-Fév-90	2	
1002	25-févr-90	26-févr-90	1	
	Février 90			
1002	08-févr-91	10-févr-91	2	2.0
	Février 91			
1002	23-févr-92	26-févr-92	3	
1002	27-févr-92	27-févr-92	0	
	Février 92			
				1.5

Tableau 9.6 : Données simplifiées organisées pour le calcul du délai moyen de livraison, par client, par mois, par année.

Le résultat de la requête sur les données simplifiées est le suivant

Client	90		91		92	
	Janvier	Février	Janvier	Février	Janvier	Février
1001	1,4	0	1,33	0	1,25	0
1002	0	1	0	2	0	1,5

Tableau 9.7 : Résultats de la requête.

9.3.4. Nombre moyen de lignes de commande par commande pour chaque vendeur, pour chaque mois de toutes les années confondues

Cette requête permet de calculer la charge de travail pour chaque vendeur en fonction du mois de l'année.

Vendeur	Date commande	Nombre de ligne de commande	Nombre total de lignes de commandes	Nombre moyen de lignes de commandes
1	25-janv-90	1		
1	23-janv-92	1		
1	12-janv-90	1		
1	17-janv-90	1		
1	29-janv-91	1		
1	18-janv-92	3		
	Janvier		8	1.33
1	25-févr-90	1		
1	23-févr-92	1		
1	12-Fév-90	1		
	Février		3	1.0
2	15-janv-90	2		
2	12-janv-90	3		
2	08-janv-91	3		
2	27-janv-92	1		
2	22-janv-91	1		
2	04-janv-92	3		
	Janvier		13	2.17
2	08-févr-91	3		
2	27-févr-92	1		
2	12-févr-90	3		
2	15-Fév-90	2		
	Février		9	2.25

Tableau 9.8 : Données simplifiées organisées pour calculer le nombre moyen de lignes de commande par commande pour chaque vendeur, pour chaque mois de toutes les années confondues.

Vendeur	Janvier	Février	Mars
1	1,33	1	
2	2,17	2,25	

Tableau 9.9 : Résultat de la requête.

9.3.5. Valeur commandée par marque par jour de la semaine en pour cent de la valeur totale des commandes

La requête suivante permet de calculer l'importance de la vente de chaque produit par jour de la semaine (Tableau 9.10).

Marque	jour	Montant commande	Montant total	Valeur en %
Candle	lundi	2 000 \$		
Candle	lundi	80 \$		
Candle	lundi	50 \$		
Candle	lundi	120 \$		
Candle	lundi	100 \$	2 350 \$	15.46%
Candle	mardi	525 \$		
Candle	mardi	525 \$		
Candle	mardi	400 \$		
Candle	mardi	600 \$	2 050 \$	13.49%
Candle	mercredi	200 \$	200 \$	1.32%
Candle	jeudi	250 \$		
Candle	jeudi	2 000 \$		
Candle	jeudi	2 000 \$		
Candle	jeudi	60 \$	4 310 \$	28.36%
Candle	vendredi	80 \$		
Candle	vendredi	50 \$		
Candle	vendredi	525 \$		
Candle	vendredi	525 \$		
Candle	vendredi	200 \$	1 380 \$	9.08%
Candle	samedi	210 \$		
Candle	samedi	150 \$		
Candle	samedi	350 \$		
Candle	samedi	150 \$	860 \$	5.66%
Candle	dimanche	250 \$		
Candle	dimanche	2 000 \$	2 250 \$	14.80%
Sony	lundi	120 \$		
Sony	lundi	80 \$	200 \$	1.31%
Sony	mardi	450 \$	450 \$	2.96%
Sony	jeudi	40 \$	40 \$	0.26%
Sony	vendredi	120 \$		
Sony	vendredi	450 \$	570 \$	3.75%
Sony	samedi	240 \$		
Sony	samedi	300 \$	540 \$	3.55%
Montant total			15 200 \$	100.00%

Tableau 9.10 : Données simplifiées organisées pour le calcul de la valeur commandée par marque par jour de la semaine en pour cent de la valeur totale des commandes.

Marque	Lundi	Mardi	Mercredi	Jeudi	Vendredi	Samedi	Dimanche	Total
Candle	15,46%	13,49%	1,32%	28,36%	9,08%	5,66%	14,80%	88,17%
Sony	1,31%	2,96%	0,00%	0,26%	3,75%	3,55%	0,00%	11,83%
	16,77%	16,45%	1,32%	28,62%	12,83%	9,21%	14,80%	100,00%

Tableau 9.11 : Résultats de la requête sur les données simplifiées.

9.3.6. Rabais en pour-cent par vendeur, par marque, par mois et par année

Cette requête permet de calculer le rabais consenti par chaque vendeur pour chaque marque, en fonction du mois et de l'année (Tableau 9.12).

Vendeur	Marque	Date commande	Quantité	Prix	Montant ligne	Prix suggéré	montant suggéré	montant total	rabais	rabais total	rabais total %
1	Candle	12-janv-90	8	25 \$	200 \$	32,95 \$	263,60 \$		63,60 \$		
1	Candle	17-janv-90	4	50 \$	200 \$	51,95 \$	207,80 \$		7,80 \$		
1	Candle	25-janv-90	10	25 \$	250 \$	40,95 \$	409,50 \$	880,90 \$	159,50 \$	230,90 \$	26,21%
1	Candle	12-févr-90	4	25 \$	100 \$	32,95 \$	131,80 \$		31,80 \$		
1	Candle	25-févr-90	10	25 \$	250 \$	40,95 \$	409,50 \$	541,30 \$	159,50 \$	191,30 \$	35,34%
1	Candle	29-janv-91	20	30 \$	600 \$	32,95 \$	659,00 \$	659,00 \$	59,00 \$	59,00 \$	8,95%
1	Candle	18-janv-92	10	35 \$	350 \$	40,95 \$	409,50 \$		59,50 \$		
1	Candle	18-janv-92	6	25 \$	150 \$	32,95 \$	197,70 \$		47,70 \$		
1	Candle	23-janv-92	40	50 \$	2 000 \$	51,95 \$	2 078,00 \$	2 685,20 \$	78,00 \$	185,20 \$	6,90%
1	Candle	23-févr-92	40	50 \$	2 000 \$	51,95 \$	2 078,00 \$	2 078,00 \$	78,00 \$	78,00 \$	3,75%
1	Sony	18-janv-92	10	30 \$	300 \$	45,95 \$	459,50 \$	459,50 \$	159,50 \$	159,50 \$	34,71%
2	Candle	12-janv-90	2	40 \$	80 \$	40,95 \$	81,90 \$		1,90 \$		
2	Candle	12-janv-90	2	25 \$	50 \$	32,95 \$	65,90 \$		15,90 \$		
2	Candle	15-janv-90	4	30 \$	120 \$	32,95 \$	131,80 \$	279,60 \$	11,80 \$	29,60 \$	10,59%
2	Candle	12-févr-90	2	40 \$	80 \$	40,95 \$	81,90 \$		1,90 \$		
2	Candle	12-févr-90	2	25 \$	50 \$	32,95 \$	65,90 \$		15,90 \$		
2	Candle	15-févr-90	2	30 \$	60 \$	32,95 \$	65,90 \$	213,70 \$	5,90 \$	23,70 \$	11,09%
2	Candle	8-janv-91	15	35 \$	525 \$	40,95 \$	614,25 \$		89,25 \$		
2	Candle	8-janv-91	21	25 \$	525 \$	32,95 \$	691,95 \$		166,95 \$		
2	Candle	22-janv-91	10	40 \$	400 \$	40,95 \$	409,50 \$	1 715,70 \$	9,50 \$	265,70 \$	15,49%
2	Candle	8-févr-91	15	35 \$	525 \$	40,95 \$	614,25 \$		89,25 \$		
2	Candle	8-févr-91	21	25 \$	525 \$	32,95 \$	691,95 \$	1 306,20 \$	166,95 \$	256,20 \$	19,61%
2	Candle	4-janv-92	6	35 \$	210 \$	40,95 \$	245,70 \$		35,70 \$		
2	Candle	4-janv-92	6	25 \$	150 \$	32,95 \$	197,70 \$		47,70 \$		
2	Candle	27-janv-92	10	200 \$	2 000 \$	200,95 \$	2 009,50 \$	2 452,90 \$	9,50 \$	92,90 \$	3,79%
2	Candle	27-févr-92	10	200 \$	2 000 \$	200,95 \$	2 009,50 \$	2 009,50 \$	9,50 \$	9,50 \$	0,47%
2	Sony	12-janv-90	3	40 \$	120 \$	45,95 \$	137,85 \$		17,85 \$		
2	Sony	15-janv-90	2	40 \$	80 \$	45,95 \$	91,90 \$	229,75 \$	11,90 \$	29,75 \$	12,95%
2	Sony	12-févr-90	3	40 \$	120 \$	45,95 \$	137,85 \$		17,85 \$		
2	Sony	15-févr-90	1	40 \$	40 \$	45,95 \$	45,95 \$	183,80 \$	5,95 \$	23,80 \$	12,95%
2	Sony	8-janv-91	15	30 \$	450 \$	45,95 \$	689,25 \$	689,25 \$	239,25 \$	239,25 \$	34,71%

2	Sony	8-févr-91	15	30 \$	450 \$	45,95 \$	689,25 \$	689,25 \$	239,25 \$	239,25 \$	34,71 %
2	Sony	4-janv-92	8	30 \$	240 \$	45,95 \$	367,60 \$	367,60 \$	127,60 \$	127,60 \$	34,71 %

Tableau 9.12 : Données simplifiées organisées pour le calcul du rabais en pour-cent par vendeur, par marque, par mois et par année.

Vendeur	Marque	90		91		92	
		Janvier	Février	Janvier	Février	Janvier	Février
1	Candle	26,21%	35,34%	8,95%	0,00%	6,90%	3,75%
1	Sony	0,00%	0,00%	0,00%	0,00%	34,71%	0,00%
2	Candle	10,59%	11,09%	15,49%	19,61%	3,79%	0,47%
2	Sony	12,95%	12,95%	34,71%	34,71%	34,71%	0,00%

Tableau 9.13 : Résultats de la requête sur les données simplifiées.

Le résultat de cette requête pourra être faux si le prix suggéré de vente a changé entre le moment où la commande a été passée et celui où l'analyse est effectuée. Pour garantir la justesse du calcul, il faut mémoriser l'évolution temporelle du prix suggéré.

9.3.7. Ventes totales par année, par client, par magasin, par marque.

Cette requête permet de calculer l'importance des ventes par client par année pour chaque marque (Tableau 9.14).

Client	Marque	Année	Quantité	Prix	Montant ligne	Montant année
1001	Candle	1990	2	40 \$	80 \$	
1001	Candle	1990	2	25 \$	50 \$	
1001	Candle	1990	10	25 \$	250 \$	
1001	Candle	1990	8	25 \$	200 \$	
1001	Candle	1990	4	30 \$	120 \$	
1001	Candle	1990	4	50 \$	200 \$	
1001	Candle	1990	10	40 \$	400 \$	
1001	Candle	1990	20	30 \$	600 \$	1 900 \$
1001	Candle	1991	15	35 \$	525 \$	
1001	Candle	1991	21	25 \$	525 \$	1 050 \$
1001	Candle	1992	40	50 \$	2 000 \$	
1001	Candle	1992	10	200 \$	2 000 \$	
1001	Candle	1992	6	35 \$	210 \$	
1001	Candle	1992	6	25 \$	150 \$	
1001	Candle	1992	10	35 \$	350 \$	
1001	Candle	1992	6	25 \$	150 \$	4 860 \$
1001	Sony	1990	3	40 \$	120 \$	
1001	Sony	1990	2	40 \$	80 \$	200 \$
1001	Sony	1991	15	30 \$	450 \$	450 \$

1001	Sony	1992	8	30 \$	240 \$	
1001	Sony	1992	10	30 \$	300 \$	540 \$
1002	Candle	1990	2	40 \$	80 \$	
1002	Candle	1990	2	25 \$	50 \$	
1002	Candle	1990	10	25 \$	250 \$	
1002	Candle	1990	4	25 \$	100 \$	
1002	Candle	1990	2	30 \$	60 \$	540 \$
1002	Candle	1991	15	35 \$	525 \$	
1002	Candle	1991	21	25 \$	525 \$	1 050 \$
1002	Candle	1992	40	50 \$	2 000 \$	
1002	Candle	1992	10	200 \$	2 000 \$	4 000 \$
1002	Sony	1990	3	40 \$	120 \$	
1002	Sony	1990	1	40 \$	40 \$	160 \$
1002	Sony	1991	15	30 \$	450 \$	450 \$

Tableau 9.14 : Données simplifiées organisées pour calculer le montant des ventes totales par année, par client, par magasin, par marque.

Client	Marque	1990	1991	1992
1001	Candle	1 900,00 \$	1 050,00 \$	4 860,00 \$
1001	Sony	200,00 \$	450,00 \$	540,00 \$
1002	Candle	540,00 \$	1 050,00 \$	4 000,00 \$
1002	Sony	160,00 \$	450,00 \$	- \$

Tableau 9.15 : Résultats de la requête sur les données simplifiées.

9.3.8. Prix moyen de vente effectif par produit par année, par mois

Cette requête permet de calculer le prix moyen effectif des produits qui ont été vendus par produit, par année et par mois (Tableau 9.16).

No produit	Date commande	Quantité	Prix	Montant ligne	Montant total	Somme des quantités	prix moyen
1001	25-janv-90	10	25 \$	250 \$			
1001	12-janv-90	2	40 \$	80 \$	330 \$	12	27,50 \$
1001	25-févr-90	10	25 \$	250 \$			
1001	12-févr-90	2	40 \$	80 \$	330 \$	12	27,50 \$
1001	8-janv-91	15	35 \$	525 \$			
1001	22-janv-91	10	40 \$	400 \$	925 \$	25	37,00 \$
1001	8-févr-91	15	35 \$	525 \$	525 \$	15	35,00 \$
1001	18-janv-92	10	35 \$	350 \$			
1001	4-janv-92	6	35 \$	210 \$	560 \$	16	35,00 \$
1002	12-janv-90	3	40 \$	120 \$			
1002	15-janv-90	2	40 \$	80 \$	200 \$	5	40,00 \$
1002	12-févr-90	3	40 \$	120 \$			
1002	15-févr-90	1	40 \$	40 \$	160 \$	4	40,00 \$
1002	8-janv-91	15	30 \$	450 \$	450 \$	15	30,00 \$

1002	8-févr-91	15	30 \$	450 \$	450 \$	15	30,00 \$
1002	18-janv-92	10	30 \$	300 \$			
1002	4-janv-92	8	30 \$	240 \$	540 \$	18	30,00 \$
1003	12-janv-90	8	25 \$	200 \$			
1003	12-janv-90	2	25 \$	50 \$			
1003	15-janv-90	4	30 \$	120 \$	370 \$	14	26,43 \$
1003	12-févr-90	4	25 \$	100 \$			
1003	12-févr-90	2	25 \$	50 \$			
1003	15-févr-90	2	30 \$	60 \$	210 \$	8	26,25 \$
1003	29-janv-91	20	30 \$	600 \$			
1003	8-janv-91	21	25 \$	525 \$	1 125 \$	41	27,44 \$
1003	8-févr-91	21	25 \$	525 \$	525 \$	21	25,00 \$
1003	18-janv-92	6	25 \$	150 \$			
1003	4-janv-92	6	25 \$	150 \$	300 \$	12	25,00 \$
1005	17-janv-90	4	50 \$	200 \$	200 \$	4	50,00 \$
1005	23-janv-92	40	50 \$	2 000 \$	2 000 \$	40	50,00 \$
1005	23-févr-92	40	50 \$	2 000 \$	2 000 \$	40	50,00 \$
1009	27-janv-92	10	200 \$	2 000 \$	2 000 \$	10	200,00 \$
1009	27-févr-92	10	200 \$	2 000 \$	2 000 \$	10	200,00 \$

Tableau 9.16 : Données simplifiées pour le calcul du prix moyen de vente effectif par produit par année, par mois.

Produit	90		91		92	
	Janvier	Février	Janvier	Février	Janvier	Février
1001	27,50 \$	27,50 \$	37,00 \$	35,00 \$	35,00 \$	- \$
1002	40,00 \$	40,00 \$	30,00 \$	30,00 \$	30,00 \$	-
1003	26,43 \$	26,25 \$	27,44 \$	25,00 \$	25,00 \$	-
1005	50,00 \$	-	-	-	50,00 \$	50,00 \$
1009	-	-	-	-	200,00 \$	200,00 \$

Tableau 9.17 : Résultats de la requête sur les données simplifiées.

9.4. Création des vues pour les modèles multi dimensionnels

9.4.1. Modèle de Moody et Kortink (2000)

```
CREATE VIEW MK_COMMANDE AS
SELECT
CO.NO_CLIENT ,
CO.NO_MAGASIN ,
CO.NO_VENDEUR ,
CO.DATE_COMMANDE ,
AVG (CO.DATE_LIVRAISON - CO.DATE_COMMANDE) AS MK_DELAI ,
SUM(DC.MONTANT) AS MK_MONTANT_COMMANDE
```

```
FROM
    COMMANDE CO,
MK_MONTANT_COMMANDE DC
WHERE
    CO.NO_COMMANDE=DC.NO_COMMANDE
GROUP BY
    CO.NO_CLIENT ,
    CO.NO_MAGASIN ,
CO.NO_VENDEUR ,
    CO.DATE_COMMANDE;

CREATE VIEW MK_LI_COMMANDE AS
SELECT
DC.NO_PRODUIT ,
    CO.NO_CLIENT ,
    CO.NO_MAGASIN ,
CO.NO_VENDEUR ,
    CO.DATE_COMMANDE ,
    SUM(DC.QTE_COMMANDEE) AS SUM_QTE_COMMANDEE,
    SUM(DC.QTE_EXPEDIEE) AS SUM_QTE_EXPEDIEE,
    SUM(DC.PRIX*DC.QTE_EXPEDIEE) / SUM(DC.QTE_EXPEDIEE) AS PRIX_MOYEN
FROM
    COMMANDE CO,DET_COMMANDE DC
WHERE
    CO.NO_COMMANDE=DC.NO_COMMANDE
AND DC.QTE_EXPEDIEE > 0
GROUP BY
DC.NO_PRODUIT ,
    CO.NO_CLIENT ,
    CO.NO_MAGASIN ,
CO.NO_VENDEUR ,
    CO.DATE_COMMANDE;

CREATE VIEW MK_PRODUIT AS
SELECT
NO_PRODUIT,
DESCRIPTION,
MARQUE,
PRIX_SUGGERE,
POINT_LIMITE,
QTE_A_COMMANDER,
    0 AS PRIX_ACHAT
FROM PRODUIT ;

CREATE VIEW MK_MAGASIN AS
SELECT
MA.NO_MAGASIN,
MA.NO_CLIENT,
MA.ADRESSE AS ADRESSE_MAGASIN,
MA.ACTIF,
MA.CODE_POSTAL AS CODE_POSTAL_MAGASIN,
MA.VILLE AS VILLE_MAGASIN,
MA.PROVINCE AS PROVINCE_MAGASIN,
MA.NO_TEL AS NO_TEL_MAGASIN,
MA.NO_FAX AS NO_FAX_MAGASIN,
EN.ADRESSE AS ADRESSE_ENTREPOT,
    EN.VILLE AS VILLE_ENTREPOT,
EN.PROVINCE AS PROVINCE_ENTREPOT,
EN.CODE_POSTAL AS CODE_POSTAL_ENTREPOT,
EN.NO_TEL AS NO_TEL_ENTREPOT,
```



```
EN.NO_FAX AS NO_FAX_ENTREPOT
FROM
MOD_MAGASIN MA,ENTREPOT EN
WHERE MA.NO_ENTREPOT=EN.NO_ENTREPOT;
```

```
CREATE VIEW MK_CLIENT AS
SELECT
    NO_CLIENT,
    NOM,
    ADRESSE_FACTURATION,
    CODE_POSTAL,
    VILLE,
    PROVINCE,
    NO_TEL,
    NO_FAX
FROM
MOD_CLIENT ;
```

```
CREATE VIEW MK_VENDEUR AS
SELECT
    NO_VENDEUR,
    NOM,
    PRENOM,
    NAS
FROM
VENDEUR ;
```

```
CREATE VIEW MK_PERIODE AS
SELECT *
FROM PERIODE ;
```

```
CREATE VIEW MK_REGION_TEL AS
SELECT *
FROM REGION_TEL ;
```

```
CREATE VIEW MK_REGION_CP AS
SELECT *
FROM REGION_CP ;
```

9.4.2. Modèle de Hüsemann et al (2000)

```
CREATE VIEW HU_COMMANDE AS
SELECT
    NO_COMMANDE,
    DATE_COMMANDE,
    DATE_LIVRAISON-DATE_COMMANDE AS HU_DELAI
FROM COMMANDE ;
```

```
CREATE VIEW HU_LI_COMMANDE AS
SELECT
    CO.NO_COMMANDE,
    CO.DATE_COMMANDE,
    DC.NO_PRODUIT,
    DC.QTE_COMMANDEE AS HU_QTE_COMMANDEE,
    DC.QTE_EXPEDIEE AS HU_QTE_EXPEDIEE,
    DC.PRIX AS HU_PRIX_VENTE_EFFECTIF
FROM COMMANDE CO, DET_COMMANDE DC
```

```
WHERE CO.NO_COMMANDE=DC.NO_COMMANDE
AND DC.QTE_EXPEDIEE > 0
;
```

```
CREATE VIEW HU_DIM_COMMANDE AS
SELECT
    NO_COMMANDE,
    NO_CLIENT,
    NO_MAGASIN,
    NO_VENDEUR,
    NO_REQUISITION,
    NOM_RECEVEUR
FROM COMMANDE ;
```

```
CREATE VIEW HU_CLIENT AS
SELECT
    NO_CLIENT,
    NOM,
    ADRESSE_FACTURATION,
    CODE_POSTAL,
    VILLE,
    PROVINCE,
    NO_TEL,
    NO_FAX
FROM
    MOD_CLIENT ;
```

```
CREATE VIEW HU_MAGASIN AS
SELECT
    NO_MAGASIN,
    NO_CLIENT,
    ADRESSE,
    ACTIF,
    CODE_POSTAL,
    VILLE,
    PROVINCE,
    NO_TEL,
    NO_FAX,
    NO_ENTREPOT
FROM
    MOD_MAGASIN;
```

```
CREATE VIEW HU_VENDEUR AS
SELECT
    NO_VENDEUR,
    NOM,
    PRENOM,
    NAS
FROM
    VENDEUR ;
```

```
CREATE VIEW HU_PRODUIT AS
SELECT
    NO_PRODUIT,
    MARQUE,
    DESCRIPTION,
    PRIX_SUGGERE AS PRIX_VENTE_SUGGERE,
    POINT_LIMITE,
    QTE_A_COMMANDER,
    0 AS PRIX_ACHAT
```

```
FROM
    PRODUIT ;
```

```
CREATE VIEW HU_ENTREPOT AS
SELECT
    NO_ENTREPOT,
    ADRESSE,
    VILLE,
    PROVINCE,
    CODE_POSTAL,
    NO_TEL,
    NO_FAX
FROM
    ENTREPOT ;
```

```
CREATE VIEW HU_PERIODE AS
SELECT *
FROM PERIODE ;
```

```
CREATE VIEW HU_REGION_CP AS
SELECT *
FROM REGION_CP;
```

```
CREATE VIEW HU_REGION_TEL AS
SELECT *
FROM REGION_TEL;
```

9.4.3. Modèle de Kimball (1996)

```
CREATE VIEW KI_COMMANDE AS
SELECT
    CO.NO_COMMANDE,
    CO.NO_CLIENT,
    CO.NO_MAGASIN,
    CO.NO_VENDEUR,
    CO.DATE_COMMANDE,
    CO.DATE_LIVRAISON-CO.DATE_COMMANDE AS KI_DELAI,
    COUNT(DC.NO_PRODUIT) AS KI_NB_LIGNES,
    SUM(DC.QTE_EXPEDIEE*DC.PRIX) AS KI_MONTANT_COMMANDE
FROM COMMANDE CO, DET_COMMANDE DC
WHERE CO.NO_COMMANDE=DC.NO_COMMANDE
AND DC.QTE_EXPEDIEE > 0
GROUP BY
    CO.NO_COMMANDE,
    CO.NO_CLIENT,
    CO.NO_MAGASIN,
    CO.NO_VENDEUR,
    CO.DATE_COMMANDE,
    CO.DATE_LIVRAISON
;

CREATE VIEW KI_LI_COMMANDE AS
SELECT
    DC.NO_COMMANDE,
    CO.NO_CLIENT,
```



```
CO.NO_MAGASIN,
DC.NO_PRODUIT,
CO.NO_VENDEUR,
CO.DATE_COMMANDE,
DC.QTE_COMMANDEE,
DC.QTE_EXPEDIEE,
DC.QTE_EXPEDIEE*DC.PRIX AS KI_MONTANT_LIGNE,
DC.PRIX AS PRIX_VENTE_EFFECTIF
FROM COMMANDE CO, DET_COMMANDE DC
WHERE CO.NO_COMMANDE=DC.NO_COMMANDE
AND DC.QTE_EXPEDIEE > 0
;
CREATE VIEW KI_PRODUIT AS
SELECT
    NO_PRODUIT,
    MARQUE,
    DESCRIPTION,
    PRIX_SUGGERE AS PRIX_VENTE_SUGGERE,
    POINT_LIMITE,
    QTE_A_COMMANDER,
    0 AS PRIX_ACHAT
FROM
    PRODUIT ;

CREATE VIEW KI_MAGASIN AS
SELECT
    MA.NO_MAGASIN,
    MA.NO_CLIENT,
    MA.ADRESSE AS ADRESSE_MAGASIN,
    MA.VILLE AS VILLE_MAGASIN,
    MA.PROVINCE AS PROVINCE_MAGASIN,
    MA.CODE_POSTAL AS CODE_POSTAL_MAGASIN,
    RP.REGION AS REGION_CP_MAGASIN,
    MA.NO_TEL AS NO_TEL_MAGASIN,
    RL.REGION AS REGION_TEL_MAGASIN,
    MA.NO_FAX AS NO_FAX_MAGASIN,
    RF.REGION AS REGION_FAX_MAGASIN,
    MA.ACTIF,
    MA.NO_ENTREPOT
FROM MOD_MAGASIN MA, REGION_TEL RL, REGION_CP RP, REGION_TEL RF
WHERE
    SUBSTR(MA.CODE_POSTAL,1,3)=RP.PREFIXE
AND
    SUBSTR(MA.NO_TEL,LENGTH(MA.NO_TEL)-7,3)=RL.PREFIXE
AND
    SUBSTR(MA.NO_FAX,LENGTH(MA.NO_FAX)-7,3)=RF.PREFIXE;

CREATE VIEW KI_CLIENT AS
SELECT
    CL.NO_CLIENT,
    CL.NOM AS NOM_CLIENT,
    CL.ADRESSE_FACTURATION,
    CL.VILLE AS VILLE_FACTURATION,
    CL.PROVINCE AS PROVINCE_FACTURATION,
    CL.CODE_POSTAL AS CODE_POSTAL_FACTURATION,
    RP.REGION AS REGION_CP_FACTURATION,
    CL.NO_TEL AS NO_TEL_FACTURATION,
    RL.REGION AS REGION_TEL_FACTURATION,
    CL.NO_FAX AS NO_FAX_FACTURATION,
    RF.REGION AS REGION_FAX_FACTURATION
FROM MOD_CLIENT CL, REGION_TEL RL, REGION_CP RP, REGION_TEL RF
WHERE
```

```

        SUBSTR(CL.CODE_POSTAL,1,3)=RP.PREFIXE
AND      SUBSTR(CL.NO_TEL,LENGTH(CL.NO_TEL)-7,3)=RL.PREFIXE
AND      SUBSTR(CL.NO_FAX,LENGTH(CL.NO_FAX)-7,3)=RF.PREFIXE;

CREATE VIEW KI_VENDEUR AS
SELECT
    NO_VENDEUR,
    NOM AS NOM_VENDEUR,
    PRENOM AS PRENOM_VENDEUR,
    NAS AS NAS_VENDEUR
FROM VENDEUR;

CREATE VIEW KI_ENTREPOT AS
SELECT
    NO_ENTREPOT,
    EN.ADRESSE AS ADRESSE_ENTREPOT,
    EN.VILLE AS VILLE_ENTREPOT,
    EN.PROVINCE AS PROVINCE_ENTREPOT,
    EN.CODE_POSTAL AS CODE_POSTAL_ENTREPOT,
    RP.REGION AS REGION_CP_FACTURATION,
    EN.NO_TEL AS NO_TEL_ENTREPOT,
    RL.REGION AS REGION_TEL_FACTURATION,
    EN.NO_FAX AS NO_FAX_ENTREPOT,
    RF.REGION AS REGION_FAX_FACTURATION
FROM ENTREPOT EN, REGION_TEL RL, REGION_CP RP, REGION_TEL RF
WHERE
    SUBSTR(EN.CODE_POSTAL,1,3)=RP.PREFIXE
AND      SUBSTR(EN.NO_TEL,LENGTH(EN.NO_TEL)-7,3)=RL.PREFIXE
AND      SUBSTR(EN.NO_FAX,LENGTH(EN.NO_FAX)-7,3)=RF.PREFIXE;

CREATE VIEW KI_PERIODE AS
SELECT *
FROM PERIODE ;

```

9.4.4. Modèle de Ballard et al (1998)

```

CREATE VIEW BA_COMMANDE AS
SELECT
    CO.NO_COMMANDE,
    NO_CLIENT,
    NO_MAGASIN,
    NO_VENDEUR,
    DATE_COMMANDE,
    DATE_LIVRAISON-DATE_COMMANDE AS BA_DELAI,
    SUM(QTE_EXPEDIEE*PRIX) AS BA_MONTANT_COMMANDE,
    COUNT(NO_PRODUIT) AS BA_NB_LIGNES
FROM COMMANDE CO, DET_COMMANDE DC
WHERE CO.NO_COMMANDE=DC.NO_COMMANDE
GROUP BY CO.NO_COMMANDE,
    NO_CLIENT,
    NO_MAGASIN,
    NO_VENDEUR,
    DATE_COMMANDE,
    DATE_BON,
    DATE_LIVRAISON,
    DATE_FACTURE,
    DATE_PAYEE

```

;

```
CREATE VIEW BA_COMMANDE_MOIS AS
SELECT
NO_CLIENT,
    TO_CHAR (DATE_COMMANDE, 'MONTH') AS MOIS,
    TO_CHAR (DATE_COMMANDE, 'YY') AS ANNEE,
    (SUM (BA_MONTANT_COMMANDE) ) AS BA_MONTANT_MENSUEL
FROM BA_COMMANDE
GROUP BY
    NO_CLIENT,
    TO_CHAR (DATE_COMMANDE, 'MONTH'),
    TO_CHAR (DATE_COMMANDE, 'YY');
```

```
CREATE VIEW BA_LI_COMMANDE AS
SELECT
    CO.NO_COMMANDE,
    DC.NO_PRODUIT,
    NO_CLIENT,
    NO_MAGASIN,
    NO_VENDEUR,
    DATE_COMMANDE,
    QTE_EXPEDIEE,
    PRIX,
    (DC.QTE_EXPEDIEE*( PR.PRIX_SUGGERE-DC.PRIX)) AS BA_RABAIS,
    (DC.QTE_EXPEDIEE*PR.PRIX_SUGGERE) AS BA_MONTANT_SUGGERE,
    (QTE_EXPEDIEE*PRIX) AS BA_MONTANT_LIGNE
FROM COMMANDE CO,DET_COMMANDE DC,PRODUIT PR
WHERE CO.NO_COMMANDE=DC.NO_COMMANDE
AND DC.NO_PRODUIT=PR.NO_PRODUIT
AND DC.QTE_EXPEDIEE > 0
;
```

```
CREATE VIEW BA_CLIENT AS
SELECT
    MA.NO_MAGASIN,
    MA.NO_CLIENT,
    MA.ADRESSE AS ADRESSE_MAGASIN,
    MA.VILLE AS VILLE_MAGASIN,
    MA.PROVINCE AS PROVINCE_MAGASIN,
    MA.CODE_POSTAL AS CODE_POSTAL_MAGASIN,
    MP.REGION AS REGION_CP_MAGASIN,
    MA.NO_TEL AS NO_TEL_MAGASIN,
    ML.REGION AS REGION_TEL_MAGASIN,
    MA.NO_FAX AS NO_FAX_MAGASIN,
    MF.REGION AS REGION_FAX_MAGASIN,
    MA.ACTIF,
    MA.NO_ENTREPOT,
    CL.NOM AS NOM_CLIENT,
    CL.ADRESSE_FACTURATION,
    CL.VILLE AS VILLE_FACTURATION,
    CL.PROVINCE AS PROVINCE_FACTURATION,
    CL.CODE_POSTAL AS CODE_POSTAL_FACTURATION,
    RP.REGION AS REGION_CP_FACTURATION,
    CL.NO_TEL AS NO_TEL_FACTURATION,
    RL.REGION AS REGION_TEL_FACTURATION,
    CL.NO_FAX AS NO_FAX_FACTURATION,
    RF.REGION AS REGION_FAX_FACTURATION
```



```
FROM MOD_CLIENT CL,MOD_MAGASIN MA, REGION_TEL RL, REGION_CP RP, REGION_TEL RF,
REGION_TEL ML, REGION_CP MP, REGION_TEL MF
WHERE CL.NO_CLIENT=MA.NO_CLIENT
AND SUBSTR(CL.CODE_POSTAL,1,3)=RP.PREFIXE
AND SUBSTR(CL.NO_TEL,LENGTH(CL.NO_TEL)-7,3)=RL.PREFIXE
AND SUBSTR(CL.NO_FAX,LENGTH(CL.NO_FAX)-7,3)=RF.PREFIXE
AND SUBSTR(MA.CODE_POSTAL,1,3)=MP.PREFIXE
AND SUBSTR(MA.NO_TEL,LENGTH(MA.NO_TEL)-7,3)=ML.PREFIXE
AND SUBSTR(MA.NO_FAX,LENGTH(MA.NO_FAX)-7,3)=MF.PREFIXE
;
```

```
CREATE VIEW BA_VENDEUR AS
SELECT *
FROM VENDEUR;
```

```
CREATE VIEW BA_PRODUIT AS
SELECT
    NO_PRODUIT,
    DESCRIPTION,
    MARQUE,
    PRIX_SUGGERE AS PRIX_VENTE_SUGGERE,
    POINT_LIMITE,
    QTE_A_COMMANDER,
    0 AS PRIX_ACHAT
FROM PRODUIT;
```

```
CREATE VIEW BA_ENTREPOT AS
SELECT *
FROM ENTREPOT;
```

```
CREATE VIEW BA_PERIODE AS
SELECT *
FROM PERIODE ;
```

9.5. Commandes SQL pour les requêtes

9.5.1. Sur le modèle selon MK

9.5.1.1. ____ Valeur moyenne des montants mensuels totaux, par client.

La valeur moyenne commandée est calculée à partir de la valeur totale commandée pendant un mois de l'année par client et par magasin et la moyenne est calculée entre les mois identiques d'une année à l'autre.

Tout d'abord on crée une vue pour obtenir la somme du montant des commandes par client, par magasin, par mois, par année.

```
CREATE VIEW MK_VENTES_CLIENTS_MOIS_AN AS
SELECT NO_CLIENT,
       TO_CHAR (DATE_COMMANDE, 'MM') AS MOIS,
       TO_CHAR (DATE_COMMANDE, 'YY') AS ANNEE,
       (SUM (SUM_QTE_EXPEDIEE * PRIX_MOYEN) ) AS MONTANT
FROM MK_LI_COMMANDE
GROUP BY
NO_CLIENT,
TO_CHAR (DATE_COMMANDE, 'MM') ,
TO_CHAR (DATE_COMMANDE, 'YY') ;
```

Ensuite on fait une deuxième requête sur la vue pour calculer la moyenne des montants des commandes par client, par année, par mois :

```
SELECT
NO_CLIENT,
MOIS,
AVG (MONTANT)
FROM MK_VENTES_CLIENTS_MOIS_AN
WHERE NO_CLIENT=1001
GROUP BY
NO_CLIENT,
MOIS;
```

9.5.1.2. ____ Nombre de commandes et ventes totales par vendeur, par client, par mois, par année

Requête impossible.

9.5.1.3. ____ Délai moyen de livraison, par client, par mois, par année

Requête impossible.

9.5.1.4. ____ Nombre moyen de lignes de commande par commande pour chaque vendeur, pour chaque mois de toutes les années confondues

Requête impossible.

9.5.1.5. ____ Valeur commandée par marque par jour de la semaine en pour cent de la valeur totale des commandes

On crée tout d'abord une vue pour calculer le montant commandé pour chaque marque pour chaque jour de la semaine.

```

CREATE VIEW MK_MONTANT_MARQUE_JOUR AS
SELECT
    PR.MARQUE,
    TO_CHAR(CO.DATE_COMMANDE, 'DAY') as JOUR,
    SUM(SUM_QTE_EXPEDIEE*PRIX_MOYEN) AS MONTANT_MA_JOUR
FROM MK_LI_COMMANDE CO, MK_PRODUT PR
WHERE CO.NO_PRODUT=PR.NO_PRODUT
GROUP BY
    PR.MARQUE,
    TO_CHAR(CO.DATE_COMMANDE, 'DAY') ;

```

Ensuite, on crée une vue pour calculer le montant total commandé.

```

CREATE VIEW MK_MONTANT_TOTAL AS
SELECT
    SUM(SUM_QTE_EXPEDIEE*PRIX_MOYEN) AS MONTANT_TOTAL
FROM MK_LI_COMMANDE;

```

Finalement, on fait une requête sur les deux vues pour calculer le pourcentage commandé pour chaque marque et pour chaque jour de la semaine.

```

SELECT
    VM.MARQUE,
    VM.JOUR,
    SUM(VM.MONTANT_MA_JOUR/VT.MONTANT_TOTAL*100) AS MONTANT
FROM MK_MONTANT_MARQUE_JOUR VM, MK_MONTANT_TOTAL VT
WHERE MARQUE='A.D.S'
GROUP BY
    VM.MARQUE,
    VM.JOUR;

```

9.5.1.6. ____ Rabais en pour-cent par vendeur, par marque, par mois et par année

```

SELECT
    CO.NO_VENDEUR,
    PR.MARQUE,
    TO_CHAR(CO.DATE_COMMANDE, 'YY') AS ANNEE,
    TO_CHAR(CO.DATE_COMMANDE, 'MM') AS MOIS,
    100*(1-SUM(SUM_QTE_EXPEDIEE*CO.PRIX_MOYEN)/SUM(SUM_QTE_EXPEDIEE*PR.PRIX_SUGGERE))
AS RABAIS
FROM MK_LI_COMMANDE CO, MK_PRODUT PR
WHERE CO.NO_PRODUT=PR.NO_PRODUT
AND (SUM_QTE_EXPEDIEE*PR.PRIX_SUGGERE) > 0
AND PR.MARQUE='A.D.S'
GROUP BY
    CO.NO_VENDEUR,
    PR.MARQUE,
    TO_CHAR(CO.DATE_COMMANDE, 'YY'),
    TO_CHAR(CO.DATE_COMMANDE, 'MM');

```


Le résultat de cette requête pourra être faux si le prix suggéré de vente a changé entre le moment où la commande a été passée et celui où l'analyse est effectuée. Pour garantir la justesse du calcul, il faut mémoriser l'évolution temporelle du prix suggéré.

9.5.1.7. ____ Ventes totales par année, par client, par marque.

```
SELECT
CO.NO_CLIENT,
MARQUE,
TO_CHAR (DATE_COMMANDE, 'YY'),
SUM (PRIX_MOYEN*SUM_QTE_EXPEDIEE)
FROM MK_LI_COMMANDE CO, MK_PRODUT PR
WHERE CO.NO_PRODUT=PR.NO_PRODUT
AND CO.NO_CLIENT=1001
AND MARQUE='A.D.S'
GROUP BY
CO.NO_CLIENT,
MARQUE,
TO_CHAR (DATE_COMMANDE, 'YY');
```

9.5.1.8. ____ Prix moyen de vente effectif par produit par année, par mois

```
SELECT
PR.MARQUE,
DC.NO_PRODUT,
TO_CHAR (DATE_COMMANDE, 'MM') AS MOIS,
TO_CHAR (DATE_COMMANDE, 'YY'),
SUM (SUM_QTE_EXPEDIEE*PRIX_MOYEN) / SUM (SUM_QTE_EXPEDIEE)
FROM MK_LI_COMMANDE DC, MK_PRODUT PR
WHERE DC.NO_PRODUT= PR.NO_PRODUT
AND PR.MARQUE='A.D.S'
GROUP BY
PR.MARQUE,
DC.NO_PRODUT,
TO_CHAR (DATE_COMMANDE, 'YY'),
TO_CHAR (DATE_COMMANDE, 'MM')
ORDER BY 1;
```

9.5.2. Sur le modèle de HU

9.5.2.1. ____ Valeur moyenne des montants mensuels totaux, par client.

La valeur moyenne commandée est calculée à partir de la valeur totale commandée pendant un mois de l'année par client et par magasin et la moyenne est calculée entre les mois identiques d'une année à l'autre.

Tout d'abord on crée une vue pour obtenir la somme du montant des commandes par client, par magasin, par mois, par année.

```
CREATE VIEW HU_VENTES_CLIENTS_MOIS_AN AS
SELECT DI.NO_CLIENT,
       TO_CHAR(DC.DATE_COMMANDE, 'MM') AS MOIS,
       TO_CHAR(DC.DATE_COMMANDE, 'YY') AS ANNEE,
       (SUM(DC.HU_QTE_EXPEDIEE*DC.HU_PRIX_VENTE_EFFECTIF) ) AS MONTANT
FROM HU_LI_COMMANDE DC, HU_DIM_COMMANDE DI
WHERE DI.NO_COMMANDE=DC.NO_COMMANDE
GROUP BY
DI.NO_CLIENT,
TO_CHAR(DC.DATE_COMMANDE, 'MM'),
TO_CHAR(DC.DATE_COMMANDE, 'YY');
```

Ensuite on fait une deuxième requête sur la vue pour calculer la moyenne des montants des commandes par client, par année, par mois :

```
SELECT
NO_CLIENT,
MOIS,
AVG(MONTANT)
FROM HU_VENTES_CLIENTS_MOIS_AN
WHERE NO_CLIENT=1001
GROUP BY
NO_CLIENT,
MOIS;
```

9.5.2.2. ____ Nombre de commandes et ventes totales par vendeur, par client, par mois, par année

Pour calculer le nombre de commandes et le montant des commandes, on crée tout d'abord une vue dans laquelle on calcule le montant de chaque commande. La requête SQL pour créer la vue est la suivante :

```
CREATE VIEW HU_MONTANT_PAR_COMMANDE AS
SELECT
DI.NO_COMMANDE,
DI.NO_VENDEUR,
       DI.NO_CLIENT,
       DI.NO_MAGASIN,
       DC.DATE_COMMANDE,
       SUM(DC.HU_QTE_EXPEDIEE*DC.HU_PRIX_VENTE_EFFECTIF) AS MONTANT_COMMANDE
FROM HU_LI_COMMANDE DC, HU_DIM_COMMANDE DI
WHERE DC.NO_COMMANDE=DI.NO_COMMANDE
GROUP BY
DI.NO_COMMANDE,
DI.NO_VENDEUR,
DI.NO_CLIENT ,
DI.NO_MAGASIN,
DC.DATE_COMMANDE ;
```

La requête SQL pour calculer le montant par vendeur, par client et par mois est la suivante :

```
SELECT      NO_VENDEUR,
            NO_CLIENT,
            TO_CHAR (DATE_COMMANDE, 'YY') ,
            TO_CHAR (DATE_COMMANDE, 'MONTH') AS MOIS,
            SUM (MONTANT_COMMANDE) ,
            COUNT (NO_COMMANDE)
FROM HU_MONTANT_PAR_COMMANDE
WHERE NO_VENDEUR =1
AND NO_CLIENT=1001
GROUP BY    NO_VENDEUR,
            NO_CLIENT ,
            TO_CHAR (DATE_COMMANDE, 'YY') ,
            TO_CHAR (DATE_COMMANDE, 'MONTH') ;
```

9.5.2.3. ____ Délai moyen de livraison, par client, par mois, par année

```
SELECT
    DI.NO_CLIENT,
    TO_CHAR (CO.DATE_COMMANDE, 'MM') ,
    TO_CHAR (CO.DATE_COMMANDE, 'YY') ,
    AVG (CO.HU_DELAI) AS DELAI,
    COUNT (CO.NO_COMMANDE)
FROM HU_COMMANDE CO, HU_DIM_COMMANDE DI
WHERE CO.NO_COMMANDE=DI.NO_COMMANDE
AND NO_CLIENT=1001
GROUP BY
    DI.NO_CLIENT,
    TO_CHAR (CO.DATE_COMMANDE, 'YY') ,
    TO_CHAR (CO.DATE_COMMANDE, 'MM') ;
```

9.5.2.4. ____ Nombre moyen de lignes de commande par commande pour chaque vendeur, pour chaque mois de toutes les années confondues

```
CREATE VIEW HU_NB_LI_VENDEUR_MOIS AS
SELECT
    DC.NO_COMMANDE,
    DI.NO_VENDEUR,
    TO_CHAR (DC.DATE_COMMANDE, 'MM') AS MOIS,
    COUNT (DC.NO_PRODUIT) AS NB_LI_MOIS
FROM HU_LI_COMMANDE DC, HU_DIM_COMMANDE DI
WHERE DC.NO_COMMANDE=DI.NO_COMMANDE
GROUP BY
    DC.NO_COMMANDE,
    DI.NO_VENDEUR,
    TO_CHAR (DC.DATE_COMMANDE, 'MM') ;
```



```

SELECT      NO_VENDEUR,
            MOIS,
            AVG(NB_LI_MOIS)
FROM HU_NB_LI_VENDEUR_MOIS
WHERE NO_VENDEUR=1
GROUP BY
NO_VENDEUR,
MOIS;

```

9.5.2.5. ____ Valeur commandée par marque par jour de la semaine en pour cent de la valeur totale des commandes

```

CREATE VIEW MK_MONTANT_TOTAL AS
SELECT
    SUM(SUM_QTE_EXPEDIEE*PRIX_MOYEN) AS MONTANT_TOTAL
FROM MK_LI_COMMANDE;

SELECT
    PR.MARQUE,
    TO_CHAR(CO.DATE_COMMANDE, 'DAY') as JOUR,
    SUM(SUM_QTE_EXPEDIEE*PRIX_MOYEN) / VT.MONTANT_TOTAL*100 AS MONTANT
FROM MK_LI_COMMANDE CO, MK_PRODUT PR, MK_MONTANT_TOTAL VT
WHERE CO.NO_PRODUT=PR.NO_PRODUT
AND MARQUE='A.D.S'
GROUP BY
    PR.MARQUE,
    TO_CHAR(CO.DATE_COMMANDE, 'DAY'),
    VT.MONTANT_TOTAL;

```

9.5.2.6. ____ Rabais en pour-cent par vendeur, par marque, par mois et par année

```

SELECT
    DI.NO_VENDEUR,
    PR.MARQUE,
    TO_CHAR(DC.DATE_COMMANDE, 'YY') AS ANNEE,
    TO_CHAR(DC.DATE_COMMANDE, 'MM') AS MOIS,
    100*(1-
    SUM(DC.HU_QTE_EXPEDIEE*DC.HU_PRIX_VENTE_EFFECTIF)/SUM(DC.HU_QTE_EXPEDIEE*PR.PRIX_VENTE
    _SUGGERE))
    AS RABAIS
FROM HU_LI_COMMANDE DC, HU_PRODUT PR, HU_DIM_COMMANDE DI
WHERE DC.NO_PRODUT=PR.NO_PRODUT
AND DC.NO_COMMANDE=DI.NO_COMMANDE
AND (DC.HU_QTE_EXPEDIEE*PR.PRIX_VENTE_SUGGERE) > 0
AND NO_VENDEUR=1
AND PR.MARQUE='A.D.S'
GROUP BY
    DI.NO_VENDEUR,
    PR.MARQUE,
    TO_CHAR(DC.DATE_COMMANDE, 'YY'),

```

```
TO_CHAR(DC.DATE_COMMANDE, 'MM');
```

9.5.2.7. ____ Ventes totales par année, par client, par marque.

```
SELECT
DI.NO_CLIENT,
PR.MARQUE,
TO_CHAR(DC.DATE_COMMANDE, 'YY'),
SUM(DC.HU_PRIX_VENTE_EFFECTIF*DC.HU_QTE_EXPEDIEE)
FROM HU_LI_COMMANDE DC, HU_PRODUIT PR, HU_DIM_COMMANDE DI
WHERE DC.NO_PRODUIT=PR.NO_PRODUIT
AND DC.NO_COMMANDE=DI.NO_COMMANDE
AND DI.NO_CLIENT=1001
AND PR.MARQUE='A.D.S'
GROUP BY
DI.NO_CLIENT,
PR.MARQUE,
TO_CHAR(DC.DATE_COMMANDE, 'YY');
```

9.5.2.8. ____ Prix moyen de vente effectif par produit par année, par mois

```
SELECT
PR.MARQUE,
DC.NO_PRODUIT,
TO_CHAR(DC.DATE_COMMANDE, 'MM') AS MOIS,
TO_CHAR(DC.DATE_COMMANDE, 'YY'),
SUM(DC.HU_QTE_EXPEDIEE*DC.HU_PRIX_VENTE_EFFECTIF)/SUM(DC.HU_QTE_EXPEDIEE)
FROM HU_LI_COMMANDE DC, HU_PRODUIT PR
WHERE DC.NO_PRODUIT= PR.NO_PRODUIT
AND DC.HU_QTE_EXPEDIEE>0
AND PR.MARQUE='A.D.S'
GROUP BY
PR.MARQUE,
DC.NO_PRODUIT,
TO_CHAR(DC.DATE_COMMANDE, 'YY'),
TO_CHAR(DC.DATE_COMMANDE, 'MM')
ORDER BY 1;
```

9.5.3. Sur le modèle de KI

9.5.3.1. ____ Valeur moyenne des montants mensuels totaux, par client.

```
CREATE VIEW KI_VENTES_CLIENTS_MOIS_AN AS
SELECT NO_CLIENT,
TO_CHAR(DC.DATE_COMMANDE, 'MM') AS MOIS,
TO_CHAR(DC.DATE_COMMANDE, 'YY') AS ANNEE,
SUM(KI_MONTANT_LIGNE) AS MONTANT
```

```

FROM KI_LI_COMMANDE
GROUP BY
NO_CLIENT,
TO_CHAR (DATE_COMMANDE, 'MM') ,
TO_CHAR (DATE_COMMANDE, 'YY') ;

SELECT
NO_CLIENT,
MOIS,
AVG (MONTANT)
FROM KI_VENTES_CLIENTS_MOIS_AN
WHERE NO_CLIENT=1001
GROUP BY
NO_CLIENT,
MOIS;

```

9.5.3.2. ____ Nombre de commandes et ventes totales par vendeur, par client, par mois, par année

```

SELECT      NO_VENDEUR,
            NO_CLIENT,
            TO_CHAR (DATE_COMMANDE, 'YY') ,
            TO_CHAR (DATE_COMMANDE, 'MONTH') AS MOIS,
            SUM (KI_MONTANT_COMMANDE) ,
            COUNT (NO_COMMANDE)
FROM KI_COMMANDE
WHERE NO_VENDEUR =1
AND NO_CLIENT=1001
GROUP BY    NO_VENDEUR,
            NO_CLIENT ,
            TO_CHAR (DATE_COMMANDE, 'YY') ,
            TO_CHAR (DATE_COMMANDE, 'MONTH') ;

```

9.5.3.3. ____ Délai moyen de livraison, par client, par mois, par année

```

SELECT
NO_CLIENT,
TO_CHAR (DATE_COMMANDE, 'MM') ,
TO_CHAR (DATE_COMMANDE, 'YY') ,
AVG (KI_DELAI) AS DELAI,
COUNT (NO_COMMANDE)
FROM KI_COMMANDE CO
WHERE NO_CLIENT=1001
GROUP BY
NO_CLIENT,
TO_CHAR (DATE_COMMANDE, 'YY') ,
TO_CHAR (DATE_COMMANDE, 'MM') ;

```

9.5.3.4. ____ Nombre moyen de lignes de commande par commande pour chaque vendeur, pour chaque mois de toutes les années confondues

La requête est effectuée avec les commandes SQL suivantes :

```
SELECT      NO_VENDEUR,
            TO_CHAR (DATE_COMMANDE, 'MM') AS MOIS,
            AVG (KI_NB_LIGNES) AS NB_LI_MOIS
FROM KI_COMMANDE
WHERE NO_VENDEUR=1
GROUP BY
            NO_VENDEUR,
            TO_CHAR (DATE_COMMANDE, 'MM');
```

9.5.3.5. ____ Valeur commandée par marque par jour de la semaine en pour cent de la valeur totale des commandes

```
CREATE VIEW KI_MONTANT_TOTAL AS
SELECT
            SUM (KI_MONTANT_COMMANDE) AS MONTANT_TOTAL
FROM KI_COMMANDE;

SELECT
            PR.MARQUE,
            TO_CHAR (DC.DATE_COMMANDE, 'DAY') as JOUR,
            SUM (DC.KI_MONTANT_LIGNE / VT.MONTANT_TOTAL*100) AS MONTANT
FROM KI_LI_COMMANDE DC, KI_PRODUIT PR, KI_MONTANT_TOTAL VT
WHERE DC.NO_PRODUIT=PR.NO_PRODUIT
AND PR.MARQUE='A.D.S'
GROUP BY
            PR.MARQUE,
            TO_CHAR (DC.DATE_COMMANDE, 'DAY'),
            VT.MONTANT_TOTAL;
```

9.5.3.6. ____ Rabais en pour-cent par vendeur, par marque, par mois et par année

```
SELECT
            DC.NO_VENDEUR,
            PR.MARQUE,
            TO_CHAR (DC.DATE_COMMANDE, 'YY') AS ANNEE,
            TO_CHAR (DC.DATE_COMMANDE, 'MM') AS MOIS,
            100 * (1 - SUM (DC.KI_MONTANT_LIGNE) / SUM (DC.QTE_EXPEDIEE * PR.PRIX_VENTE_SUGGERE))
AS RABAIS
FROM KI_LI_COMMANDE DC, KI_PRODUIT PR
WHERE DC.NO_PRODUIT=PR.NO_PRODUIT
AND (DC.QTE_EXPEDIEE * PR.PRIX_VENTE_SUGGERE) > 0
AND NO_VENDEUR=1
AND PR.MARQUE='A.D.S'
GROUP BY
            DC.NO_VENDEUR,
            PR.MARQUE,
            TO_CHAR (DC.DATE_COMMANDE, 'YY'),
```

```
TO_CHAR(DC.DATE_COMMANDE, 'MM');
```

9.5.3.7. ____ Ventes totales par année, par client, par marque.

```
SELECT
DC.NO_CLIENT,
PR.MARQUE,
TO_CHAR(DC.DATE_COMMANDE, 'YY'),
SUM(DC.KI_MONTANT_LIGNE)
FROM KI_LI_COMMANDE DC, KI_PRODUIT PR
WHERE DC.NO_PRODUIT=PR.NO_PRODUIT
AND DC.NO_CLIENT=1001
AND PR.MARQUE='A.D.S'
GROUP BY
DC.NO_CLIENT,
PR.MARQUE,
TO_CHAR(DC.DATE_COMMANDE, 'YY');
```

9.5.3.8. ____ Évolution du prix moyen de vente effectif par produit par année, par mois

```
SELECT
PR.MARQUE,
DC.NO_PRODUIT,
TO_CHAR(DC.DATE_COMMANDE, 'MONTH') AS MOIS,
TO_CHAR(DC.DATE_COMMANDE, 'YY'),
SUM(DC.KI_MONTANT_LIGNE)/SUM(DC.QTE_EXPEDIEE)
FROM KI_LI_COMMANDE DC, KI_PRODUIT PR
WHERE DC.NO_PRODUIT= PR.NO_PRODUIT
AND DC.QTE_EXPEDIEE>0
AND PR.MARQUE='A.D.S'
GROUP BY
PR.MARQUE,
DC.NO_PRODUIT,
TO_CHAR(DC.DATE_COMMANDE, 'YY'),
TO_CHAR(DC.DATE_COMMANDE, 'MONTH')
ORDER BY 1;
```

9.5.4. Sur le modèle de BA

9.5.4.1. ____ Valeur moyenne des montants mensuels totaux, par client.

```
SELECT NO_CLIENT,
MOIS,
AVG(BA_MONTANT_MENSUEL)
FROM BA_COMMANDE_MOIS
WHERE NO_CLIENT=1001
GROUP BY
NO_CLIENT,
MOIS;
```

9.5.4.2. ____ Nombre de commandes et ventes totales par vendeur, par client, par mois, par année

```

SELECT      NO_VENDEUR,
            NO_CLIENT,
            TO_CHAR (DATE_COMMANDE, 'YY') ,
            TO_CHAR (DATE_COMMANDE, 'MONTH') AS MOIS,
            COUNT (NO_COMMANDE) ,
            SUM (BA_MONTANT_COMMANDE)
FROM BA_COMMANDE
WHERE NO_CLIENT=1001
AND NO_VENDEUR=1
GROUP BY NO_VENDEUR,
NO_CLIENT ,
TO_CHAR (DATE_COMMANDE, 'YY') ,
TO_CHAR (DATE_COMMANDE, 'MONTH') ;

```

9.5.4.3. ____ Délai moyen de livraison, par client, par mois, par année

```

SELECT
            NO_CLIENT,
            TO_CHAR (DATE_COMMANDE, 'MM') ,
            TO_CHAR (DATE_COMMANDE, 'YY') ,
            AVG (BA_DELAI) AS DELAI
FROM BA_COMMANDE
WHERE NO_CLIENT=1001
GROUP BY
            NO_CLIENT,
            TO_CHAR (DATE_COMMANDE, 'MM') ,
            TO_CHAR (DATE_COMMANDE, 'YY') ;

```

9.5.4.4. ____ Nombre moyen de lignes de commande par commande pour chaque vendeur, pour chaque mois de toutes les années confondues

```

SELECT
            NO_VENDEUR,
            TO_CHAR (DATE_COMMANDE, 'MONTH') AS MOIS,
            AVG (BA_NB_LIGNES)
FROM BA_COMMANDE
WHERE NO_VENDEUR =1
GROUP BY
            NO_VENDEUR,
            TO_CHAR (DATE_COMMANDE, 'MONTH') ;

```

9.5.4.5. ____ Valeur commandée par marque par jour de la semaine en pour cent de la valeur totale des commandes

```

CREATE VIEW BA_MONTANT_TOTAL AS
SELECT
            SUM (QTE_EXPEDIEE*PRIX) AS MONTANT_TOTAL
FROM DET_COMMANDE;

```

```

SELECT
            PR.MARQUE,
            TO_CHAR (DATE_COMMANDE, 'DAY') as JOUR,

```



```

SUM(BA_MONTANT_LIGNE) AS MONTANT_MA_JOUR,
SUM(BA_MONTANT_LIGNE)/VT.MONTANT_TOTAL*100 AS MONTANT_POURCENT
FROM BA_LI_COMMANDE CO,BA_PRODUIT PR,BA_MONTANT_TOTAL VT
WHERE CO.NO_PRODUIT=PR.NO_PRODUIT
AND PR.MARQUE='A.D.S'
GROUP BY
    PR.MARQUE,
    TO_CHAR (DATE_COMMANDE, 'DAY'),
    VT.MONTANT_TOTAL;

```

9.5.4.6. ____ Rabais en pour-cent par vendeur, par marque, par mois et par année

```

SELECT
    NO_VENDEUR,
    MARQUE,
    TO_CHAR (DATE_COMMANDE, 'YY') AS ANNEE,
    TO_CHAR (DATE_COMMANDE, 'MM') AS MOIS,
    SUM(BA_RABAIS)/SUM(BA_MONTANT_SUGGERE) AS RABAIS_POURCENT
FROM BA_LI_COMMANDE DC, BA_PRODUIT PR
WHERE DC.NO_PRODUIT=PR.NO_PRODUIT
AND PR.MARQUE='CANDLE'
AND NO_VENDEUR=1
GROUP BY
    NO_VENDEUR,
    MARQUE,
    TO_CHAR (DATE_COMMANDE, 'YY'),
    TO_CHAR (DATE_COMMANDE, 'MM');

```

9.5.4.7. ____ Ventes totales par année, par client, par marque.

```

SELECT
    NO_CLIENT,
    MARQUE,
    TO_CHAR (DATE_COMMANDE, 'YY'),
    SUM(BA_MONTANT_LIGNE)
FROM BA_LI_COMMANDE DC, BA_PRODUIT PR
WHERE DC.NO_PRODUIT=PR.NO_PRODUIT
AND NO_CLIENT=1001
AND MARQUE='ALPINE'
GROUP BY
    NO_CLIENT,
    MARQUE,
    TO_CHAR (DATE_COMMANDE, 'YY');

```

9.5.4.8. ____ Évolution du prix moyen de vente effectif par produit par année, par mois

```

SELECT
    MARQUE,
    DC.NO_PRODUIT,
    TO_CHAR (DATE_COMMANDE, 'MONTH') AS MOIS,
    TO_CHAR (DATE_COMMANDE, 'YY'),
    SUM(BA_MONTANT_LIGNE)/SUM(QTE_EXPEDIEE)
FROM BA_LI_COMMANDE DC, BA_PRODUIT PR
WHERE DC.NO_PRODUIT=PR.NO_PRODUIT

```

```
AND MARQUE='A.D.S'
GROUP BY
    MARQUE,
    DC.NO_PRODUIT,
    TO_CHAR (DATE_COMMANDE, 'YY'),
    TO_CHAR (DATE_COMMANDE, 'MONTH')
ORDER BY 1;
```