

MA-004-363-1

République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Saâd Dahlab, Blida



Faculté des sciences

Département Informatique

Mémoire de fin d'études

Pour l'obtention du diplôme de Master en informatique

Option : Ingénierie de logiciel

Thème :

Conception et réalisation d'un système d'information décisionnel avec un système de recommandation

Domaine d'étude :

Entreprise de transformation et distribution de plastique PTD

Réalisé par

Mansouri Khaled

Messedaa Riad Faïçal

P. Rassi et al
BOUMENDI

Encadré par :

Mme. N.Rezoug

M. T.Bachir

M. A.Ramoul

Promotion : 2015/2016

MA-004-363-1

Remerciements

Avant toute chose, Nous remercierons ALLAH le tout puissant, de nous avoir donnée la force et la patience pour mener à terme ce travail

Nous présentons nos sincères remerciements à notre promotrice Mme REZOUG Nachida pour avoir assuré le suivi de ce projet mais aussi pour leur soutien, leurs conseils et leur disponibilité.

Nous tenons également à remercier, M. BACHIR Tadjeddine, Président de LeaderSoft, et M. RAMOUL AMIN Directeur commercial de PTD pour nous avoir acceptés et suivis tout au long de ce projet de fin d'études.

Nous tenons à remercier aussi tous nos enseignants qui ont participé à notre formation tout au long de notre cursus.

Nous tenons à remercier nos parents, nos familles et amis pour leur soutien tout au long de ce mémoire.

Nous remercions également toutes ces personnes sur internet qui ont partagé leurs connaissances et expériences et répondu à nos questions, ils nous ont été d'une grande aide.

Enfin, nous remercions tous ceux qui ont aidé de près ou de loin à l'élaboration de ce mémoire et à la réussite de ce projet.

Merci

Résumé

L'accès à la bonne information au bon moment telle est la principale préoccupation des entreprises qui veulent assurer leur croissance et leur pérennité. Evoluant dans des environnements à la fois complexes et compétitifs, les dirigeants s'orientent de plus en plus vers la mise en place de systèmes d'aide à la décision, aux seins de leurs entreprises, leur fournissant l'information nécessaire au bon pilotage opérationnel et stratégique.

PTD (plastique transformation et distribution) est une entreprise moyenne qui a l'ambition de dominer le marché de production des articles en plastique à l'échelle nationale. Dans ce cadre et afin de répondre aux ambitions des dirigeants en termes d'aide à la décision, la mise en place d'une solution d'un système décisionnel est devenue nécessaire.

Il nous a été confié, dans le cadre de ce projet de fin d'étude, le développement d'un système décisionnel pour améliorer la qualité de la décision ce qui va automatiquement améliorer le rendement de l'entreprise. Ce système a pour objectif de donner à tout moment une vision claire sur la situation actuelle de l'entreprise, et de permettre l'accès aux informations nécessaires pour le pilotage et la prise de décision. Notre travail consiste à la mise en place d'un entrepôt de données accessible de manière simple et facile avec un système de recommandation qui doit faciliter et enrichir l'expérience des décideurs.

ملخص

الحصول على المعلومات الصحيحة في الوقت المناسب هو الشغل الشاغل للشركات التي تريد ضمان نموها واستدامتها. بحيث تعمل هذه الشركات في بيئة معقدة وتنافسية، مما جعل مسيرتها يتجهون أكثر فأكثر نحو تطوير نظم دعم القرار، التي تزويد الشركة بالمعلومات اللازمة لإدارة العمليات والاستراتيجيات المناسبة .

شركة PTD (تحويل بلاستيك و توزيعه) هي شركة متوسطة تطمح للهيمنة على سوق صناعة البلاستيك على مستوى الوطن. في هذا السياق، ومن أجل تلبية احتياجات المسيرين من حيث دعم اتخاذ القرار، استلزم على الشركة إدراج مشروع لإنجاز نظام دعم القرار .

في إطار مشروع التخرج كلفنا بتطوير نظام لدعم اتخاذ القرارات لتنمية وتحسين نوعية القرارات المتخذة. كما يقوم هذا الندام بإعطاء رؤية واضحة للوضع الحالي للشركة في أي لحظة والسماح بالوصول إلى المعلومات اللازمة بهدف التحكم واتخاذ القرار. عملنا ينص على تطوير مستودع بيانات يسمح لمستخدميه الاستفادة من المعلومات بأفضل طريقة بالإضافة إلى نظام توصية وتوجيه وذلك لتسهيل وإثراء تجربة صانعي القرار.

Abstract

Accessing to the right information at the right time is the main concern for companies that want to ensure their growth and sustainability. Operating in complex and competitive environments, managers are moving more and more towards the use of decision support systems in their businesses, to provide them with the information necessary for the proper operational and strategic management.

PTD (plastic transformation and distribution) is a medium enterprise who has the ambition to dominate the nationwide production market of plastic transformation. In this context and in order to satisfy leader's ambitions in terms of decision support, the implementation of a decision support system has become necessary.

In the context of this graduation project, we were in charge of the development of a decision support system to improve the quality of the decision, which will automatically improve the performance of the company. This system aims to give a clear view of the current situation of the company at any time, and enable access to information necessary for monitoring and decision-making. Our work consists in the establishment of an accessible data warehouse in a simple and customizable manner with a recommendation system that should facilitate and enrich the experience of decision makers.

SOMMAIRE

Remerciement.....	2
Résumé.....	3
Sommaire.....	7
Listes des figures.....	12
Listes des tableaux	13
Listes des algorithmes	14
Introduction	15
PARTIE 1 : ETAT DE L'ART.....	19
Chapitre I : L'informatique décisionnelle.....	20
I.1 Introduction.....	20
I.2 Définitions	21
I.3 Architecture d'un système d'information décisionnel.....	21
I.3.1 Sources de données.....	22
I.3.2 Outil d'extraction, transformation et chargement ETL	22
I.3.3 Entrepôt de données	24
I.3.4 Traitement analytique en ligne OLAP.....	28
I.3.5 Outils de visualisation.....	29
I.4 Conclusion	29
Chapitre II : La modélisation multidimensionnelle.....	30
II.1 Introduction.....	30
II.2 Définitions.....	31
II.3 Modèles de présentation des données	32
II.3.1 Modèle conceptuel.....	32
II.3.2 Modélisation.....	33
II.3.3 Modèle logique.....	36
II.4 Manipulation multidimensionnelle des données.....	38

II.4.1	Opérations classiques.....	38
II.4.2	Opérations agissant sur la granularité.....	39
II.4.3	Opérations agissant sur la structure.....	40
II.4.4	MDX	41
II.5.	Conclusion.....	41
	Chapitre III: Systèmes de Recommandation.....	42
III.1	Introduction.....	42
III.2	Définitions.....	43
III.3	Phases de processus de recommandation	43
	III.3.1 Phase de collecte d'information	43
	III.3.2 Phase d'apprentissage.....	44
	III.3.3 Phase de recommandation.....	44
III.4	Techniques de filtrages des recommandations.....	45
	III.4.1 Filtrage personnalisé.....	45
	III.4.2 Filtrage basé sur le contenu.....	45
	III.4.3 Filtrage collaboratif.....	46
	III.4.4.Filtrage hybride.....	47
III.5	Conclusion.....	48
	PARTIE 2 : ETUDE DE L'EXISTANT ET LA COLLECTE DES BESOIN.....	49
	Chapitre IV : Présentation de l'organisme d'accueil.....	50
IV.1	Introduction.....	50
IV.2	LeaderSoft.....	51
IV.3	Présentation de PTD.....	51
	IV.3.1 SARL PTD en bref.....	51
	IV.3.2 Les principaux objectifs de l'entreprise.....	51
	IV.3.3 Organisation de PTD.....	52
IV.4	Contexte de l'étude.....	53

IV.4.1	Service logistique.....	53
IV.4.2	Service de production.....	53
IV.4.3	Service commercial.....	54
IV.5	Conclusion.....	54
Chapitre V : Étude de l'existant Et Étude des besoins.....		55
V.1	Introduction.....	55
V.2	Etat du décisionnel.....	56
	V.2.1 Solution du reporting existante.....	56
	V.2.2 Limites du reporting existant.....	56
V.3	Etude des besoins.....	57
	V.3.1 Définition des besoins.....	57
V.4	Conclusion.....	63
PARTIE 3 : CONCEPTION DE LA SOLUTION.....		64
Chapitre VI : Conception de l'entrepôt de données.....		65
VI.1	Introduction.....	65
VI.2	Activité de vente (commercialisation).....	66
	VI.2.1 Présentation de l'activité.....	66
	VI.2.2 Le fait « ventes ».....	66
VI.3	Activité d'achat.....	68
	VI.3.1 Présentation de l'activité.....	68
	VI.3.2 Le faits « achat ».....	68
VI.4	Activité de production.....	69
	VI.4.1 Présentation de l'activité.....	69
	VI.4.2 Le fait « Production In ».....	70
	VI.4.3 Le fait « Production Out ».....	71
VI.5	Les dimensions.....	72

VI.5.1	Dimension temps « Time ».....	72
VI.5.2	Dimension « Client ».....	73
VI.5.3	Dimension fournisseur « Supplier ».....	74
VI.5.4	Dimension matière première « ArticleRaw »	75
VI.5.5	Dimension produit fini « ArticleFinished ».....	76
VI.6	Dimensions participantes pour chaque fait.....	77
VI.7	Schéma globale de l'entrepôt	78
VI.8	Conclusion.....	79
	Chapitre VII : Conception de la zone d'alimentation.....	80
VII.1	Introduction.....	80
VII.2	Identification des sources de données.....	81
VII.3	Extraction des données.....	81
VII.4	Transformation.....	81
VII.5	Chargement.....	83
VII.6	Conclusion.....	84
	Chapitre VIII : Conception des cubes dimensionnels.....	85
VIII.1	Introduction.....	85
VIII.2	Structure hiérarchique des dimensions.....	86
VIII.3	Listes des cubes.....	87
VIII.4	Schémas des cubes.....	88
	VIII.4.1.Fait « Purchase ».....	88
	VIII.4.2.Fait « Production In ».....	89
	VIII.4.3.Fait « Production Out ».....	89
	VIII.4.4.Fait « Sales ».....	90
	VIII.4.Conclusion.....	90
	Chapitre IX : Conception du système de recommandation.....	91
IX.1	Introduction.....	91
IX.2	Technique de filtrage « hybride ».....	92

IX.2.1	Phase de collecte d'information.....	92
IX.2.2	Phase d'apprentissage.....	95
IX.2.3	Phase de recommandation.....	98
IX.3	Conclusion.....	99
	PARTIE 4 : REALISATION ET DEPLOIEMENT.....	100
	ChapitreX :Réalisation et déploiement de la solution.....	101
X.1	Introduction.....	101
X.2	Environnement de déploiement.....	102
X.2.1	Entrepôt de données.....	102
X.2.2	Restitution.....	104
X.2.3	Microsoft Azure.....	105
X.3	Réalisation du projet.....	106
X.3.1	Construction de l'entrepôt de données.....	106
X.3.2	Zone d'alimentation (ETL).....	107
X.3.3	Construction et chargement des cubes.....	109
X.3.4	Zone de restitution.....	110
X.3.5	système de recommandation.....	114
X.4	Conclusion.....	116

LISTE DES TABLEAUX

Tableau I-1: Avantages et inconvénients de l'approche Top-Down	27
Tableau I-2: Avantages et inconvénients de l'approche Bottom-Up	27
Tableau V-1 : questionnaire d'élaboration des besoins.....	58
Tableau V-2 : tableau descriptif des indicateurs du volet des achats.....	59
Tableau V-3 : tableau descriptif des indicateurs du volet des productions.....	61
Tableau V-4 : tableau descriptif des indicateurs du volet des ventes.....	63
Tableau VI-1:Tableau descriptif du fait « vente ».....	67
Tableau VI-2 :Tableau descriptif du fait « Achat »	69
Tableau VI-3 :Tableau descriptif du fait « Production In »	71
Tableau VI-4 :Tableau descriptif du fait « Production Out ».....	72
Tableau VI-5 :Tableau descriptif de la dimension « Time ».....	73
Tableau VI-6 : Tableau descriptif de la dimension « Client ».....	74
Tableau VI-7 : Tableau descriptif de la dimension « Supplier».....	75
Tableau VI-8 : Tableau descriptif de la dimension « ArticleRaw ».....	76
Tableau VI-9 : Tableau descriptif de la dimension « ArticleFinished ».....	77
Tableau VI-10 : Tableau descriptif des dimensions participantes pour chaque fait.....	77
Tableau VIII-1 : Tableau donnant les niveaux et les hiérarchies de chaque dimension.....	87
Tableau VIII-2 : Liste des cubes.....	88
Tableau IX-1 : description de la structure de stockage.....	93

LISTE DES FIGURE

Figure I-1 : Ensemble des composants d'un système d'information décisionnel	21
Figure II-1 : Exemple d'un cube de données	31
Figure II-2 : Exemple de fait	32
Figure II-3 : Exemple de dimensions	33
Figure II-4 : Exemple d'une modélisation en étoile	34
Figure II-5: Exemple de modélisation en constellation	35

Figure II-6 : Exemple d'une modélisation en flocon	36
Figure II-7 : Moteur R-OLAP : Mondrian	37
Figure II-8 : Structure d'un moteur M-OLAP	37
Figure II-9 : Structure d'un moteur H-OLAP	38
Figure II-10 : Principe de "slice & dice"	39
Figure II-11 : Principe du forage	40
Figure II-11 : Principe de rotation	40
Figure III-1 : Processus de filtrage collaboratif	46
Figure IV-1 : organigramme de PTD.....	52
Figure V-1: les dimensions métier du volet des achats.....	59
Figure V-2: les dimensions métier du volet des productions.....	60
Figure V-3: Les dimensions métier du volet des ventes.....	62
Figure VI-1: Le fait « vente ».....	67
Figure VI-2 : le fait « Achat ».....	69
Figure VI-3 : Le fait « Production In ».....	70
Figure VI-4 : Le fait « Production Out ».....	72
Figure VI-5 : La dimension « Time ».....	73
Figure VI-6 : La dimension « Client ».....	74
Figure VI-7 : La dimension « Supplier ».....	75
Figure VI-8 : La dimension « ArticleRaw ».....	76
Figure VI-9 : La dimension « ArticleFinished ».....	76
Figure VI-10 : schéma en constellation de l'activité « globale ».....	78
Figure VII-1 : Exemple de changement des libellés.....	82
Figure VII-2 : Diagramme d'activité de processus de chargement.....	83
Figure VIII-1 : Cube dimensionnel « Purchase ».....	88
Figure VIII-2 : Cube dimensionnel « Prodcution in ».....	89
Figure VIII-3 : Cube dimensionnel « Prodcution Out ».....	89
Figure VIII-4 : Cube dimensionnel « Sales ».....	90
Figure IX-1: diagramme de class de la structure de stockage des requêtes MDX.....	93

Figure IX-2 : Schéma de fonctionnement de l'algorithme d'apprentissage utilisé.....	96
Figure X-1 : requête de la création de la table « FactSales ».....	106
Figure X-2 : schema de remplissage globale.....	107
FigureX-3 : remplissage de la table de fait « FactSales ».....	108
Figure X-4 : construction des cubes.....	109
Figure X-5 : page login.....	110
Figure X-6 : code source d'authentification.....	111
Figure X-1 : requête de la création de la table « FactSales »	107
Figure X-2 : schema de remplissage globale	108
Figure X-3 : remplissage de la table de fait « FactSales »	109
Figure X-4 : construction des cubes	110
Figure X-5 : page login	111
Figure X-6 : code source d'authentification	112
Figure X-7 : la page d'accueil de service commercial	113
Figure X-8 : interface d'un cube dimensionnel	114
Figure X-9 : Chart de comparaison des chiffres d'affaires des trois années.....	115
Figure X-10 : Code source de l'algorithme de procédure de collecte 1	115
Figure X-11 : Code source de l'algorithme de procédure de collecte 2	116
Figure X-12 : Code source de l'algorithme d'apprentissage « initialisation MDX _N ».....	117

LISTE DES ALGORITHMES

Algorithme IX-1 : Procédure de collecte1 (lancé lors de l'exécution d'une requête).....	94
Algorithme IX-2 : Procédure de collecte 2 (lancé lors de fermeture de la session).....	95
Algorithme IX-3 : algorithme d'apprentissage.....	97

Introduction générale

Contexte

Face à la mondialisation, la concurrence et la crise subie par le système financier mondial, les entreprises ont été incitées à examiner de près l'ensemble de leurs projets et de leurs dépenses. En outre, la prise de décision est devenue encore plus cruciale pour les dirigeants d'entreprises.

A cet effet, les directions financières doivent pouvoir offrir aux entités dirigeantes une vision synthétique et suffisamment détaillée de la situation économique de l'entreprise.

Pour cela, il ne suffit pas d'enrichir son système d'information de données multiples, correspondant aux flux générés par l'ensemble des événements commerciaux, matériels, juridiques et économiques, il faut aussi pouvoir les analyser, les corréler et fournir à l'ensemble des décideurs les moyens de prendre les bonnes décisions à court, moyen et long terme.

L'efficacité de la prise de décision repose sur la mise à disposition du décideur des informations pertinentes au moment opportun, et des outils adaptés à l'analyse. Des exigences auxquels ne répondent pas les systèmes opérationnels, et qui ont donné lieu à l'élaboration de nouveaux systèmes, dits décisionnels.

Les systèmes décisionnels reposent sur un espace de stockage centralisé, appelé entrepôt de données, et proposent des fonctionnalités d'extraction et d'analyse. Ils permettent notamment de collecter des informations provenant de sources différentes afin de les exploiter à travers d'interfaces de représentation.

Plusieurs entreprises dans le domaine de l'informatique sont intéressées par les systèmes décisionnels et les domaines d'analyses des données, car ils sont d'une importance cruciale pour le développement des entreprises, LeaderSoft est l'une des leaders dans ce domaine (conception des logiciels), cette dernière veut satisfaire les demandes de ces clients (dans notre cas, mise en place d'un système décisionnel pour PTD).

Dans un tel contexte LeaderSoft nous confie dans le cadre de notre projet de fin d'étude, de concevoir et de réaliser un système décisionnel pour l'entreprise PTD, et améliorer la qualité de la décision ce qui améliorera le rendement de l'entreprise.

PTD est une entreprise qui veut conquérir le marché algérien dans le domaine de transformation du plastique, une entreprise avec des grandes ambitions qui veut exploiter tous les données et les informations à son service.

Problématique

Selon un sondage réalisé par la société d'entreposage de données « Teradata », près des trois quarts des professionnels de la finance ont du mal à recueillir des données complexes, d'effectuer les analyses nécessaires pour répondre aux questions sur la performance financière de l'entreprise, et de faciliter la prise de décision.

Plusieurs problèmes et besoins ont déclenché la démarche de création d'une solution décisionnelle pour l'entreprise PTD, parmi eux :

- Le besoin d'analyser les ventes, pour la restriction et la connaissance des éléments qui influent sur les ventes (par exemple la température et comprendre le rapport entre la température et les ventes) ce qui permettra à l'entreprise d'augmenter son chiffre d'affaire en manipulant ses éléments à son avantage et en profitant des factures naturelles.
- La gestion de stock, soit en termes de matière première ou bien produit fini, peut devenir un processus difficile, et l'erreur dans cet aspect peut résulter à des fonds bloqués ou d'un stock surchargé. D'où le besoin d'analyse sur la production pour déterminer la réalisation des services de production et la réponse au besoin de marché.
- En plus PTD est une entreprise avec un système d'approvisionnement basé sur l'importation, cette dernière prend de temps (la réception prend presque deux mois après la commande) d'où le besoin d'un système décisionnel se manifeste pour planifier les achats à long terme.

Objectifs

Dans le cadre de notre projet et afin de répondre aux problèmes mentionnés dans la problématique, l'objectif de ce travail est de concevoir et implémenter un système décisionnel qui se propose de :

- Réduire le temps consacré à l'analyse.
- Permettre un accès facile et rapide à l'ensemble des données en les centralisant.
- Permettre d'établir un diagnostic spécifique aux besoins des décideurs en donnant l'accès à des indicateurs pertinents, fiables, facilement interprétables et mesurés dans le temps.
- Mettre à la disposition des utilisateurs, dans des délais très courts, des états de sorties multiples et personnalisés (rapport, graphes, tableau de bord ...).
- Offrir à l'utilisateur la possibilité de naviguer dans les données de manière interactive à l'aide des cubes OLAP.
- Offrir à l'utilisateur un système de recommandation qui lui permet d'accéder au contenu voulue avec un nombre minimal de clique.
- Un accès sur n'importe qu'elle type d'appareille (pc, portable, tablette...).
- Une interface dynamique et fluide avec des charts et des graphes pour simplifier l'analyse.

Contenu du mémoire :

Pour présenter au mieux notre travail, nous avons structuré ce document en quatre parties. Avant d'entamer ces quatre parties nous avons introduit le contexte de notre étude et fixé la problématique ainsi que les objectifs du projet.

Le contenu des quatre parties peut être résumé comme suit :

- **Partie 1 Etat de l'art :** Il s'agit d'une partie théorique où nous présentons les notions relatives à notre projet dans les domaines des systèmes décisionnels et de la modélisation multidimensionnelles ainsi que les systèmes de recommandations.
- **Partie 2 Etude de l'existant et collecte des besoins :** Nous abordons la première étape de la partie pratique de notre projet en présentant l'organisme d'accueil, l'étude de l'existant en termes de décisionnel puis nous exposons notre démarche de collecte des besoins.
- **Partie 3 Conception de la solution :** Dans cette partie, nous présentons les différentes étapes de la conception de la solution qui consiste en un système décisionnel qui s'articule autour d'un entrepôt de données. Nous abordons donc la conception de l'entrepôt, de sa zone d'alimentation et la conception des cubes dimensionnels. Enfin nous entamons la conception de notre système de recommandation des requêtes MDX.
- **Partie 4 Réalisation et déploiement :** C'est la dernière partie de notre projet où nous procédons à la réalisation et au déploiement de la solution conçue. Nous débutons avec la présentation de l'environnement de déploiement et décrivant les outils utilisés puis nous présentons les différentes étapes de la réalisation qui aboutiront à un système fonctionnel.

Nous clôturons notre document avec une conclusion générale où nous synthétisons notre travail.

PARTIE 1 : ETAT DE L'ART

Nous présenterons dans cette partie une synthèse bibliographique sur l'informatique décisionnelle, la modélisation multidimensionnelle, ainsi que les systèmes de recommandation.

Chapitre I

L'informatique décisionnelle

I.1 Introduction

L'une des plus grandes richesses d'une entreprise est bien son information. Cette dernière est noyée sous de nombreuses données, éparpillées, non structurées et hétérogènes, et donc les dirigeants sont face à une problématique de taille : comment analyser ces données ? dans un temps raisonnable. Ces décideurs nécessitent qu'on leur expose les faits importants, base de leurs décisions.

C'est ce à quoi l'**informatique décisionnelle** (ou bien *BI pour Business Intelligence*) est destinée. Elle permet une sélection des informations opérationnelles pertinentes pour l'entreprise, celles-ci sont ensuite normalisées pour alimenter un entrepôt de données (processus ETL). De ce concept est née la notion de modélisation dimensionnelle. Cette dernière est fondamentale pour répondre aux exigences de rapidité et de facilité d'analyse. Elle permet, en outre, de rendre les données d'un entrepôt cohérentes, lisibles, intelligibles et faciles d'accès.

L'informatique décisionnelle doit produire des indicateurs et des rapports à l'attention des analystes. Elle doit également proposer des outils de navigation, d'interrogation et de visualisation de l'entrepôt.

I.2 Définitions

L'informatique décisionnelle est définie comme « outil d'aide à la décision, basé sur une base de données fédérant et homogénéisant les informations des différents services d'une organisation. » [1].

Les données de l'informatique décisionnelle sont affichées d'une manière qui est adaptée à chaque type d'utilisateur, les analystes pourront percer dans les données détaillées, les cadres verront des informations résumées, et les gérants intermédiaires verront les données présentées au niveau de détail dont ils ont besoin pour prendre de bonnes décisions [2].

I.3 Architecture d'un système d'information décisionnel

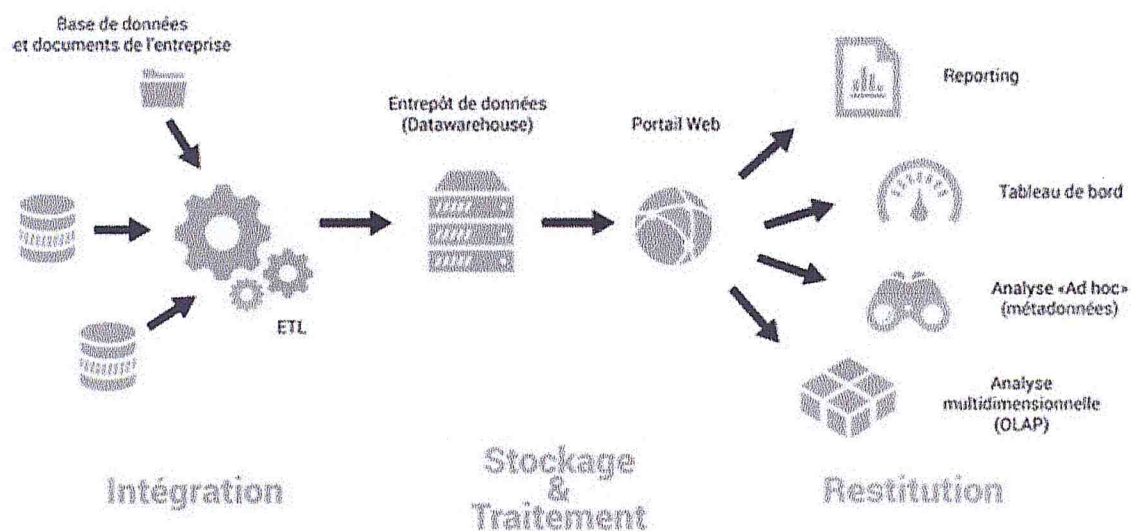


Figure I-1 : Ensemble des composants d'un système d'information décisionnel [3]

Cette section a pour bût de parcourir les différents éléments nécessaires à la mise en place d'une solution d'aide à la décision, depuis l'extraction des données jusqu'à leurs restitution sous formes agrégées, synthétisées et normalisées.

I.3.1 Sources de données

Pour alimenter les entrepôts, les données doivent être identifiées puis extraites de leurs emplacements originels. Les sources de données figurants dans un entrepôt de données peut être groupées dans quatre catégories [4] :

- **Données de productions** : il s'agit de données internes à l'entreprise, mais diffuses, car stockées dans les bases de données de production des différents services. Elles sont caractérisées par leur disparité (plusieurs services, technologies, environnements, ...).
- **Données internes** : ce sont des données qui concernent l'entreprise (brouillons, profils des clients, ...) mais elles sont détenues par des employés ou des départements dans leurs fichiers et bases de données personnels, leur extraction et intégration est de ce fait complexe.
- **Données archivées** : il s'agit des données anciennes stockées périodiquement par le système opérationnel dans des fichiers ou bases de données, ces données sont d'une importance immense dans n'importe quel système décisionnel afin d'aider à faire des analyses sur les différentes périodes passées sur les activités de l'entreprise et extraire des prévisions critiques.
- **Données externes** : données provenant de sources externes, il peut s'agir de statistiques relatives à leur industrie produits par des agences externes, données publiques sur les concurrents de marché, ... nécessaires pour se situer par rapport à la concurrence et repérer les tendances du marché.

I.3.2 Outil d'extraction, transformation et chargement ETL

Aussi connus sous le terme Extract Transform Load (ETL), ces outils sont conçus afin de faciliter l'intégration de données hétérogènes, leur normalisation puis ils les rendent cohérentes entre elles [5], pour qu'elles puissent être utilisées conjointement. Les données sont présentées dans un format permettant une exploitation immédiate sans recalculs par les décideurs et les analystes.

Un outil ETL est beaucoup plus que la plomberie pour obtenir des données depuis les sources vers l'entrepôt, mais il ajoute une valeur significative aux données.

Plus précisément, un outil ETL [6] :

- Supprime les erreurs et corrige les données manquantes.
- Fournit des mesures documentées de confiance dans les données.
- Capture le flux de données transactionnelles pour la garde.
- Ajuste les données provenant de sources multiples pour être utilisées ensemble.
- Structure les données pour être utilisables par les outils de l'utilisateur final.

Les outils ETL passent par trois grandes étapes :

I.3.2.1 Extraction :

Lors de l'extraction, les données sont identifiées et extraites de différentes sources, y compris les systèmes et les applications des bases de données. Très souvent, il est impossible d'identifier le sous-ensemble spécifique d'intérêt, donc plus de données que nécessaire doivent être extraites, de sorte que l'identification des données pertinentes se fera à un moment ultérieur. En fonction des capacités du système de la source, certaines transformations peuvent avoir lieu au cours de ce processus d'extraction. La taille des données extraites varie de quelques centaines de kilo-octets jusqu'à giga-octets, en fonction du système source et la situation de l'entreprise. La même chose est vraie pour le delta de temps entre deux extractions (logiquement) identiques : l'intervalle de temps peut varier entre jours / heures et minutes jusqu'à temps quasi réel.

Le processus d'extraction est constitué de deux phases, l'extraction initiale et l'extraction des données modifiées [7]. Dans l'extraction initiale les données sont obtenues pour la première fois depuis les différentes sources opérationnelles. Ce processus se fait qu'une seule fois après la construction de l'entrepôt de données. Par contre, dans l'extraction des données modifiées (en anglais, CDC pour Changed Data Capture), les outils ETL rafraîchissent l'Entrepôt avec les données modifiées et ajoutées dans les systèmes sources depuis la dernière extraction. Ce processus est périodique selon le cycle de rafraîchissement et les besoins des entreprises.

Après que les données sont extraites, elles doivent être physiquement transportées vers le système cible ou à un système intermédiaire pour un traitement ultérieur. Comme déjà dit, certaines transformations peuvent être effectuées au cours de ce processus de transportation.

I.3.2.2 Transformation :

L'étape de transformation implique l'application d'une série de règles et de fonctions sur les données extraites. Elle comprend la validation des enregistrements et leur rejet si elles ne sont pas acceptables. Cette seconde phase du processus assure la fiabilité des données et leur qualité en exécutant certaines tâches :

- Consolidation des données.
- Correction des données et élimination de toute ambiguïté.
- Élimination des données redondantes.
- Compléter et renseigner les valeurs manquantes.

I.3.2.3 Chargement :

C'est la dernière étape du processus ETL. Les données extraites et transformées vont être écrites dans les structures dimensionnelles réellement accessibles par les utilisateurs finaux. L'étape de chargement comprend le chargement des tables de dimension ainsi que les tables de faits.

I.3.3 Entrepôt de données

Afin de produire une analyse pertinente des données, les systèmes décisionnels s'appuient sur des données historiques qui ont été extraites des divers applicatifs de l'entreprise dans le processus ETL. Cela requiert la gestion d'un volume de données très important, et donc l'implémentation d'entrepôt de données.

I.3.3.1 Définitions :

Noirault a défini l'entrepôt de données par son but d'utilisation : « une base de données architecturée pour des requêtes et des analyses, plutôt que pour le traitement transactionnel des données » [8].

I.3.3.2 Data warehouse et data mart :

L'entrepôt de données peut prendre la forme d'un *data warehouse* ou d'un *data mart* :

- **Data warehouse** : Le data warehouse ou l'entrepôt globalise toutes les données de toutes les activités d'une entreprise.
- **Data mart** : Le data mart ou le magasin de données est une version plus réduite du data warehouse. Le data mart est orienté métier et donc stock les données d'une seule activité (par exemple, commercialisation)

I.3.3.3 Caractéristiques :

Bill Inmon a aussi défini l'entrepôt de données par ses caractéristiques : « collection de données orientées sujet, intégrées, non volatiles et variantes dans le temps, organisées pour le support d'un processus d'aide à la décision. » [9] :

- **Données orientées sujet** : Les données sont organisées par sujet ou thème, contrairement aux systèmes opérationnels où les données sont organisées par processus fonctionnel.
- **Données intégrées** : De tous les aspects d'un entrepôt de données, l'intégration est la plus importante. Les données provenant des différents systèmes opérationnels doivent être unifiées et normalisées avant d'être chargées dans l'entrepôt.
- **Données non volatiles** : Dans les systèmes opérationnels, les données insérées et accédées un enregistrement à la fois et peuvent être modifiées ou supprimées complètement. Dans l'entrepôt de données par contre, les données sont chargées sous format du *Snapshot* statique. Lorsque les modifications se produisent, un nouvel enregistrement est écrit. Ce faisant une histoire de données est conservée dans l'entrepôt.
- **Données variantes dans le temps** : La variation temporelle implique que chaque donnée dans l'entrepôt est précise dans un moment dans le temps. D'où, tous les enregistrements ont une forme de marquage du temps.

I.3.3.4 La granularité des données :

Dans un système opérationnel, les données sont généralement maintenues au plus bas niveau de détail, et à chaque fois que vous avez besoin des données de synthèse, vous devez ajouter les transactions individuelles. Par contre, lorsqu'un utilisateur interroge l'entrepôt de données pour l'analyse, fréquemment il commence à un niveau élevé et se déplace vers le bas à des niveaux inférieurs de détail. Par conséquent, c'est plus efficace de conserver les données dans l'entrepôt sous formes résumées à différents niveaux de granularité [4].

I.3.3.5 Métadonnées :

Le terme métadonnées désigne les données qui décrivent d'autres données [10]. Et pour gérer physiquement et sémantiquement l'ensemble des données, l'entrepôt doit disposer de métadonnées.

Les métadonnées dans un entrepôt de données se répartissent en trois grandes catégories [4] :

- **Métadonnées opérationnelles** : contiennent toutes les informations sur les sources de données provenant du système opérationnel.
- **Métadonnées d'extraction et de transformation** : contiennent toutes les informations sur l'extraction, à savoir, les fréquences d'extraction, les méthodes d'extraction, et les règles d'extraction. Ainsi que les informations sur les transformations opérées sur les données.
- **Métadonnées de l'utilisateur final** : jouent le rôle de la carte de navigation de l'entrepôt de données. Elles permettent aux utilisateurs finaux de trouver les informations facilement dans l'entrepôt de données.

I.2.3.6 Approches de mise en place d'un entrepôt de données :

Ils existent plusieurs approches pour la mise en place d'un entrepôt de données. Les deux approches les plus répandues sont : l'approche "Top-Down" (de haut en bas) de **Inmon**, l'approche "Bottom-up" (de bas en haut) de **Kimball** [4].

- **L'approche descendante Top-Down (Inmon)**

Cette approche nous incite à construire en premier lieu l'entrepôt de données, en prenant en considération les besoins de l'entreprise dans sa globalité. Dès que l'entrepôt est mis en place,

il est alors possible de créer des magasins de données contenant individuellement des parties de l'entrepôt pour répondre aux besoins de services particuliers.

Avantages	Inconvénients
<ul style="list-style-type: none"> - Vision claire et globale des données de l'entreprise - Architecture intégrée, pas une union de magasins de données disparates - Stockage simple et centralisé des données - Règles et un contrôle centralisé - Maintenance facile 	<ul style="list-style-type: none"> - Longue et coûteuse à développer - Risque d'échec important - Résultats non immédiats

Tableau I-1 : Avantages et inconvénients de l'approche Top-Down [4]

- L'approche ascendante Bottom-Up (Kimball)

Par opposition à l'approche descendante, l'approche ascendante commence par la mise en place des magasins de données, ceux-ci sont rassemblés pour former l'entrepôt de données pour l'ensemble de l'entreprise.

Avantages	Inconvénients
<ul style="list-style-type: none"> - Implémentation rapide, facile et peu coûteuse. - Résultats rapides - Risque d'échec minime - Approche incrémentale (magasins importants d'abord) 	<ul style="list-style-type: none"> - Chaque magasin de donnée indépendant a sa propre vision limitée des données, - Incohérences et redondance des données entre les magasins de données, - Fragmentation des données, - Performance non optimale des analyses impliquant plusieurs magasins de données - Volume de travail d'intégration important pour obtenir un entrepôt de données.

Tableau I-2 : Avantages et inconvénients de l'approche Bottom-Up [4]

I.3.4 Traitement analytique en ligne OLAP

En 1993, Edgar Frank Codd a introduit le terme On-Line Analytical Processing (OLAP) qui à la différence de l'OLTP, introduit une technologie optimisée pour les requêtes humaines plutôt que pour les transactions [11].

Il a aussi introduit 12 « règles de base » permettant de qualifier l'OLAP :

1. **Vue conceptuelle multidimensionnelle** : permet d'avoir une vision multidimensionnelle des données.
2. **Transparence** : l'utilisateur doit pouvoir accéder aux données, sans se préoccuper des sources de données.
3. **Accessibilité** : les données doivent toutes être accessibles, sans ambiguïté.
4. **Performance consistante des rapports** : les performances ne doivent pas être diminuées lors de l'augmentation du nombre de dimension ou de la taille de la base de données.
5. **Architecture Client/Serveur** : il est essentiel que le produit soit Client-Serveur mais aussi que les composants serveurs d'un produit OLAP intègrent facilement ses différents clients.
6. **Dimensionnalité générique** : toutes les dimensions doivent être équivalentes par rapport à leur structure et leurs capacités opérationnelles.
7. **Ajustement automatique du niveau physique** : le système OLAP ajuste automatiquement son schéma physique pour s'adapter au type du modèle et au volume des données.
8. **Support multiutilisateur** : l'outil doit fournir des accès concurrents, l'intégrité et la sécurité.
9. **Opérations cross-dimensionnel libre** : les calculs doivent être possibles à travers toutes les dimensions.
10. **Manipulation intuitive des données** : la manipulation des données se fait directement à travers les cellules dans le model analytique, sans recourir aux menus ou aux actions multiples.

11. **Rapports flexibles** : lors de la création de rapports, les dimensions peuvent être présentées de n'importe quelle manière.

12. **Dimensions et niveaux d'agrégation illimités.**

Entre entrepôt et OLAP, il n'y a qu'un pas. En effet, l'entrepôt est le lieu de stockage physique des données, tandis que l'OLAP est l'outil permettant leur analyse multidimensionnelle. Afin de rendre l'analyse la moins contraignante et la plus souple possible, l'OLAP propose des opérateurs. Il s'agit de mécanismes servant à naviguer dans les hiérarchies et les dimensions. Nous allons entrer dans les détails de ces opérateurs et mécanismes dans le deuxième chapitre.

I.3.5 Outils de visualisation

La dernière étape consiste à utiliser les données contenues dans l'entrepôt de données. Les outils de restitution doivent offrir aux analystes la possibilité de visualiser et manipuler les données contenues dans les entrepôts et les magasins de données dans un temps acceptable (de l'ordre de quelques secondes), et sans connaissance particulière dans le domaine de l'informatique. Il existe plusieurs types d'outils de visualisation, parmi eux :

- **Les outils de reporting** : permettent de produire des tableaux de bord, des graphes, et des rapports des activités de l'entreprise en fonctions des dimensions de l'entrepôt (temps, produit, client, ...) afin d'évaluer la performance de l'entreprise [12].
- **Les outils de datamining** : permettent d'extraire de la connaissance à partir des volumes de données dans l'entrepôt [13].

I.4 Conclusion

Nous avons introduit dans ce chapitre la notion de l'informatique décisionnelle. Puis nous avons décrit l'architecture d'un système d'information décisionnel, qui se compose d'un entrepôt de données (*data warehouse*) alimenté depuis les sources de données opérationnelles en utilisant des outils d'extraction transformation et chargement (ETL), cet entrepôt va contenir des données agrégées et résumées qui vont être exploitées à l'aide des outils de restitution. Dans les systèmes décisionnels, les données de *data warehouse* sont organisées d'une manière multidimensionnelle, d'où, le chapitre suivant est consacré à l'étude de la modélisation multidimensionnelle.

Chapitre II

La modélisation multidimensionnelle

II.1 Introduction

Contrairement aux bases de données relationnelles, l'intérêt des bases de données décisionnelles ne se situe pas au niveau de l'individu (enregistrement) mais plutôt au niveau de l'identification des tendances dans un ensemble ou un groupe.

Les données à analyser doivent refléter la vision des analystes, c'est-à-dire apparaître sous une forme facilitant les prises de décision. Cette vision correspond à une structuration des données selon plusieurs axes d'analyse représentant des notions diverses telles que le temps, la localisation géographique, etc. On parle d'analyse multidimensionnelle.

II.2 Définitions

La modélisation multidimensionnelle, aussi appelée modélisation OLAP [11], est une technique qui vise à organiser les données de telle sorte que les applications OLAP soient performantes et efficaces [14]. Elle permet aux décideurs de faire des analyses sur les mesures commerciales de l'entreprise de différentes façons [4].

❖ La métaphore du cube

La vision multidimensionnelle des données peut être vue comme un *cube de données*. Chaque donnée représente une cellule du cube et les arêtes représentent les axes d'analyse des données. Sur chaque arête, une graduation est choisie afin d'observer les données à un niveau adéquat de détail.

Exemple : la figure II-1 représente un exemple de cube de données qui permet l'analyse des publications scientifiques.

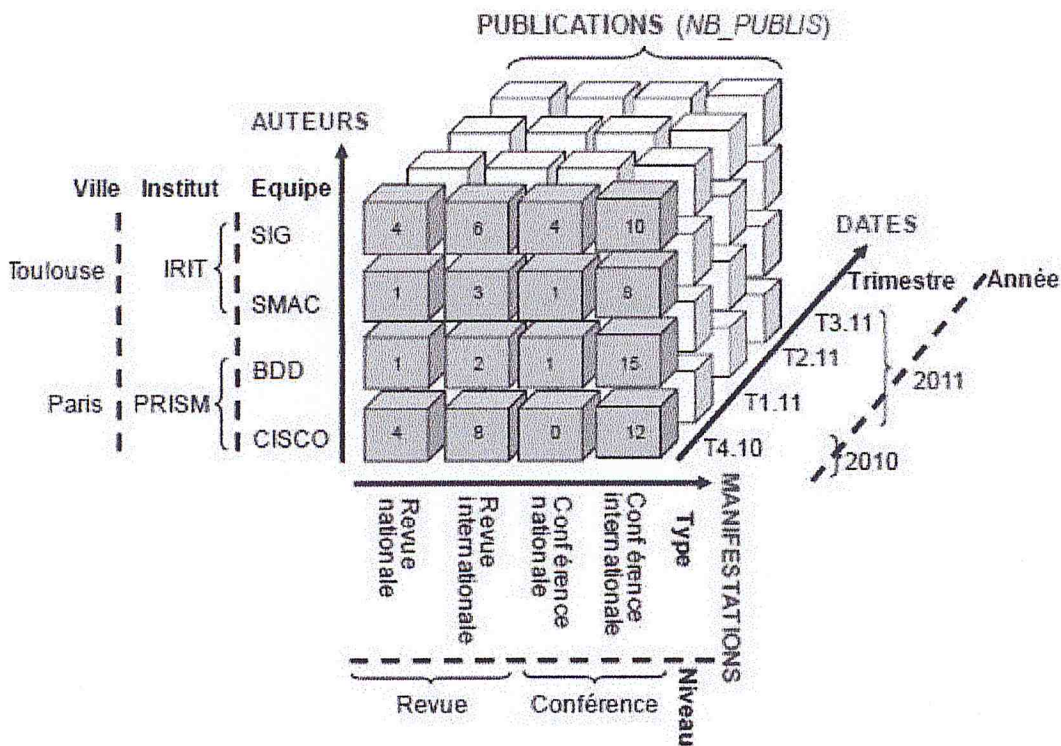


Figure II-1 : Exemple d'un cube de données [15]

II.3 Modèles de présentation des données

La modélisation des données au sein d'un système OLAP peut être représentée conformément aux niveaux d'abstraction classiques définie dans les méthodes de conception des bases de données [15] :

- Modèle conceptuel : Description de la base multidimensionnelle indépendamment des choix d'implantation [16].

Concepts :

- ◆ Dimensions et hiérarchies
 - ◆ Faits et mesures
- Modèle logique : Description de la base multidimensionnelle suivant la technologie utilisée :
 - ◆ ROLAP (Relational-OLAP)
 - ◆ MOLAP (Multidimensional-OLAP)
 - ◆ HOLAP (Hybrid-OLAP)

II.3.1 Modèle conceptuel

La modélisation multidimensionnelle a donné naissance à différents concepts, qui reposent sur la métaphore des cubes [17].

II.3.1.1. Concept de fait :

- Sujet analysé.
- Un ensemble d'attributs appelés mesure.
- Un fait représente la valeur d'une mesure, calculée ou mesurée, selon un membre de chacune.
- Des dimensions [16].

Exemple : soit un fait de vente (figure II-2) constitué des mesures suivantes : quantité des produits vendus et montant des ventes. Le fait est représenté par un rectangle englobant les différentes mesures qu'il contient. En outre le symbole d'un cube estampille le fait.

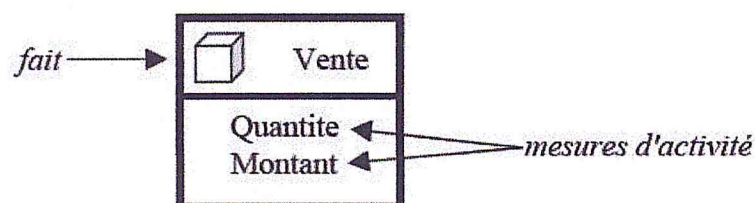


Figure II-2 : Exemple de fait [18]

II.3.1.2 Concept de dimension :

Elément de données sur lequel se portent les analyses, en fonction des différentes dimensions. Ces valeurs sont les résultats des opérations d'agrégations sur les données [16].

Exemple : Le fait de l'exemple précédent peut être analysé suivant trois dimensions qui représentent les perspectives d'analyse : le temps, la catégorie et la situation géographique.

Une dimension est représentée par un rectangle englobant les différents paramètres qu'elle contient. En outre un symbole représentant trois axes estampille les dimensions (Figure II-3).

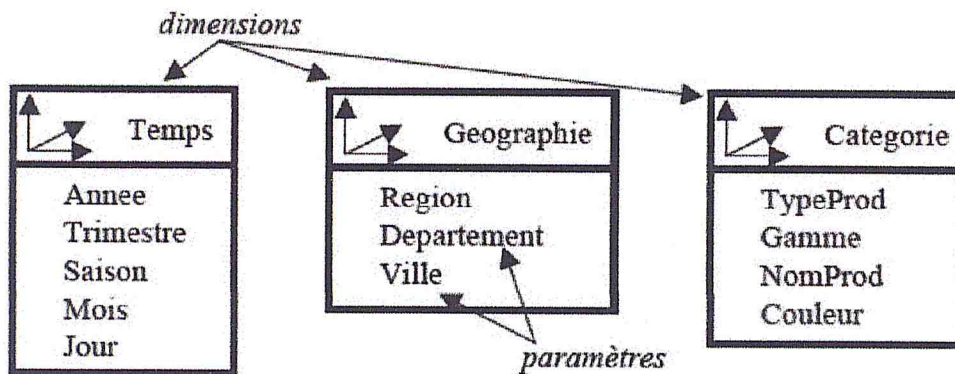


Figure II-3 : Exemple de dimensions [18]

Chaque dimension est munie d'une ou plusieurs hiérarchies qui permettent d'organiser les paramètres conformément à leur niveau de détail.

Exemple : la dimension Géographie est organisée suivant la hiérarchie suivante :

Ville → Département → Région.

Ainsi chaque ville appartient à un département qui est situé dans une région.

L'association de ces concepts compose un modèle.

II.3.2 Modélisation

Au niveau conceptuel, il existe 2 modèles :

- ◆ En étoile (star schema).
- ◆ En constellation (fact constellation schema).

II.3.2.1 Modèle en étoile (Star schema) :

Constitué d'une table de fait centrale et des tables de dimensions, ces derniers ne sont pas reliés.

Avantage :

- ◆ Facilité la navigation.
- ◆ Nombre de jointures limité.

Inconvénients :

- ◆ Redondance dans les tables de dimensions.
- ◆ Toutes les dimensions ne concernent pas les mesures.

Exemple : la figure II-4 correspond à un schéma en étoile modélisant une analyse d'achat, selon cinq dimensions : le temps, le produit, le client, la région et le magasin.

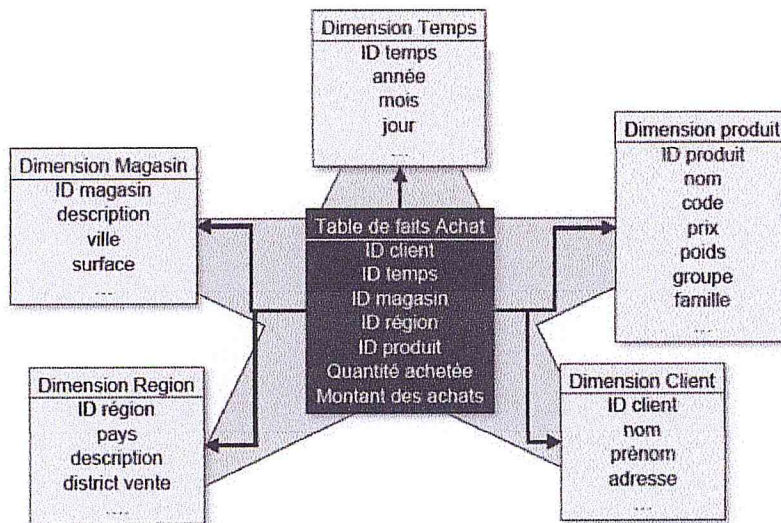


Figure II-4 : Exemple d'une modélisation en étoile [18]

II.3.2.2 Modèle en constellation :

Une série d'étoiles :

- ◆ Fusion de plusieurs modèles en étoile qui utilisent des dimensions communes.
- ◆ Plusieurs tables de faits et tables de dimensions, éventuellement communes.

Exemple : la figure II-5 illustre une modélisation en constellation constituée de deux schémas en étoile : l'un correspond aux ventes des pharmacies et l'autre aux prescriptions des médecins.

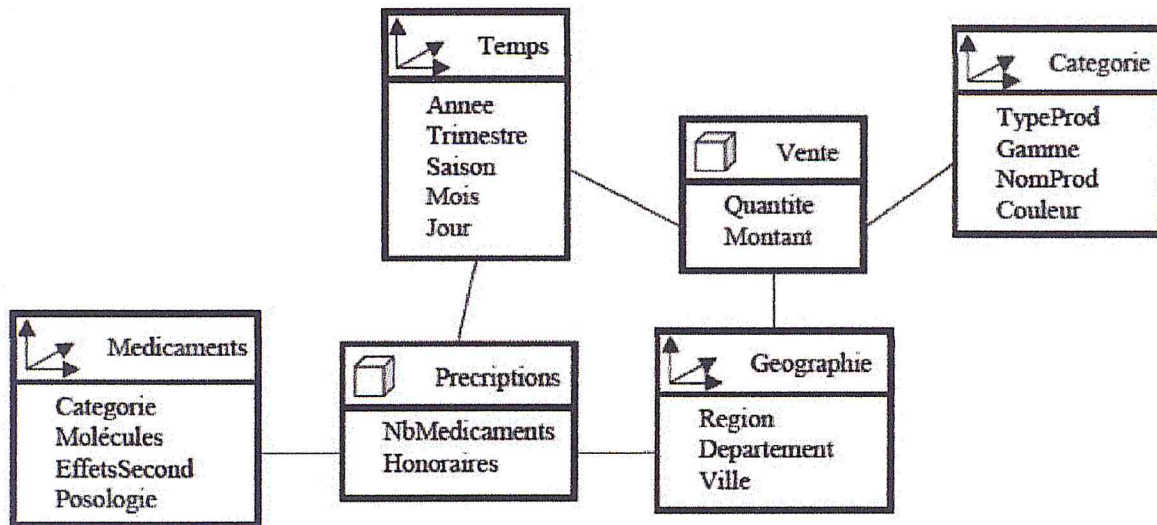


Figure II-5 : Exemple de modélisation en constellation [18]

Au niveau logique, il existe 1 seul modèle :

II.3.2.3 Modèle en flocon (snowflake) :

Modèle en étoile avec la normalisation des dimensions :

- ◆ Une table de fait et des dimensions en sous-hiérarchies ;
- ◆ Un seul niveau hiérarchique par table de dimension ;
- ◆ La table de dimension de niveau hiérarchique le plus bas est reliée à la table de fait (elle a la granularité la plus fine) ;

Avantage : ◆ Normalisation des dimensions ;

- ◆ Economie d'espace disque (réduction du volume) ;

Inconvénients : ◆ Modèle plus complexe (nombreuses jointures) ;

- ◆ Requêtes moins performantes ;
- ◆ Navigation difficile.

Exemple : la figure II-6 correspond à un modèle en flocon ; le fait est conservé, et les dimensions sont éclatées en formant ainsi une sorte de flocon.

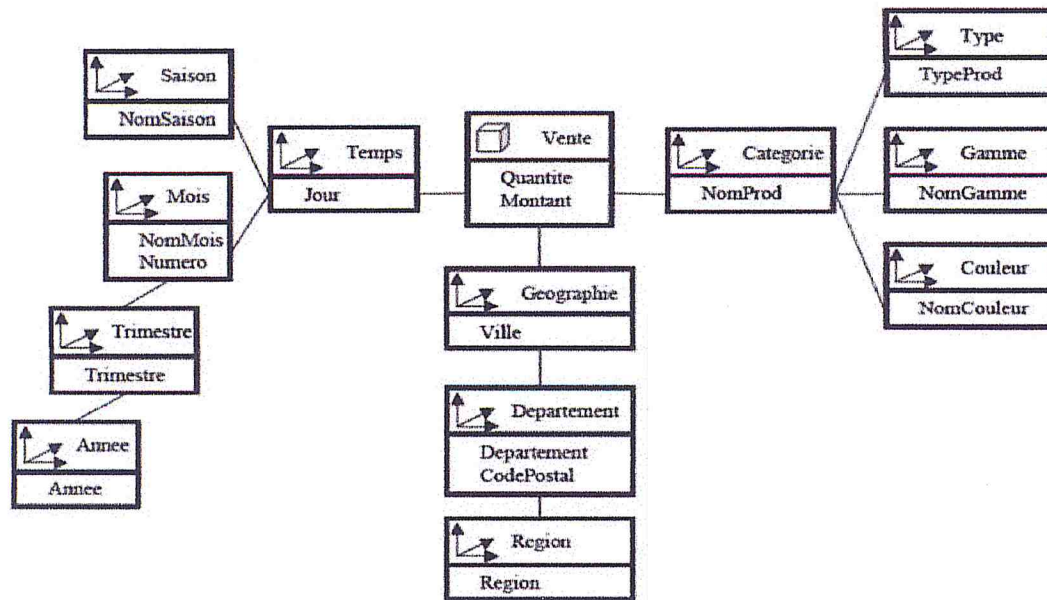


Figure II-6 : Exemple d'une modélisation en flocon [18]

II.3.3 Modèle logique

Description de la base multidimensionnelle suivant la technologie utilisée :

- ◆ ROLAP (Relational-OLAP)
- ◆ MOLAP (Multidimensional-OLAP)
- ◆ HOLAP (Hybrid-OLAP)

II.3.3.1 R-OLAP :

Les données sont stockées dans une BD relationnelle. Un moteur OLAP permet de simuler le comportement d'un SGBD multidimensionnel

- Avantage :
- ◆ Facile à mettre en place ;
 - ◆ Peu coûteux ;
 - ◆ Evolution facile ;
 - ◆ Stockage de gros volumes ;

Inconvénients : ◆ Moins performant lors des phases de calculs.

Exemple de moteur ROLAP : Mondrian (Figure II-7)

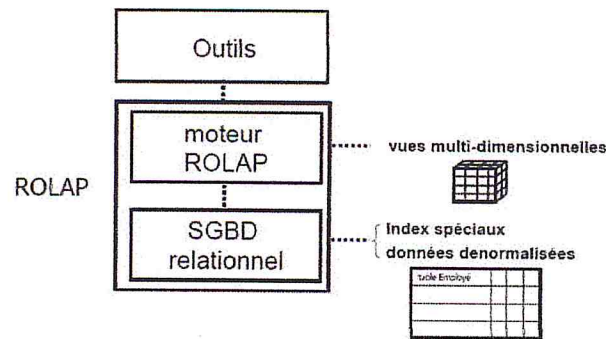


Figure II-7 : Moteur R-OLAP : Mondrian [19]

II.3.3.2 M-OLAP :

Les données sont stockées comme des matrices à plusieurs dimensions :

Cube [1:m, 1:n, 1:p] (mesure)

Avantages : ♦ Rapidité ;

- ♦ Accès direct aux données dans le cube.

Inconvénients : ♦ Difficile à mettre en place ;

- ♦ Formats souvent propriétaires ;
- ♦ Ne supporte pas de gros volumes de données ;

Exemple de moteurs M-OLAP (Figure II-8) :

- ♦ Microsoft Analysis Services
- ♦ Hyperion

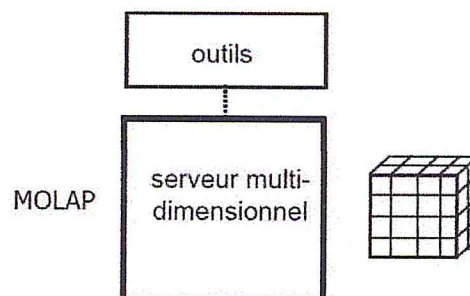


Figure II-8 : Structure d'un moteur M-OLAP [19]

II.3.3.3 H-OLAP :

Solution hybride entre R-OLAP et M-OLAP.

Données de base stockées dans un SGBD relationnel (tables de faits et de dimensions) + données agrégées (Figure II-9).

Avantages / inconvénients : ♦ Au niveau des coûts et des performances (les requêtes vont chercher les données dans les tables et le cube).

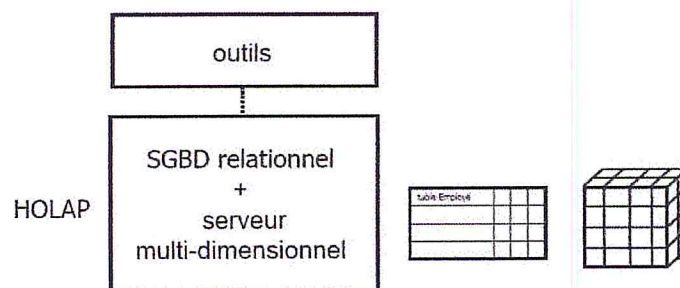


Figure II-9 : Structure d'un moteur H-OLAP [19]

II.4 Manipulation multidimensionnelle des données

Il existe de nombreuses propositions concernant la définition d'opérations de manipulation OLAP dans laquelle les structures manipulées sont les cubes de données et les tranches du cube de données. Cependant, il n'existe pas de consensus sur la définition d'un ensemble minimum d'opérateurs assurant l'intégralité des opérations de manipulation OLAP, mais la plupart des propositions offrent un support partiel des différentes catégories d'opérations suivantes :

II.4.1 Opérations classiques

- **Opérations de sélection :** Ces opérations permettent à un utilisateur de restreindre l'ensemble des données analysées [20].
 - **La spécification d'une « tranche de cube » (slice) :** consiste à exprimer une restriction sur une des données de l'un des axes d'analyse. Cela correspond à une projection selon une dimension du cube (figure II-10).
 - **La spécification d'un « sous-cube » (dice) :** consiste à exprimer une restriction sur les données d'un indicateur d'analyse (figure II-10).

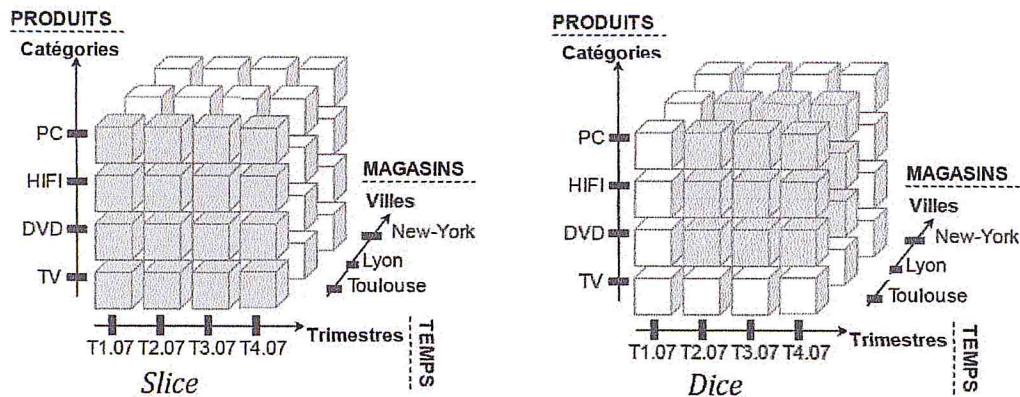


Figure II-10 : Principe de "slice & dice" [15]

Des auteurs ont aussi proposé des opérations additionnelles :

- **Opérateurs binaires.** Certains auteurs proposent aussi l'emploi des opérations binaires ensemblistes : union, différence et intersection. Certains travaux ont aussi proposé la notion de jointure (join) inspirée de la jointure relationnelle, mais d'un intérêt limité dans un environnement multidimensionnel. Il est intéressant de noter que les opérateurs binaires sont des opérateurs nécessitant de très fortes contraintes.

Exemple : l'union entre deux structures multidimensionnelles nécessite une compatibilité presque complète des deux structures.

II.4.2 Opérations agissant sur la granularité

Opérations de forage : Ces opérations permettent la navigation au moyen de la structure hiérarchique des axes d'analyses, afin de permettre l'analyse d'un indicateur avec plus ou moins de précision (Figure II-11) [21].

- **Le forage vers le haut (roll-up)** consiste à analyser les données en fonction d'un niveau de granularité supérieur conformément à la hiérarchie définie sur la dimension.
- **Le forage vers le bas (drill-down)** à l'inverse du roll-up, il permet d'analyser les données avec un niveau de granularité inférieur, donc sous une forme plus détaillée.

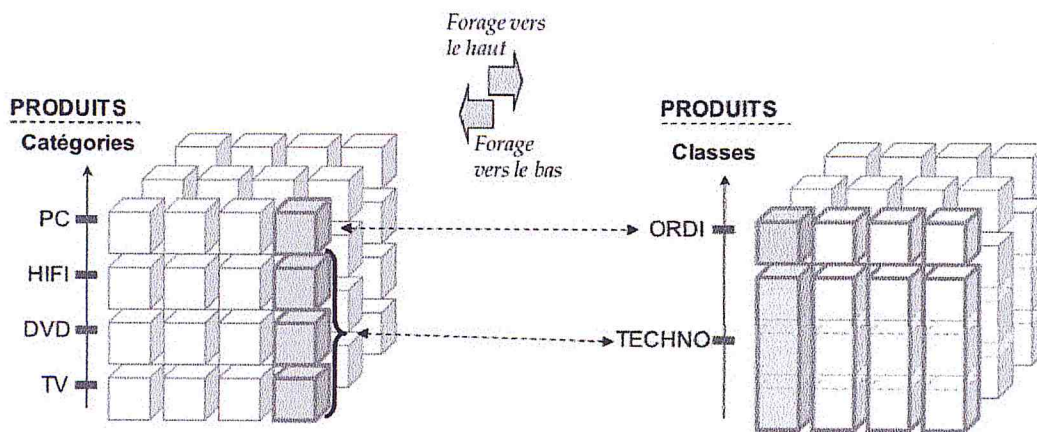


Figure II-11 : Principe du forage [15]

II.4.3 Opérations agissant sur la structure :

- **Opérations de rotation :** Ces opérations permettent la réorientation d'une analyse (Figure II-11). Elles permettent de changer l'un des axes d'analyse en cours d'utilisation (rotation de dimension), de changer le sujet de l'analyse (rotation de fait ou drill-across) et de changer de perspective d'analyse (rotation de hiérarchie) [22].

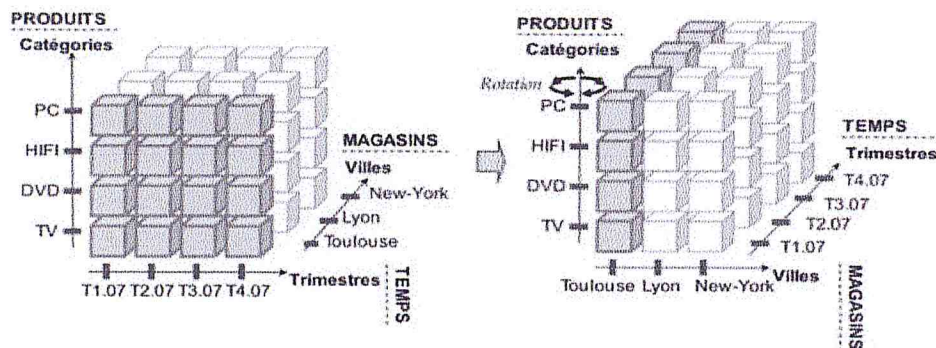


Figure II-11 : Principe de rotation [15]

- **La permutation (switch)** Change la position des membres d'une dimension.
- **La division (split)** consiste à présenter chaque tranche du cube et de passer de sa présentation tridimensionnelle à sa présentation sous la forme d'un ensemble de tables. Plus généralement, cette opération permet de réduire le nombre de dimensions.
- **L'emboîtement (nest)** Imbrique des membres issus de dimensions différentes.
- **L'enfoncement (push)** Combine les membres d'une dimension aux mesures (les membres deviennent le contenu des cellules)

- **L'opération inverse de retrait (pull)** permet de transformer une mesure en paramètre en changeant le statut de certaines mesures du cube pour constituer une nouvelle dimension.
- **La factualisation (fold)** consiste à transformer une dimension en mesure(s); cette opération permet de transformer en mesure l'ensemble des paramètres d'une dimension.
- **La paramétrisation (unfold)** permet de transformer une mesure en un paramètre dans une nouvelle dimension.
- **L'opération Cube** permet de calculer des sous-totaux et un total final dans le cube.

II.4.4 MDX

Le MDX (Multidimensional Expressions) est un langage de requête pour les bases de données OLAP, analogue au rôle de SQL pour les bases de données relationnelles. C'est aussi un langage de calcul avec une syntaxe similaire à celle des tableurs.

Le langage des expressions multidimensionnelles possède une syntaxe appropriée à l'interrogation et manipulation des données multidimensionnelles mémorisées dans un cube OLAP. Bien qu'il soit possible de traduire certaines expressions dans le langage SQL traditionnel, cela nécessite une syntaxe SQL souvent maladroite même pour des expressions MDX très simples. MDX a été adopté par une large majorité de fournisseur de la technologie OLAP et est devenu un standard de facto pour les systèmes OLAP [2].

II.5 Conclusion

Dans ce chapitre nous avons introduit la modélisation multidimensionnelle des données qui repose sur les concepts de fait et de dimension. L'association de fait et de dimension compose un schéma qui peut être soit en étoile, en flocon ou en constellation. L'implémentation de ces schémas se fait principalement selon des modèles R-OLAP et M-OLAP, qui diffèrent essentiellement dans la manière de stockage. Plusieurs opérations permettent la navigation et la manipulation multidimensionnelle des données stockées. Le modèle multidimensionnel est la structure de données la plus utilisée et la plus appropriée aux requêtes et analyses des utilisateurs d'entrepôts de données.

Chapitre III

Systemes de Recommandation

III.1 Introduction

On peut considérer une session d'analyse OLAP comme une session interactive durant laquelle un utilisateur lance des requêtes pour naviguer dans un cube. Choisir quelle partie du cube va être naviguée par la suite, et, de ce fait, concevoir la prochaine requête, est très souvent une tâche difficile. D'où, il est très utile d'avoir un système de recommandation qui utilise ce que tous les utilisateurs du système OLAP ont fait pendant leurs précédentes explorations du cube afin de recommander des requêtes MDX à l'utilisateur.

III.2 Définitions

Un système de recommandation est un système de filtrage qui cherche à présenter les éléments d'information susceptibles d'intéresser l'utilisateur [23]. Il joue le rôle d'un guide pour l'utilisateur lors de son exploration d'une grande quantité d'informations en lui cherchant, les informations qui paraissent pertinentes [24]. La conception d'un tels système dépend du domaine et des caractéristiques particulières des données disponibles [25].

III.3 Phases de processus de recommandation

Un processus de recommandation doit procéder par trois phases [26] :

III.3.1 Phase de collecte d'information

Dans cette phase on va collecter les informations pertinentes des utilisateurs pour générer un profil d'utilisateur ou un modèle pour les tâches de prédiction incluant les attributs de cet utilisateur. Le système a besoin de savoir autant que possible de l'utilisateur afin de fournir des recommandations raisonnables dès le début.

Les systèmes de recommandation reposent sur différents types d'intrants tels que les évaluations explicites de haute qualité, qui comprend l'entrée explicite par les utilisateurs au sujet de leur intérêt pour un objet, ou une inférence implicite en déduisant les préférences des utilisateurs indirectement en observant leurs comportements [27]. L'évaluation hybride peut également être obtenue par la combinaison des deux évaluations explicites et implicites.

III.3.1.1 L'évaluation explicite :

Le système invite l'utilisateur via une interface pour fournir des évaluations de l'information qu'on lui ait proposé afin de construire et améliorer son modèle. La précision de la recommandation dépend de la quantité d'évaluations fournies par l'utilisateur. Le seul inconvénient de cette méthode est qu'elle exige plus d'effort des utilisateurs qui ne sont pas toujours prêts à fournir suffisamment d'informations. Malgré ça, l'évaluation explicite fournisse toujours les données les plus fiables, car elle ne dépend pas de l'extraction des préférences depuis les actions, et elle fournit également une transparence dans le processus de recommandation qui se traduit par une qualité plus élevée de recommandations et plus de confiance dans le système [28].

III.3.1.2 L'évaluation implicite :

Le système déduit automatiquement les préférences de l'utilisateur en surveillant ses différentes actions tels que l'histoire des achats, historique de navigation, le temps passé sur certaines pages web, les liens suivis par l'utilisateur, le contenu du courrier électronique et de clics de bouton entre autres. L'évaluation implicite réduit la charge sur les utilisateurs en déduisant leurs préférences à partir de leurs comportements avec le système. Cette méthode ne nécessite pas un effort de l'utilisateur, mais elle est moins précise. En outre, les données de préférences implicites pourraient en réalité être plus objective que celles explicites, car il n'y a pas de biais résultant d'utilisateurs qui ont répondu d'une manière socialement souhaitable [28] et il n'y a pas de problèmes d'image de soi et aucun besoin de maintenir une image pour les autres [29].

III.3.1.3 L'évaluation hybride :

Les points forts des deux évaluations implicites et explicites peuvent être combinés dans un système hybride afin de minimiser leurs faiblesses et obtenir un système mieux performant. Ceci peut être réalisé en utilisant une donnée implicite comme une vérification de l'évaluation explicite ou permettant à l'utilisateur de donner une évaluation explicite seulement quand il choisit d'exprimer un intérêt explicite [26].

III.3.2 Phase d'apprentissage

Dans cette phase, un algorithme d'apprentissage est appliqué pour filtrer et exploiter les fonctionnalités de l'utilisateur depuis les évaluations recueillies dans la phase de collecte d'informations.

III.3.3 Phase de recommandation

Le système peut maintenant recommander ou prédire le genre d'éléments que l'utilisateur peut préférer. Ceci peut être fait soit directement sur la base de l'ensemble de données recueillies dans la phase de collecte d'informations, ou à travers les activités observées de l'utilisateur.

III.4 Techniques de filtrages des recommandations

L'utilisation des techniques de recommandation efficaces et précises est très importante pour un système qui fournira de bon et utile recommandations à ses utilisateurs. On distingue quatre approches pour produire une liste de recommandations [30] :

III.4.1 Filtrage personnalisé

Il s'agit de recommander des objets sur la base du comportement passé de l'utilisateur seulement. Cette technique est destinée pour les systèmes où il est difficile de classifier les articles et le nombre d'utilisateurs est petit.

III.4.1.1 Avantages :

- Rapide à implémenter.
- Ne nécessite pas de grandes informations sur les utilisateurs pour la création de leurs profils.
- Ne nécessite pas de grandes informations sur les objets pour la création de leurs profils.
- Temps de réponse négligeable.

III.4.1.2 Inconvénients :

- Ne peut pas fournir de recommandations pour les nouveaux utilisateurs ou ceux avec peu d'interactions avec le système.
- Ne peut pas recommander de nouveaux objets.

III.4.2 Filtrage basé sur le contenu

La technique à base de contenu met l'accent davantage sur l'analyse des attributs des objets afin de générer des prédictions. La recommandation est faite en basant sur les profils des utilisateurs à l'aide de caractéristiques extraites à partir du contenu des objets que l'utilisateur a évalué dans le passé [31].

III.4.2.1 Avantages :

- Peut recommander de nouveaux objets.
- La précision des recommandations n'est pas affectée gravement pour les nouveaux utilisateurs.

- Adapte rapidement aux changements des préférences des utilisateurs.
- Les utilisateurs peuvent recevoir les recommandations sans avoir besoin de partager leurs profils, qui assure la confidentialité [32].

III.4.2.2 Inconvénients : [33]

- Exige une description riche sur les items.
- Les utilisateurs sont limités à avoir des recommandations similaires aux items déjà définis dans leurs profils.

III.4.3 Filtrage collaboratif

Le filtrage collaboratif est une technique de prédiction dépendante du domaine pour le contenu qui ne peuvent pas facilement et adéquatement être décrit par les métadonnées. Cette technique fonctionne en construisant une base de données (matrice utilisateur-article) des préférences pour les articles par les utilisateurs. Elle correspond aux utilisateurs des intérêts et préférences pertinentes en calculant les similitudes entre leurs profils pour faire des recommandations [34].

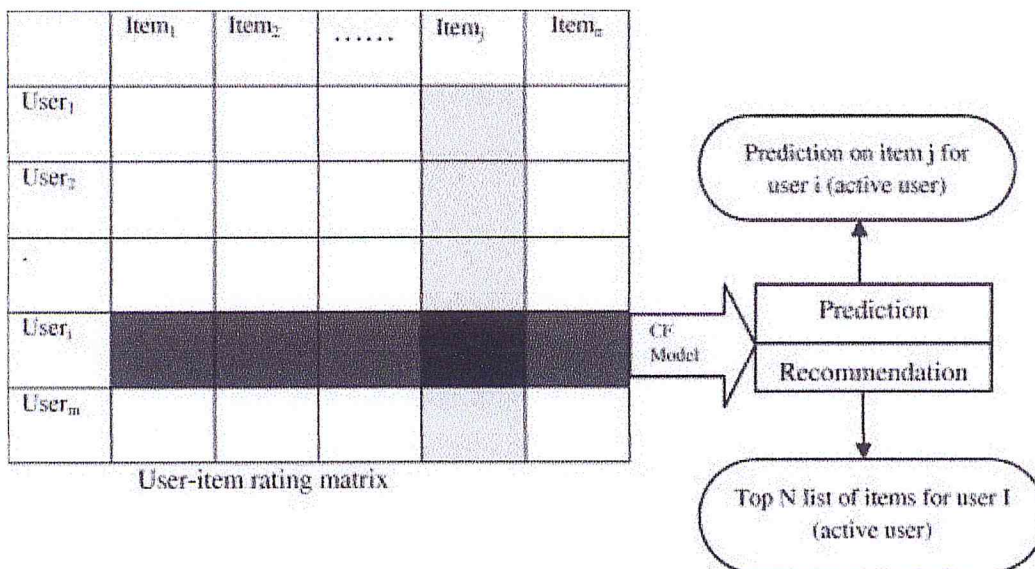


Figure III-1 : Processus de filtrage collaboratif [26]

III.4.3.1 Avantages :

- Excellente performance dans les domaines où les articles manquent de contenu descriptif.
- Peut faire des recommandations pertinentes des articles même si le contenu ne figure pas dans le profil de l'utilisateur [35].

III.4.3.2 Inconvénients :

- **Problème de démarrage à froid** : le profil d'un nouvel utilisateur sera vide car il n'a pas évalué aucun élément, par conséquent, son goût est inconnu pour le système [36].
- **Problème de dispersion des données** : ce problème se produit quand seulement quelques-uns du nombre total d'articles disponibles dans une base de données sont évalués par les utilisateurs [37] qui conduit toujours à une matrice utilisateur-élément creuse.
- **Évolutivité** : ceci est un autre problème associé aux algorithmes de recommandation collaboratifs, car le calcul croît de façon linéaire avec le nombre d'utilisateurs et des éléments [37].
- **Synonymie** : c'est la tendance des éléments très similaires à avoir des noms différents. La plupart des systèmes de recommandation trouvent qu'il est difficile de faire la distinction entre les éléments étroitement liés. Les systèmes de filtrage collaboratif trouvent des difficultés dans ce point, d'où, différentes méthodes telles que l'expansion automatique des termes et la construction d'un thésaurus, sont utilisées. Mais l'inconvénient de ces méthodes est que certains termes ajoutés peuvent avoir des significations différentes de ce qui est prévu, ce qui conduit parfois à la dégradation rapide des performances de recommandation [26].

III.4.4. Filtrage hybride

La technique de filtrage hybride combine les différentes techniques de recommandation afin d'obtenir une meilleure optimisation du système pour éviter certaines limitations et problèmes des systèmes de recommandation purs [38]. L'idée derrière les techniques hybrides est qu'une combinaison d'algorithmes fournira des recommandations plus précises et efficaces que 'un seul algorithme et que les inconvénients d'un algorithme peuvent être surmontés par un autre algorithme [35].

Il existe plusieurs types de filtrage hybride, parmi eux :

III.4.4.1 Hybridation pondérée :

On combine les résultats de différents recommandateurs pour générer une liste de recommandation ou prédiction en intégrant les scores de chacune des techniques utilisées par une formule linéaire [39].

III.4.4.2 Hybridation de commutation :

Le système permute à l'une des techniques de recommandation selon une heuristique qui reflète la capacité de système pour produire une bonne recommandation. L'hybridation de commutation a la capacité d'éviter des problèmes spécifiques à une méthode, par exemple le problème des nouveaux utilisateurs dans le filtrage basé sur le contenu en passant à un système de recommandation collaborative.

III.5 Conclusion

Dans ce chapitre, nous avons présenté la notion du système de recommandation, qui est un système de filtrage qui cherche à présenter les éléments d'information susceptibles d'intéresser l'utilisateur. Ensuite, nous avons vu que les étapes d'un processus de recommandation, qui commence par collecter les informations, puis utilise des algorithmes d'apprentissage, qui vont résulter à des recommandations pertinentes. Enfin, nous avons discuté les techniques de filtrage utilisées dans les systèmes de recommandation.

Maintenant que nous avons établi la littérature qui concerne notre projet, nous devons étudier le cas d'étude qui est présenté sous l'organisme d'accueil, afin d'avoir toutes les informations nécessaires pour concevoir notre système décisionnel.

PARTIE 2 : ETUDE DE L'EXISTANT ET LA COLLECTE DES BESOINS

Cette deuxième partie portera sur la présentation de l'organisme d'accueil et l'existant en matière de décisionnel, afin d'aborder les besoins des utilisateurs dans l'intention de proposer une solution.

Chapitre IV

Présentation de l'organisme d'accueil

IV.1 Introduction

Le marché Algérien a récemment assisté à la naissance d'un grand nombre de petites et grandes entreprises dans différents domaines. Parmi ces domaines, le domaine de l'informatique qui a vu un grand développement. Dans l'Algérie on trouve plusieurs sociétés dans ce domaine, LeaderSoft est une des plus connues dans le marché.

IV.2 LeaderSoft

LeaderSoft est une société qui fournit des services en informatique, dirigée par une équipe de professionnels. En utilisant les technologies Microsoft elle est spécialisée dans le développement de logiciels de gestion, la conception et l'hébergement des sites Web, l'installation des réseaux, et la maintenance des équipements informatiques. Cette dernière travaille avec plusieurs sociétés qui utilisent le logiciel de gestion Eureka pour assurer la bonne gestion de leurs entreprises. L'utilisation de ce logiciel durant plusieurs années résulte à un cumule énorme de données. L'exploitation de ces données est nécessaire pour le développement de n'importe quelle entreprise, mais difficile lors de l'absence d'un système d'information décisionnel. Notre stage se déroule au sein de l'entreprise PTD, une des entreprises qui utilise ce logiciel. Dans ce chapitre nous allons s'intéresser à son historique, son organisation et ses activités dans sa globalité, puis on va étudier les besoins qu'on doit satisfaire

IV.3 Présentation de PTD

IV.3.1 SARL PTD en bref...

SARL PTD (Plastique Transformation et Distribution) est une entreprise de fabrication des produits à base de plastique, créée le 2 août 1997, l'entreprise s'est spécialisée dans la fabrication de la préforme et bouchon en PEHD.

PTD est équipée de 7 chaînes de production, dont 5 ont un double moule qui peuvent créer des préformes de calibre 22/25, 30.5/37, 47/50, 60/65 et 75/82. Pour répondre aux besoins de la clientèle en matière de préformes, PTD dispose d'une capacité de production à même de répondre, dans les délais requis, à la demande de la clientèle. Les produits sont contrôlés et garantis par ses laboratoires de contrôle de la qualité qui veillent à leur conformité, notamment aux normes.

IV.3.2 Les principaux objectifs de l'entreprise

- L'amélioration des performances.
- L'accroissement économique et social.
- La réduction des incidences environnementales liées aux activités de l'entreprise.

- La qualité et la compétitive de ses produits.
- Soigner les relations avec leurs partenaires.
- Conquérir le marché national.

Ces objectifs sont le gage de la pérennité de notre entreprise et de nos produits

IV.3.3 Organisation de PTD

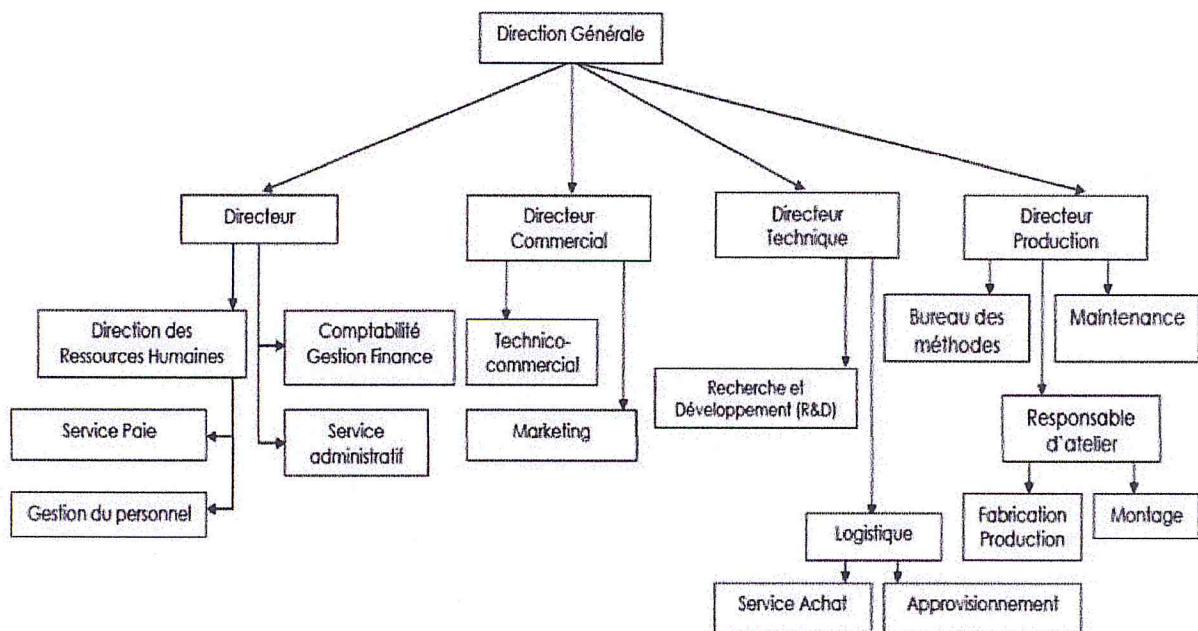


Figure IV-1 : organigramme de PTD

IV.4 Contexte de l'étude

L'étude réalisée au cours de ce mémoire s'est déroulée dans le contexte général d'un suivi des coûts et des quantités dans les trois services suivant : le service commercial, le service logistique et le service de production. On va expliquer le fonctionnement de ces trois services brièvement :

IV.4.1 Service logistique

Ce service a comme tâche d'assurer les approvisionnements de l'entreprise en matière première, emballage etc... Cet tâche est un élément clé de la concurrence d'une entreprise par les incidences qu'elle peut avoir sur ses coûts de production. Ce service doit s'assurer qu'il n'y aura pas des ruptures de stock ni un blocage de fonds en sachant que la plupart des produits sont importés et leur arrivé demande en environ deux mois. Ce service donc doit planifier la satisfaction des besoins actuels et futurs.

IV.4.2 Service de production

Ce service est le centre des activités l'entreprise car c'est une entreprise de fabrication. Parmi ses tâches :

- Il est responsable du bon fonctionnement et de l'organisation générale des ateliers de fabrication et contrôle.
- Il contrôle et améliore l'efficacité la productivité et la qualité
- Il participe à l'élaboration et la modification des dossiers techniques
- Il veille à ce que les cadences prévues et les priorités soient respectées
- Il a les compétences nécessaires pour réaliser des opérations complexes comportant des difficultés techniques, de ce fait, il encadre et assiste les opérateurs pour réaliser les différentes opérations de fabrication.
- Il décide selon les compétences des opérateurs d'organiser la fabrication pour répondre au mieux à la polyvalence nécessaire pour le bon fonctionnement de l'atelier.
- Il est responsable des demandes d'approvisionnement de matériel d'usinage
- Il participe à la formation des opérateurs dans son atelier
- Il s'assure la disponibilité des outillages et du matériel nécessaires en fabrication
- Il est responsable du respect des règles de contrôle lors de l'usinage et de la gestion des outils de contrôle lors de leur utilisation
- Il est responsable de la vente des copeaux et de l'élimination des déchets de fabrication

- Il veille à la maintenance machine et installation

IV.4.3 Service commercial

Le service commercial consiste généralement à la réalisation des ventes, au choix des politiques marketing, et à la distribution des produits. Le service commercial ne travaille pas seul mais en collaboration avec toutes les équipes des autres services. L'utilisation des outils de plus en plus performants permettra au service commercial d'entretenir une bonne relation durable avec les clients en évitant la perte de temps. La réalisation de vente sera plus rapide avec l'adoption de l'e-commerce. Le service commercial est le responsable de la gestion des vente (facture, bon de commande...etc.) et de la réalisation des rapports sur l'état de marché, et sur le développement des vents.

IV.5 Conclusion

A travers ce chapitre nous avons fourni une présentation globale de l'entreprise PTD (Plastique Transformation et Distribution), ses principales activités qui s'entoure sur la fabrication et vente des produits en plastique, et les trois services concernés par notre étude (service logistique, service de production, et le service commercial). Cette présentation nous a également fait comprendre la structuration et l'organisation de la fonction de ces trois services. Cette compréhension est nécessaire pour bien appréhender l'étude de l'existant objet de prochain chapitre.

Chapitre V

Étude de l'existant et Étude des besoins

V.1 Introduction

Afin de répondre mieux aux exigences des décideurs, une bonne compréhension de l'existant est primordiale. Il est en effet nécessaire de comprendre le système existant (fonctionnalités offertes, reporting, limites et données manipulées), car cela constituera le point de départ du système à mettre en place. De plus cela nous permettra de cerner les ressources (les sources de données par exemple) qui nous seront utiles dans notre conception.

V.2 Etat du décisionnel

L'entreprise PTD ne dispose pas d'un système d'information décisionnel, mais elle utilise une solution de reporting qui satisfait le minimum des besoins.

V.2.1 Solution du reporting existante

La direction a confié à LeaderSoft la responsabilité de consolider, de façon manuelle (tableaux croisés Excel), les rapports périodiques, et procéder par la suite à la diffusion des résultats sur les différents départements. Parmi ces rapports :

- Liste des factures par client.
- Bons par client.
- Factures par fournisseur.
- Bons par fournisseur.
- Vente d'articles par client.

V.2.2 Limites du reporting existant

Certes que le reporting offert par LeaderSoft fournit des informations utiles aux décideurs, or il présente aussi des limites :

- Il n'est pas du tout évident de comparer les données entre elles, par exemple : les ventes effectuées dans une période donnée par rapport aux différentes années.
- Le temps de réponse est ainsi très lent, si le décideur souhaite effectuer plusieurs analyses, il doit naviguer entre plusieurs fichiers Excel.
- Les données sont présentées sous un format tabulaire qui offre une lecture limitée et une interprétation difficile (difficultés de basculer entre les niveaux hiérarchiques).
- La préparation des rapports peut prendre plusieurs jours voir des semaines pour être réaliser, ce qui n'est pas apprécié par l'entreprise.

V.3. Etude des besoins

L'étude des besoins des utilisateurs est l'étape la plus importante dans un projet décisionnel. La réussite d'un tel projet dépend d'une bonne compréhension de ces besoins, un système décisionnel qui ne répond pas à cette exigence est obsolète.

Un projet décisionnel requiert du temps et des ressources importantes, avec un risque potentiel d'échec si cette étape est négligée ou mal effectuée.

Dans la définition des besoins des utilisateurs, nous avons opté pour une approche hybride, prenant en compte les besoins des utilisateurs et les données sources, afin de s'assurer que les besoins définis correspondent à ceux des utilisateurs et que les données existantes permettent de répondre à tels besoins.

V.3.1 Définition des besoins

Afin de recenser les besoins pertinents, et de répondre aux attentes des futurs utilisateurs, et aux limites existantes en termes de consolidation et reporting décisionnel, nous avons effectué plusieurs entretiens avec la direction général :

V.2.1.1 Elaboration des questionnaires :

Question	Réponse
Quels sont les processus métiers que vous pilotez dans votre secteur ?	Trois processus métiers : <ul style="list-style-type: none"> • Achat • Production • Vente
Quels sont les principaux problèmes liés à l'analyse et à la prise de décision ?	<ul style="list-style-type: none"> • Problème de production de l'information par rapport aux délais. • Lenteur dans la collecte de l'information • Absence d'un système d'information décisionnel fiable

<p>Quels sont les types d'analyse effectuée avec le système actuel ?</p>	<ul style="list-style-type: none"> • Suivi des ventes • Suivi des achats
<p>Quels sont les bons indicateurs pour la prise de décision ?</p>	<ul style="list-style-type: none"> • Le chiffre d'affaire des achats dans les différentes périodes par rapport aux différents fournisseurs. • La quantité utilisée par rapport à la quantité achetée • La quantité vendue par rapport à la quantité produite • Le taux de réalisation des ventes et des achats • Chiffre d'affaire des ventes dans les différentes périodes par rapport aux différents clients. • Les différents facteurs qui affectent la vente, par exemple la température. • L'état 80-20 des clients.
<p>Quels sont les types d'analyse souhaitez-vous au futur de notre système final ?</p>	<ul style="list-style-type: none"> • Suivi des achats • Suivi de la production et de taux de réalisation • Suivi des ventes

Tableau V-1 : questionnaire d'élaboration des besoins

Après avoir eu ces entretiens on a pu décerner les différents besoins pour chaque service :

V.3.1.2 Service logistique :

Besoins : les utilisateurs auront la possibilité de :

- Suivre les montants des achats et les quantités achetées de la matière première.
- Comparer les achats des différentes dates.
- Comparer les achats des différents fournisseurs.

Les différents suivis se font selon plusieurs axes et mesures d'analyses comme le montrent les dimensions métier dans la **figure V-1**, dont les flèches représentent les axes d'analyses et ce qui a été encadré représente les mesures d'analyses.

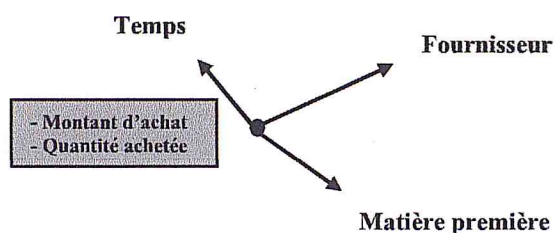


Figure V-1 : les dimensions métier du volet des achats

Indicateur		Description
Montant d'achat	Rôle	Permet de calculer le totale des achats
	Mode de calcul	\sum (prix de produit x quantité achetée)
	Unité de mesure	Devise
	Source de données	Bases de données PTD (2012, 2013, 2014), Table des factures des achats (fact_acht), Table des lignes des achats (l_fact_acht)
Quantité achetée	Rôle	Permettre de connaitre la quantité vendue
	Mode de calcul	\sum (quantité achetée)
	Unité de mesure	Kilo gram
	Source de données	Bases de données PTD (2012, 2013, 2014), Table des lignes des facture (l_fact_acht)

Tableau V-2 : tableau descriptif des indicateurs du volet des achats

V.3.1.3 Service production :

Besoins : les utilisateurs auront la possibilité de :

- Suivre les quantités utilisées dans la production et les quantités achetées.
- Suivre les quantités des produits finales et les quantités vendues.
- Comparer la production des différentes dates.
- Suivre le taux de réalisation.

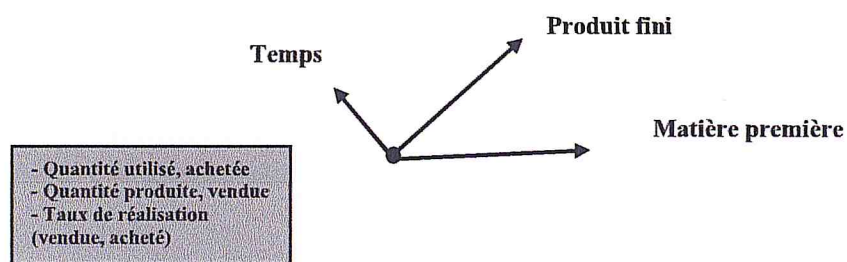


Figure V-2 : les dimensions métier du volet des productions

Indicateur		Description
Quantité utilisé	Rôle	Permettre de connaitre la quantité de la matière première utilisée
	Mode de calcul	\sum (quantité de matière première utilisée)
	Unité de mesure	Gram
	Source de données	Bases de données PTD (2012, 2013, 2014), Table des lignes des factures (l_production)
Quantité achetée	Rôle	Permettre de connaitre la quantité achetée
	Mode de calcul	\sum (quantité de matière première achetée)
	Unité de mesure	Kilo gram
	Source de données	Bases de données PTD (2012, 2013, 2014), Table des lignes des factures (l_fact_acht)

Taux de réalisation des achats	Rôle	Permettre de connaître la qualité des achats effectuée (100% étant le meilleur taux)
	Mode de calcul	$(\text{Quantité utilisé} / \text{Quantité acheté}) \times 100$
	Unité de mesure	%
	Source de données	Bases de données PTD (2012, 2013, 2014), Table des lignes des factures (l_production), Table des lignes des factures (l_fact_acht)
Quantité vendue	Rôle	Permettre de connaître la quantité vendue
	Mode de calcul	$\sum (\text{quantité de produit fini vendue})$
	Unité de mesure	UNITE
	Source de données	Bases de données PTD (2012, 2013, 2014), Table des lignes des factures (l_fact_ven)
Quantité produit	Rôle	Permettre de connaître la quantité des produits finis
	Mode de calcul	$\sum (\text{quantité de produit fini produite})$
	Unité de mesure	UNITE
	Source de données	Bases de données PTD (2012, 2013, 2014), Table des lignes des productions (l_production)
Taux de réalisation des ventes	Rôle	Permettre de connaître le rendement des ventes par rapport aux productions réalisés (100% étant le meilleur taux)
	Mode de calcul	$(\text{Quantité utilisé} / \text{Quantité acheté}) \times 100$
	Unité de mesure	%
	Source de données	Bases de données PTD (2012, 2013, 2014), Table des lignes des factures (l_production), Table des lignes des factures (l_fact_ven)

Tableau V-3 : tableau descriptif des indicateurs du volet des productions

V.3.1.4 Service commercial :

Besoins : les utilisateurs auront la possibilité de :

- Suivre le montant des ventes et la quantité vendue des produits finals.
 - Suivre le changement de température dans les dates des différentes ventes.
 - Comparer les ventes des différentes dates.
 - Comparer les ventes des différents clients.
 - État 80-20 des clients.
- ❖ La règle 80-20 également connue sous le nom de principe de Pareto, la loi des rares vital, et le principe des facteurs éparpillés [40], stipule que, pour de nombreux événements, à peu près 80% des effets proviennent de 20% de la cause [41].

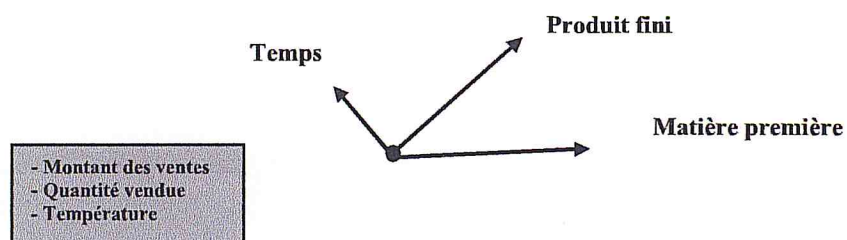


Figure V-3 : Les dimensions métier du volet des ventes

Indicateur		Description
Montant des ventes	Rôle	Permet de calculer le totale des vente
	Mode de calcule	\sum (prix de produit x quantités vendue)
	Unité de mesure	Devise
	Source de données	Bases de données PTD (2012, 2013, 2014), Table des factures des ventes (fact_ven), table des ligne des facture (l_fact_ven)

Température	Rôle	Permettre de connaître la température dans la date de vente
	Mode de calcul	AVG (température, date)
	Unité de mesure	C°
	Source de données	Fichier CSV
Quantité vendue	Rôle	Permettre de connaître la quantité vendue
	Mode de calcul	Σ (quantité vendue)
	Unité de mesure	UNITE
	Source de données	Bases de données PTD (2012, 2013, 2014), Table des lignes des facture (l_fact_ven)

Tableau V-4 : tableau descriptif des indicateurs du volet des ventes

V.4 Conclusion

A cette étape cruciale de notre projet, nous avons commencé par étudier les limites du système existant qui dépend des rapports et états des ventes réalisés par l'entreprise LeaderSoft. Cette étude nous a fait réaliser que la solution actuelle des rapports par commande n'est pas du tout efficace, d'où la nécessité d'un système décisionnel.

Pour bien comprendre les besoins des utilisateurs, nous avons entamé à la réalisation des questionnaires des utilisateurs clés (direction générale), qui nous a permis à déterminer les axes d'analyse ainsi que les processus à modéliser. Puis nous avons confronté notre définition des besoins aux utilisateurs pour s'assurer que cela correspond bien à ce qu'ils recherchent, nous nous sommes assurés de la possibilité de répondre à ces besoins avec les données disponibles.

Cette définition des besoins, permettra la conception du système, abordée dans le chapitre suivant.

PARTIE 3 : CONCEPTION DE LA SOLUTION

La solution proposée va être conçue en plusieurs étapes. La solution étant un système décisionnel s'articulant autour d'un entrepôt de données, on va donc concevoir cet entrepôt, sa zone d'alimentation, et sa zone de restitution en passant par la conception des cubes. Puis nous allons passer à la conception du système de recommandation des requêtes MDX.

Chapitre VI

Conception de l'entrepôt de données

VI.1 Introduction

Dans ce chapitre, nous présentons la conception de la zone d'entreposage qui constitue la première étape de notre conception. À l'issue de cette étape, nous aurons les mesures avec les tables de faits qui les contiennent, ainsi que les dimensions selon lesquelles les mesures vont être analysées. La démarche de **Ralph Kimball**, que nous avons suivie dans cette conception, comprend les étapes suivantes [43] :

- Sélectionner les activités à modéliser (étape effectuée lors de l'étude des besoins),
- Déclarer le grain de chaque activité (niveau de détail des données),
- Choisir les dimensions participantes à chaque activité (les axes selon lesquels les processus seront analysés),
- Identifier les faits numériques (ce que nous voulons mesurer dans l'analyse).

VI.2 Activité de vente (commercialisation)

VI.2.1 Présentation de l'activité

La commercialisation est l'activité la plus importante dans notre étude, parce qu'elle porte la mesure qui définit de plus ou moins le succès d'une entreprise, qui est le **chiffre d'affaire**. Une bonne étude de cette activité ne va pas seulement résulter à des changements dans cette dernière, mais va étendre aux autres activités. En analysant le marché, on peut ajuster la production et optimiser les achats, ce qui va bénéficier l'entreprise en total.

VI.2.2 Le fait « ventes »

Ce fait va traiter toute information qui concerne les ventes réalisées dans l'entreprise.

VI.2.2.1 Le grain :

La déclaration du grain permet de spécifier ce qu'une ligne de la table de faits représente. Le grain doit être le plus fin possible, ce qui veut dire qu'il serait avantageux de retenir les données les plus atomiques possibles (voir I.3.3.4).

Dans notre cas, le grain correspond à : connaître les ventes réalisées, par **Jour**, par **Produit fini**, et par **Client**.

VI.2.2.2 Les dimensions :

Puisque certaines dimensions participent à plusieurs activités, on va seulement les citer avec chaque activité, et laisser leurs définitions dans VI.5. Les dimensions participantes dans ce fait :

- Client : DimClient.
- Produit fini : DimArticleFinished.
- Temps : DimTime.

VI.2.2.3 Les mesures :

- La quantité vente des articles finis : « Quantity »
- Le chiffre d'affaire des ventes : « Amount »
- La température moyenne pendant la période des ventes en question : « Temperature »
- Le bénéfice net des ventes : « Benefits »

VI.2.2.4 Détails du fait :

FactSales			
Column Name	Data Type	Allow Nulls	
ClientID	varchar(30)		<input type="checkbox"/>
ArticleFinishedID	varchar(30)		<input type="checkbox"/>
Date	date		<input type="checkbox"/>
Quantity	float		<input checked="" type="checkbox"/>
Amount	float		<input checked="" type="checkbox"/>
Temperature	float		<input checked="" type="checkbox"/>
Benefits	float		<input checked="" type="checkbox"/>

Figure VI-1 : Le fait « vente »

Vente	
Attributs	Description
ClientID	Le code du client
ArticleFinishedID	Le code de l'article fini
Date	La date de vente
Quantity	La quantité des produits en unité
Amount	Le chiffre d'affaire
Temperature	La température moyenne dans une date
Benefits	Le bénéfice net

Tableau VI-1 : Tableau descriptif du fait « vente »

VI.3 Activité d'achat

VI.3.1 Présentation de l'activité

L'activité d'achat est une activité indispensable pour la production des articles finis. Pour cela PTD doit s'assurer de ne pas avoir des ruptures en stock, ni avoir un stock qui bloque les fonds.

VI.3.2 Le faits « achat »

Ce fait va traiter les achats effectués de la matière première.

VI.3.2.1 Le grain :

Le grain de ce fait correspond à : connaître la production réalisée, par **Jour**, par **Matière première**, et par **Fournisseur**

VI.3.2.2 Les dimensions :

Les dimensions participantes dans ce fait :

- Fournisseur : DimSupplier.
- Matière première : DimArticleRaw.
- Temps : DimTime.

VI.3.2.3 Les mesures :

- La quantité achète de la matière première : « Quantity »
- Le chiffre d'affaire des achats : « Amount »

VI.3.2.4 Détails du fait :

FactPurchase			
	Column Name	Data Type	Allow Nulls
PK	SupplierID	varchar(30)	<input type="checkbox"/>
PK	ArticleRawID	varchar(30)	<input type="checkbox"/>
PK	Date	date	<input type="checkbox"/>
	Quantity	float	<input checked="" type="checkbox"/>
	Amount	float	<input checked="" type="checkbox"/>

Figure VI-2 : le fait « Achat »

Achat	
Attributs	Description
SupplierID	Le code du fournisseur
ArticleRawID	Le code de la matière première
Date	La date d'achat
Quantity	La quantité des produits en gram
Amount	Le chiffre d'affaire

Tableau VI-3 : Tableau descriptif du fait « Achat »

VI.4 Activité de production

VI.4.1 Présentation de l'activité

L'activité principale de l'entreprise PTD est la production des préformes plastiques à partir de la matière première. Dans la partie de l'étude des besoins nous avons compris que l'intérêt analytique principale de l'entreprise dans l'activité de production est d'étudier la réalisation, soit en termes de matière première, ou bien de produits finis. D'où, nous avons opter de traiter cette étude en séparant l'entrée et la sortie de la production.

VI.4.2 Le fait « Production In »

Ce fait va traiter la réalisation en termes de quantité matière première achetée par rapport à celle utilisée dans la production.

VI.4.2.1 Le grain :

Le grain correspond à : connaître la quantité achetée et utilisée par **Jour**, et par **Matière première**.

VI.4.2.2 Les dimensions :

Les dimensions participantes dans ce fait :

- Matière première : DimArticleRaw.
- Temps : DimTime.

VI.4.2.3 Les mesures

- La quantité achetée : « QuantityPurchased »
- La quantité utilisée : « QuantityUsed »
- Le taux de réalisation : « RealisationRate »

VI.4.2.4. Détails du fait :

FactProductionIn			
	Column Name	Data Type	Allow Nulls
▶	ArticleRawID	varchar(30)	<input type="checkbox"/>
▶	Date	date	<input type="checkbox"/>
	QuantityPurchased	float	<input checked="" type="checkbox"/>
	QuantityUsed	float	<input checked="" type="checkbox"/>
	RealisationRate	float	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Figure VI-3 : Le fait « Production In »

Production In	
Attributs	Description
ArticleRawID	Le code de la matière première
Date	La date d'achat/production
QuantityPurchased	La quantité des produits achetés en gram
QuantityUsed	La quantité utilisée dans la production en gram
RealisationRate	Le taux de réalisation = matière première utilisée / matière première achetée

Tableau VI-3 : Tableau descriptif du fait « Production In »

VI.4.3 Le fait « Production Out »

Ce fait traite la réalisation en termes de quantité de produit fini vendu par rapport à celle fabriquée dans la production.

VI.4.3.1 Le grain :

Le grain correspond à : connaître la quantité produite et vendue par **Jour**, et par **Produit fini**.

VI.4.3.2 Les dimensions

Les dimensions participantes dans ce fait :

- Produit fini : DimArticleFinished.
- Temps : DimTime.

VI.4.2.3 Les mesures :

- La quantité produite : « QuantityProduced »
- La quantité vendue : « QuantitySold »
- Le taux de réalisation : « RealisationRate »

- Détails du fait :

FactProductionOut			
	Column Name	Data Type	Allow Nulls
PK	ArticleFinishedID	varchar(30)	
PK	Date	date	
	QuantityProduced	float	Y
	QuantitySold	float	Y
	RealisationRate	float	Y

Figure VI-4 : Le fait « Production Out »

Production Out	
Attributs	Description
ArticleFinishedID	Le code de produit fini
Date	La date de production/vente
QuantitySold	La quantité des produits vendues
QuantityProduced	La quantité des produits finis
RealisationRate	Le taux de réalisation = produit vendu / produit réalisé

Tableau VI-4 : Tableau descriptif du fait « Production Out »

VI.5 Les dimensions

VI.5.1 Dimension temps « Time »

C'est une dimension triviale dans un système décisionnel, elle permet de faire des regroupements temporels selon le jour, la semaine, le mois, le trimestre, le semestre, et l'année. Les décideurs ont besoin de suivre leur activité d'un jour à un autre et d'en garder l'historique

Column Name	Data Type	Allow Nulls
Date	date	<input type="checkbox"/>
Day	int	<input checked="" type="checkbox"/>
DayofWeekName	varchar(10)	<input checked="" type="checkbox"/>
Week	int	<input checked="" type="checkbox"/>
MonthName	varchar(10)	<input checked="" type="checkbox"/>
Quarter	int	<input checked="" type="checkbox"/>
Half	int	<input checked="" type="checkbox"/>
Year	int	<input checked="" type="checkbox"/>

Figure VI-5 : La dimension « Time »

Time	
Attributs	Description
Date	Le jour en forme yyyy-mm-dd et c'est la clé de la table DimTime
DayofWeekName	Le jour de la semaine en lettre (jeudi...)
Week	Le numéro de la semaine
MonthName	Le mois en lettre
Quarter	Le numéro du trimestre
Half	Le numéro du semestre
Year	L'année en forme yyyy

Tableau VI-5 : Tableau descriptif de la dimension « Time »

VI.5.2 Dimension « Client »

Cette dimension contient toutes les informations nécessaires concernant les clients, elle contient l'id du client, son nom, et sa la wilaya.

DimClient			
	Column Name	Data Type	Allow Nulls
▶	ClientID	varchar(30)	<input type="checkbox"/>
	ClientName	varchar(50)	<input checked="" type="checkbox"/>
	ClientWilaya	varchar(50)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Figure VI-6 : La dimension « Client »

Client	
Attributs	Description
ClientID	Le code du client
ClientName	Le nom du client
ClientWilaya	La wilaya du client

Tableau VI-6 : Tableau descriptif de la dimension « Client »

VI.5.3 Dimension fournisseur « Supplieur »

Cette dimension contient toutes les informations des fournisseurs, leurs IDs, noms, types de payement, état, et pays.

DimSupplier			
	Column Name	Data Type	Allow Nulls
PK	SupplierID	varchar(30)	<input type="checkbox"/>
	SupplierName	varchar(50)	<input checked="" type="checkbox"/>
	SupplierPaymentType	varchar(50)	<input checked="" type="checkbox"/>
	SupplierState	varchar(50)	<input checked="" type="checkbox"/>
	SupplierCountry	varchar(50)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Figure VI-7 : La dimension « Supplier »

Supplier	
Attributs	Description
SupplierID	Le code du fournisseur
SupplierName	Le nom du fournisseur
SupplierPaymentType	Le type de paiement du fournisseur (cheque, traite, banque...)
SupplierState	La wilaya du fournisseur
SupplierCountry	Le pays du fournisseur

Tableau VI-7 : Tableau descriptif de la dimension « Supplier »

VI.5.4 Dimension matière première « ArticleRaw »

Cette dimension contient toute les informations sur la matière première, l'id de la matière, son nom, et sa famille.

DimArticleRaw			
	Column Name	Data Type	Allow Nulls
🔑	ArticleRawID	varchar(30)	<input type="checkbox"/>
	ArticleName	varchar(50)	<input checked="" type="checkbox"/>
	ArticleFamily	varchar(50)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Figure VI-8 : La dimension « ArticleRaw »

ArticleRaw	
Attributs	Description
ArticleID	Le code de l'article
ArticleName	Le nome de l'article
ArticleFamily	La famille de l'article (emballage...)

Tableau VI-8 : Tableau descriptif de la dimension « ArticleRaw »

VI.5.5 Dimension produit fini « ArticleFinished »

Cette dimension contient toute les informations sur les produits finis, leurs IDs, leurs noms, et leurs types.

DimArticleFinished			
	Column Name	Data Type	Allow Nulls
🔑	ArticleFinishedID	varchar(30)	<input type="checkbox"/>
	ArticleName	varchar(50)	<input checked="" type="checkbox"/>
	ArticleType	varchar(30)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Figure VI-9 : La dimension « ArticleFinished »

ArticleFinished	
Attributs	Description
ArticleID	Le code de l'article
ArticleName	Le nom de l'article
ArticleType	Le type de l'article (38 gr, 52 gr...)

Tableau VI-9 : Tableau descriptif de la dimension « ArticleFinished »

VI.6 Dimensions participantes pour chaque fait

Fait \ Dimension	Time	ArticleRaw	ArticleFinished	Client	Supplier
Sales	✓	✓	✓	✓	
Purchase	✓	✓		✓	
ProductionIn	✓	✓		✓	✓
ProductionOut	✓	✓		✓	✓

Tableau VI-10 : Tableau descriptif des dimensions participantes pour chaque fait

VI.7 Schéma globale de l'entrepôt

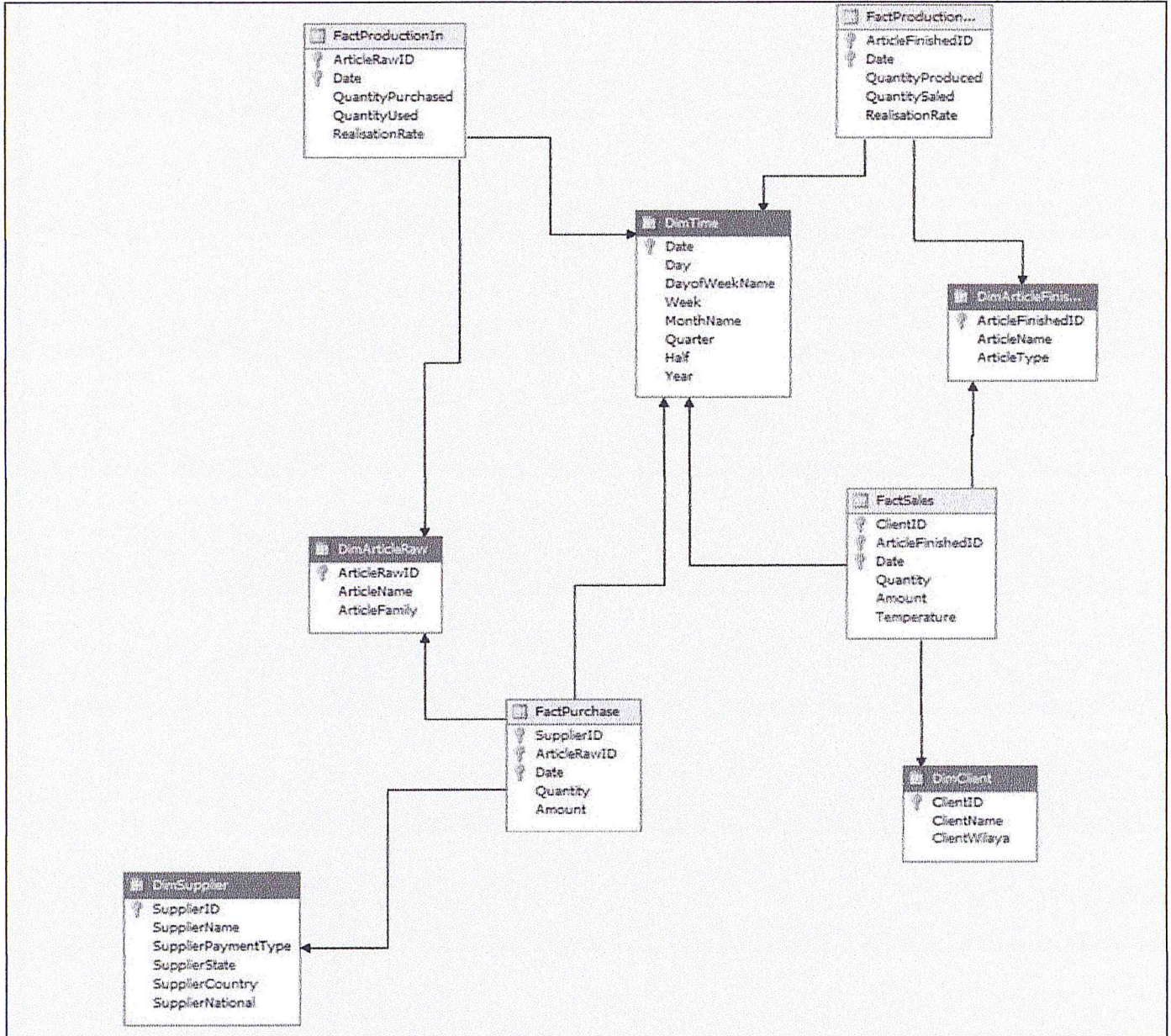


Figure VI-10 : schéma en constellation de l'activité « Production »

VI.8 Conclusion

Au terme de cette première étape, nous avons identifié les faits, leurs mesures numériques, et les dimensions selon lesquelles seront analysés ces faits. Et cela en tenant compte des besoins des utilisateurs définis précédemment.

Conformément à la définition des besoins, nous avons quatre faits : **Sales**, **Purchase**, **Production In**, **Production Out**. Ces quatre volets constituent un schéma en constellation qui va constituer l'entrepôt. Cet entrepôt que nous avons conçu va être exploité par décideurs, mais avant cela il est nécessaire de l'alimenter.

Nous passons donc à l'étape suivante qui est la construction de la zone d'alimentation de l'entrepôt.

Chapitre VII

Conception de la zone d'alimentation

VII.I Introduction

L'alimentation de l'entrepôt de données est une étape importante dans le projet décisionnel, car elle garantira la pertinence et la qualité des données que contiendra l'entrepôt et assurera au décideur l'accès à la bonne information. L'alimentation se fait grâce à l'opération d'ETL, il s'agira d'extraire les données pertinentes ce qui nécessite une bonne connaissance des sources de données, puis de transformer les données au format voulue, pour enfin les charger selon une périodicité précise et de la manière adaptée. Nous présentons dans ce chapitre les différentes étapes que nous avons suivi pour l'alimentation de notre entrepôt.

VII.2 Identification des sources de données

Lors de l'étape de l'identification des besoins, nous avons adopté une démarche qui assure que les besoins définis correspondent à ceux des utilisateurs (analystes, décideurs) et que les données existantes permettent de répondre à de tels besoins. De plus l'étape de conception de l'entrepôt a abouti un schéma en constellation qui indique les données nécessaires.

Comme vu précédemment, nos sources de données sont les bases de données de l'entreprise suite à l'utilisation de logiciel de gestion **Eureka**.

L'outil utilisé dans le processus d'ETL est **Microsoft business intelligence studio 2008 Integration Services** qui est à la fois gratuit et puissant pouvant supporter une charge importante de données.

VII.3 Extraction des données

Une fois les données identifiées, nous procédons à leur extraction, nous allons récupérer les données jugées pertinentes, destinées à être exploitées et en accord avec le résultat à obtenir à partir des bases de données *PTD 2012, 2013, 2014* on a pu extraire les données suivant :

- Fiche de client (ClientID, ClientName, ClientWilaya).
- Fiche de fournisseur (SupplierID, SupplierName, SupplierPaymentType, SupplierState, SupplierCountry, SupplierNationale).
- Fiche de l'article (ArticleID, ArticleName, ArticleFamilly)

On a introduit une autre entrée de données qui consiste à ajouter la **température** moyenne de la journée de la base de données *weather* qui est utiliser dans l'analyse :

- Temperature moyenne (date, temperature)

VII.4 Transformation

Les données extraites subissent des transformations avant d'être chargées dans l'entrepôt, nous distinguerons les opérations de :

- **Filtrage** : le filtrage nous a permet de faire des restrictions sur les données nulles, les champs vides...etc., dans notre cas l'utilisation de filtrage est d'éliminer le chargement :
 - Des associations ou le montant non présent
 - Des produits non utilisés dans une transaction.
- **Conversion** : la conversion consiste à changer le type de données de certains attributs, les unités de mesures...etc., par exemple :
 - **Date:** date (mm-dd-yyyy) → date (yyyy-mm-dd).
 - **Date Weather** varchar → date (yyyy-mm-dd).
- **Changement des libellés des champs** : le changement des libellés permet d'éviter des conflits qui se produisent lorsqu'on utilise des noms différents pour le même attribut, ainsi pour rendre les champs plus significatifs. Par exemple l'unité comptable des associations se trouve avec des différents libellés dans plusieurs fichiers sources :

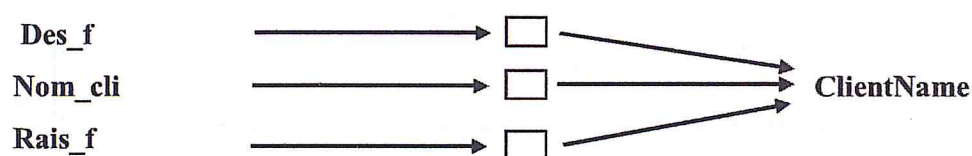


Figure VII-1 : Exemple de changement des libellés

- **Transformation de la table article** : comme mentionné dans le chapitre précédent (VI.4.1), on a séparé la table article en deux tables
 - La première est ArticleFinished qui contient les produits finis uniquement groupé par calibre (ArticleType)
 - La deuxième est ArticleRaw qui contient la matière première

De plus de ces opérations on a utilisé les opérations de sélection (sélectionner le mois ou l'année d'une date donnée), et de regroupement des données. Ces transformations se font généralement au moment du mapping (correspondance entre les données).

VII.5 Chargement

Le chargement des dimensions et des tables de faits. Le chargement des tables de faits doit venir après le chargement des dimensions car les premières référencent les secondes à l'aide des clés étrangères.

La figure suivante illustre le processus de chargement :

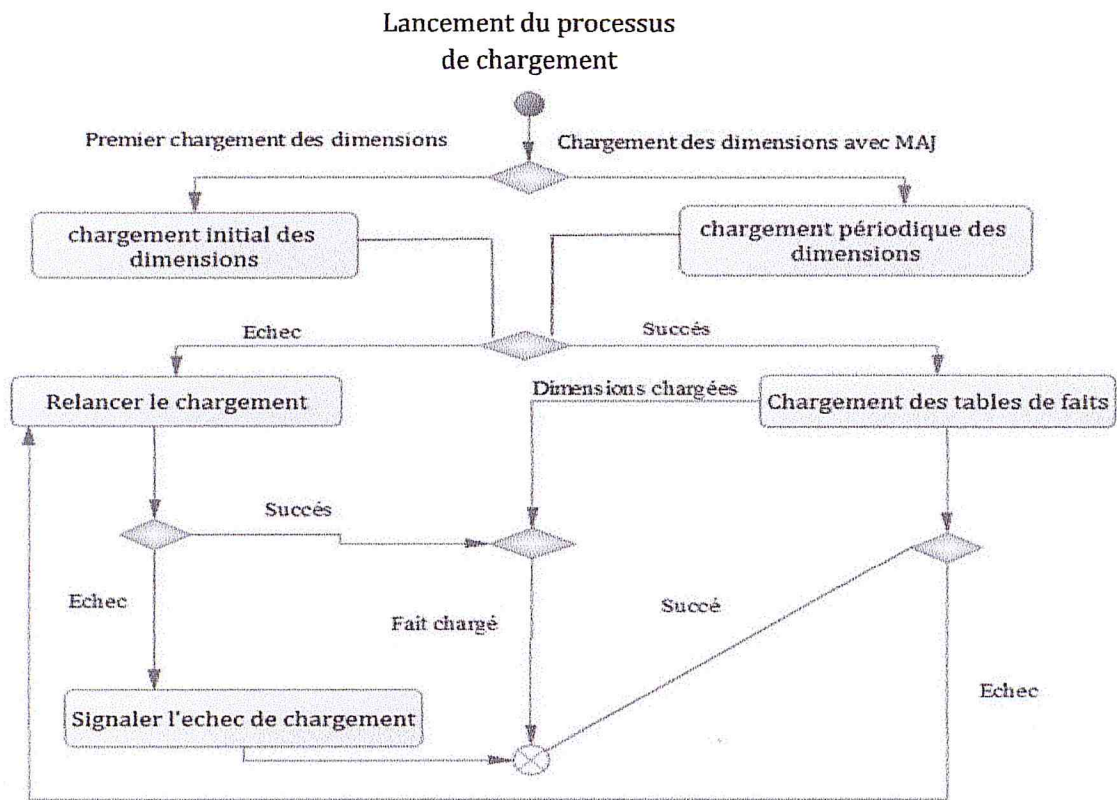


Figure VII-2 : Diagramme d'activité de processus de chargement.

VII.6 Conclusion

Dans ce chapitre nous avons abordé la conception de la zone d'alimentation qui va nous assurer des données fiables et de qualité dans l'entrepôt de données. Nous avons vu les différentes étapes : identification des sources de données, extraction, transformation et enfin chargement.

Cette étape achevée nous passons dans le chapitre suivant, à la conception des cubes de données.

Chapitre VIII

Conception des cubes dimensionnels

VIII.1 Introduction

Dans ce chapitre nous abordons la conception des cubes dimensionnels, une structure très souple à l'interrogation des données offrant une navigation adaptée et une exploitation. Un cube correspond à un sujet d'analyse : un fait décrit par les besoins des utilisateurs, il contient un ensemble d'indicateurs en hiérarchies, les hiérarchies étant décomposées en niveaux. La navigation peut se faire de manière croisée (interroger la valeur d'un indicateur selon trois axes par exemple), de manière ascendante ou descendante en jouant sur le niveau de granularité des axes d'analyse et de manière spécifique à l'aide de filtres.

L'expression des besoins des utilisateurs a conduit à la conception entre autres un schéma en constellation (VI-10) concernant les trois activités : achat, production et vente. Ce schéma correspond aux cubes qui seront conçus. Nous détaillons, dans ce qui suit la conception de chaque cube.

VIII.2 Structure hiérarchique des dimensions

Les hiérarchies de chaque dimension ainsi que les niveaux correspondants à chaque hiérarchie vont définir une structure hiérarchique. Cette structure permettra la navigation à l'intérieur de la dimension à travers les opérations de « drill down » (accès à un niveau de détail inférieur) ou de « roll up » (accès à un niveau de détail supérieur). Par exemple, l'utilisateur peut analyser la production par année (sommet de la hiérarchie) puis par mois (opération de forage vers le bas) etc...

Le tableau suivant regroupe les dimensions de l'ensemble des cubes, avec la définition des hiérarchies, niveaux et colonnes correspondantes.

Dimension	Hiérarchies	Niveaux	Colonnes
DimTime	All -> N1 -> N2 -> N3 -> N4 -> N5 -> N6	Niveau 1 = N1	Year
		Niveau 2 = N2	Half
		Niveau 3 = N3	Quarter
		Niveau 4 = N4	MonthName
		Niveau 5 = N5	Week
		Niveau 6 = N6	DayofWeekName
			Day
DimArticleFinished	All -> N1 -> N2	Niveau 1 = N1	ArticleType
		Niveau 2 = N2	ArticleName
			ArticleID
DimArticleRaw	All -> N1 -> N2	Niveau 1 = N1	ArticleFamily
		Niveau 2 = N2	ArticleName
			ArticleID

DimClient	All -> N1 -> N2	Niveau 1 = N1	ClientWilaya
		Niveau 2 = N2	ClientName
			ClientID
DimSupplier	All -> N1 -> N2 -> N3 -> N4	Niveau 1 = N1	SupplierCountry
		Niveau 2 = N2	SupplierState
		Niveau 3 = N3	SupplierPaymentType
		Niveau 4 = N4	SupplierName
			SupplierID

Tableau VIII-1 : Tableau donnant les niveaux et les hiérarchies de chaque dimension

VIII.3 Listes des cubes

Le tableau suivant liste tous les cubes à réaliser avec, pour chacun, les mesures correspondantes et les dimensions participantes

Nom de cube	Mesures	Dimensions
FactPurchase	Quantity	DimSupplier
	Amount	DimArticleRaw
		DimDate
FactProductionIn	QuantityPurchased	DimArticleRaw
	QuantityUsed	DimDate
	RealisationRate	
FactProductionOut	QuantityProduced	DimArticleFinished
	QuantitySold	DimDate
	RealisationRatio	

FactSales	Quantity	DimClient
	Amount	DimArticleFinished
	Temperature Benefits	DimTime

Tableau VIII-2 : Liste des cubes

VIII.4 Schémas des cubes

VIII.4.1 Fait « Purchase »

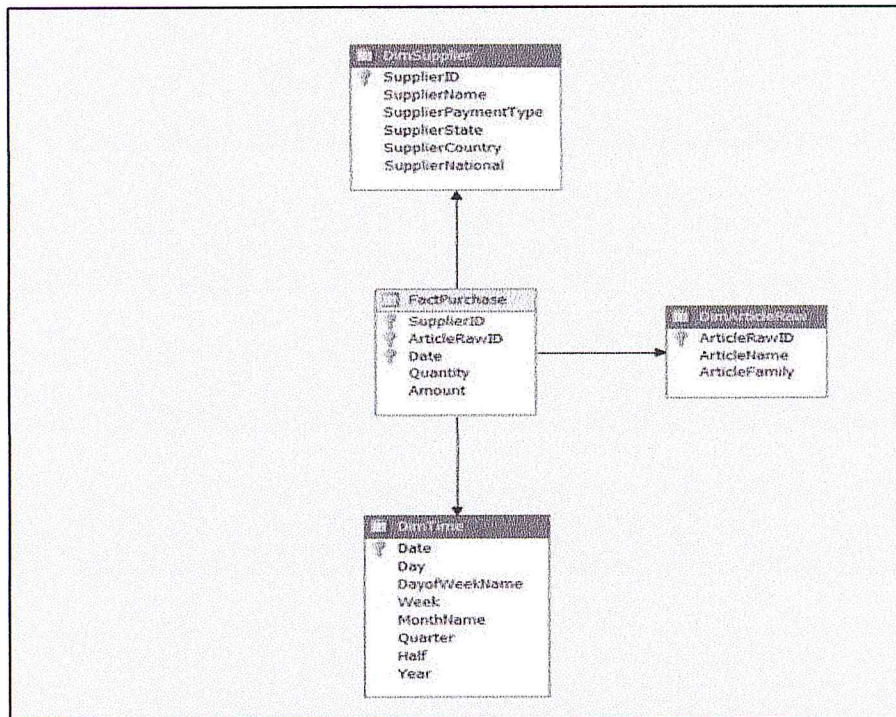


Figure VIII-1 : Cube dimensionnel « Purchase »

VIII.4.2 Fait « Production In »

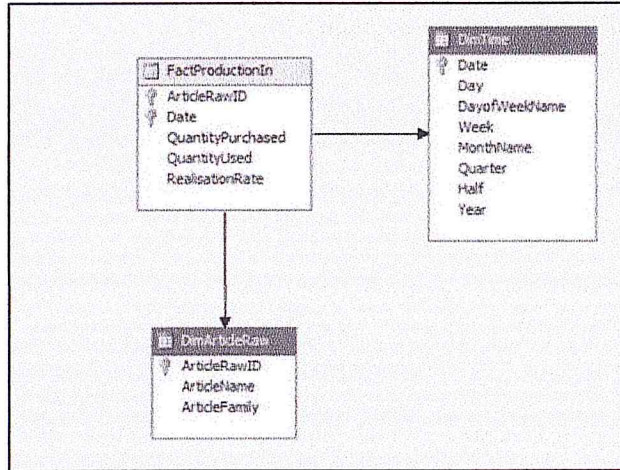


Figure VIII-2 : Cube dimensionnel « Production in »

VIII.4.3 Fait « Production Out »

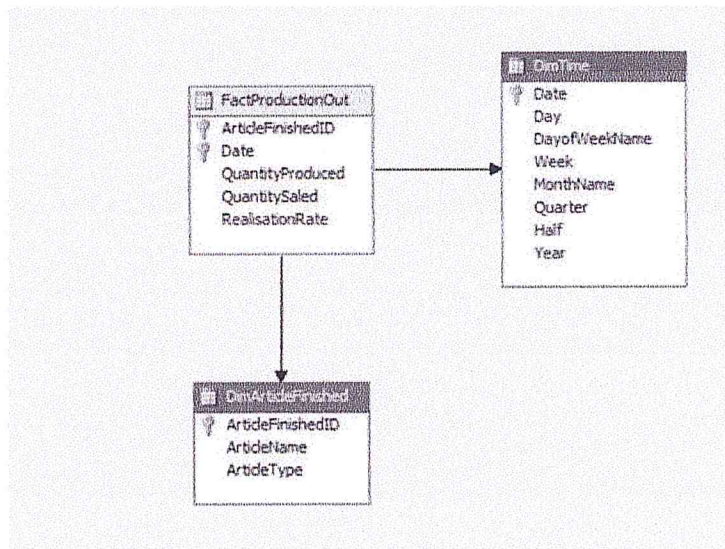


Figure VIII-3 : Cube dimensionnel « Production Out »

VIII.4.4 Fait « Sales »

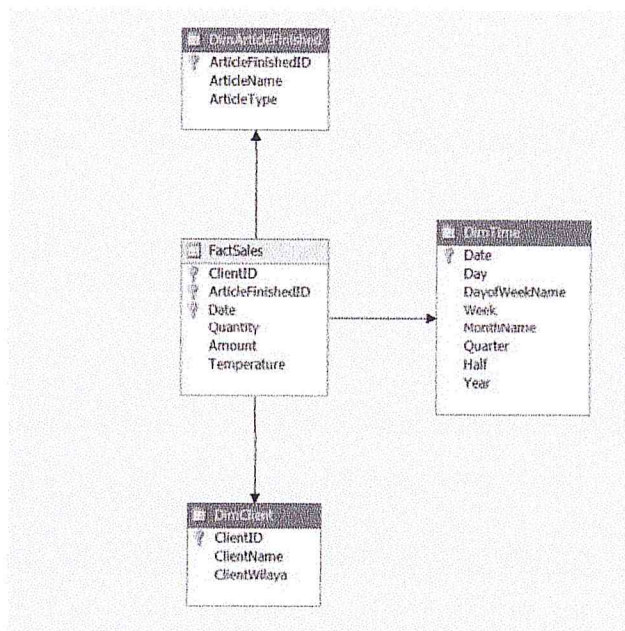


Figure VIII-4 : Cube dimensionnel « Sales »

VIII.5. Conclusion

Dans ce chapitre nous avons conçu les cubes dimensionnels qui vont permettre d'effectuer des analyses multidimensionnelles sur les données. Nous avons défini les mesures, les dimensions participantes ainsi que la structure hiérarchique pour chaque dimension. Ces cubes sont relatifs aux trois activités : achat, production et vente. Ils offrent la possibilité aux décideurs d'exploiter les informations relatives à ces activités, grâce à une navigation rapide, intuitive et personnalisable (choix des axes d'analyse, du niveau du détail...).

Chapitre IX

Conception du système de recommandation

IX.1 Introduction

Dans ce chapitre, nous présentons la conception du système de recommandation qui constitue la dernière étape de notre conception. À l'issue de cette étape, nous aurons un système capable de recommander aux utilisateurs les requêtes MDX pertinentes et leurs guider dans leurs sessions d'analyse. Les étapes qu'on va suivre pour concevoir ce système seront :

- Conception de la phase de collecte d'information.
- Conception de la phase d'apprentissage.

IX.2 Technique de filtrage « hybride »

Nous avons choisi une technique hybride personnalisé qui combine les différentes techniques de recommandation, car elle nous permet de profiter des avantages et éviter les problèmes de chaque technique.

IX.2.1 Phase de collecte d'information

Cette phase consiste à collecter le maximum d'informations concernant les activités de l'utilisateur afin de construire son profil. Pour faire cela, nous avons choisi d'utiliser l'évaluation implicite. Le système va donc déduire automatiquement les préférences de l'utilisateur en surveillant ses différentes actions, dans notre cas la navigation dans les cubes qui se manifeste comme des requêtes MDX envoyées au serveur OLAP.

IX.2.1.1 Les informations à collecter :

Les informations qui vont être collecter sont toujours en relation avec les requêtes MDX lancées par l'utilisateur, et pour chaque requête exécutée on collecte :

- Le nom d'utilisateur exécutant la requête.
- Le type de l'utilisateur exécutant la requête (service commercial, gérant, ...).
- Le texte de la requête exécutée.
- Le texte de la requête qui précède la requête exécutée si elle en existe.
- La date et le temps dans lesquelles la requête est exécutée.
- Le temps épuisé avec la requête avant passer à une autre activité.
- Si la requête exécutée est transitoire ou bien la dernière dans la session d'analyse (la dernière requête dans une session veut dire généralement que l'utilisateur a atteint son objectif).

IX.2.1.2 Structure de stockage :

Nous avons opté pour stocker les informations des requêtes MDX dans une petite base de données avec trois tables qui gardent toutes les informations nécessaires sur ces requêtes. La structure de stockage est représentée avec le diagramme de classe dans la figure IX-1 et les attributs des classes sont expliqués dans le tableau IX-1 :

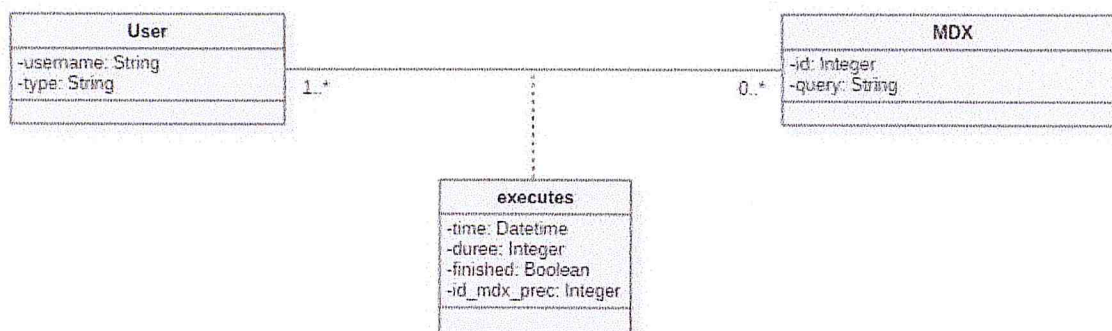


Figure IX-1 : diagramme de class de la structure de stockage des requêtes MDX

	Champs	Signification
Utilisateur (user)	username	Nom de l'utilisateur
	type	Fonction (service commercial, ...)
Requête (mdx)	id	Identifiant de la requête
	query	Texte de la requête
Executes	time	Temps et date de l'exécution
	duree	Temps épuisé durant la requête
	finished	Si la requête est la dernière
	Id_mdq_prec	Identifiant de la requête précédente

Tableau IX-1 : description de la structure de stockage

IX.2.1.3 Algorithmes de collecte :

Pour collecter les informations qui vont peupler la structure de stockage nous utiliserons deux algorithmes qui stockent tous ce qui concernent l'exécution d'une requête MDX quelconque. Ces deux algorithmes sont très similaires mais ils diffèrent dans ses raisons de déclenchement et dans l'enregistrement du type d'exécution de la requête en question (requête transitoire, ou pas).

IX.2.1.3.1 Procédure de collecte 1 :

Le premier algorithme est déclenché quand une requête MDX est exécutée par un utilisateur quelconque. À noter que cet algorithme n'enregistre pas les informations de la requête déclenchante mais celles de la requête qui la précède. Ceci est parce qu'on ne peut pas savoir le temps écoulé (durée d'analyse) et le type de l'exécution de la requête déclenchante (transitoire ou bien la dernière dans la session) tant qu'elle est encore ouverte, mais on peut savoir ces informations quand elles concernent la requête qui la précède.

Algorithme IX-1 : Procédure de collecte 1 (lancé lors de l'exécution d'une requête)

```

1 : {
2 :   Entrées :
           SESSION = {currentUser, currentType, lastMDX, preMDX, timeStamp} « 1 »,
           HISTORIQUE = {user, MDX, executes} « 2 » ;
3 :   si (SESSION.lastQuery ∉ HISTORIQUE.MDX)
           alors ajouterRequete (SESSION.lastQuery, HISTORIQUE.MDX) « 3 »
           fin si
4 :   ajouterExecution (SESSION.currentUser, SESSION.lastQuery, SESSION.timeStamp,
           false, SESSION.preMDX, HISTORIQUE.executes) « 4 »
5 : }
```

« 1 » : La session courante dans le système qui contient l'utilisateur courant, son type, sa dernière requête exécutée dans cette session, la requête qui précède la dernière requête (si elle existe), et le libellé de temps de l'exécution.

« 2 » : La base de données qui contient l'historique des tables des utilisateurs, des requêtes MDX, et des exécutions (voir Figure IX-1).

« 3 » : Ajoute la requête lastQuery dans la table MDX de la base de données HISTORIQUE.

« 4 » : Ajoute les informations de l'exécution a la table HISTORIQUE.executes avec :

HISTORIQUE.executes.time = SESSION.timeStamp

HISTORIQUE.executes.duree = timeNow() – SESSION.timeStamp

HISTORIQUE.executes.finished = false

HISTORIQUE.executes.Id_mdx_prec = SESSION.preMDX

IX.2.1.3.2 Procédure de collecte 2 :

Le deuxième algorithme de collecte est déclenché quand l'utilisateur ferme son navigateur et donc sa session d'analyse. On enregistre toutes les informations qui concernent la dernière requête MDX exécutée avant cette fermeture de session. Cet algorithme est exécuté seulement si l'utilisateur en question a déjà exécuté une requête dans sa session.

Algorithme IX-2 : Procédure de collecte 2 (lancé lors de fermeture de la session)

```

1 : {
2 :   Entrées :
           SESSION = {currentUser, currentType, lastMDX, preMDX, timeStamp},
           HISTORIQUE = {user, MDX, executes};
3 :   si (SESSION.lastQuery ∉ HISTORIQUE.MDX)
           alors ajouterRequete (SESSION.lastQuery, HISTORIQUE.MDX)
           finsi
4 :   ajouterExecution (SESSION.currentUser, SESSION.lastQuery, SESSION.timeStamp,
           true, SESSION.preMDX, HISTORIQUE.executes)
5 : }
```

IX.2.2 Phase d'apprentissage

Dans cette phase on va utiliser un algorithme personnalisé pour calculer la pertinence des requêtes MDX afin de les classer et choisir les meilleures. Nous allons établir le schéma de l'algorithme dans la figure IX-2 puis l'expliquer ensuite.

Algorithme IX-3 : algorithme d'apprentissage

```

01 : {
02 :   Entrées :
           SESSION = {currentUser, currentType, lastQuery},
           HISTORIQUE = {user, MDX, executes} ;
03 :   Sorties :
           MDXN = {(mdx1, val1), (mdx2, val2), ..., (mdxn, valn)} /* ensemble des requêtes
           MDX et leurs valeurs de pertinence */
04 :   Initialiser MDXN « 1 »
05 :   Si (SESSION.lastQuery <> null) alors
06 :     Pour tous Q∈HISTORIQUE.executes exécutée après SESSION.lastQuery faire
07 :       Si (Q.finished = true) alors MDX(Q) += 30*Q.durée « 2 »
08 :       Sinon      MDX(Q) += 10*Q.durée
           Finsi
           Finpour
           Finsi
09 :   Pour tous Q∈HISTORIQUE.executes exécutée par SESSION.currentUser faire
10 :     Si (Q exécutée cette semaine) alors
11 :       Si (Q.finished = true) alors MDX(Q) += 24*Q.durée
12 :       sinon      MDX(Q) += 8*Q.durée
           Finsi
13 :     Sinon Si (Q.finished = true) alors MDX(Q) += 12*Q.durée
14 :     sinon      MDX(Q) += 4*Q.durée
           Finsi
           Finsi
           Finpour
15 :   Pour tous Q∈HISTORIQUE.executes exécutée par SESSION.currentType faire
16 :     Si (Q exécutée cette semaine) alors

```

```

17 :          Si (Q.finished = true) alors MDX(Q) += 18*Q.durée
18 :                sinon          MDX(Q) += 6*Q.durée
                Finsi
19 :          Sinon Si (Q.finished = true) alors MDX(Q) += 9*Q.durée
20 :                sinon          MDX(Q) += 3*Q.durée
                Finsi
          Finsi
        Finpour
21 :  Pour tous Q∈HISTORIQUE.executes exécutée dans la même semaine des années passées
        faire
22 :          Si (Q.finished = true) alors MDX(Q) += 9*Q.durée
23 :          sinon          MDX(Q) += 3*Q.durée
          Finsi
        Finpour
24 :  Réorganiser les requêtes dans MDXN « 3 »

```

« 1 » : Charger MDX_N avec l'ensemble des requêtes dans HISTORIQUE.MDX qui figurent dans HISTORIQUE.executes avec un utilisateur de type currentType, et initialiser la valeur de pertinence val_i de chaque requête mdx_i à 0.

« 2 » : On ajoute à la valeur de pertinence d'une requête Q dans MDX_N la valeur 30 fois la durée de l'exécution de Q.

« 3 » : Classer les requêtes dans MDX_N selon leurs valeurs de pertinences (la première requête a la valeur de pertinence la plus grande).

IX.2.3. Phase de recommandation

Après l'exécution de l'algorithme de calcul, le système peut maintenant recommander ou prédire le genre de requêtes que l'utilisateur peut préférer en choisissant les requêtes avec les meilleures valeurs de pertinence dans la liste **recommandation**.

IX.3 Conclusion

Dans ce chapitre nous avons conçu le système de recommandation qui va guider les utilisateurs lors de leurs analyses. Nous avons choisi la technique de filtrage hybride. Puis, nous avons conçu l'environnement de collecte d'information. Ces informations vont être exploitées ensuite à l'aide d'un algorithme d'apprentissage, qui va permettre au système de choisir les recommandations les plus pertinentes.

La conception du système de recommandation achevée, on peut commencer à l'implémentation de notre système décisionnel avec ce système de recommandation. Cette implémentation est une des étapes de la réalisation et du déploiement, que nous aborderons dans la partie suivante

PARTIE 4 : REALISATION ET DEPLOIEMENT

La dernière phase du projet est la réalisation de la solution conçue. En utilisant les outils appropriés conformément à notre conception, nous aboutissons à un système décisionnel fonctionnel conforme aux attentes des utilisateurs.

Chapitre **X**

Réalisation et Déploiement de la Solution

X.1 Introduction

La conception de notre système étant achevée, nous passons à la phase de réalisation. Nous commençons par décrire l'environnement de développement ainsi que les outils que nous avons utilisés. Puis nous abordons la réalisation proprement dite avec la construction de l'entrepôt, la construction de la partie ETL, la construction et chargement des cubes, la mise en place de l'outil de restitution, et l'implémentation du système de recommandation.

Nous obtenons à l'issue de cette étape un système fonctionnel et nous donnons un aperçu visuel de ce dernier à travers quelques captures d'écran.

X.2 Environnement de déploiement

Nous avons utilisé plusieurs outils pour le déploiement de notre solution. Pour la création et l'alimentation de l'entrepôt de données et la conception des cubes dimensionnels, nous avons profité de la solution proposée par les technologies Microsoft. Et pour la phase de restitution nous avons opté pour une solution web qui peut être utilisée facilement (sans installation) par de différents périphériques (PC, Téléphones mobiles, Tablette, ...). La totalité de la solution sera déployée sur le cloud Microsoft Azure.

X.2.1 Entrepôt de données

Pour la création de l'entrepôt de données on utilise Microsoft SQL Server Management Studio 2008, pour son alimentation on utilise Microsoft SQL Server Business Intelligence Development Studio 2008 Integration Services, et pour la création des cubes on utilise Microsoft SQL Server Business Intelligence Development Studio 2008 Analyses Services.

X.2.1.1 SQL Server Management studio (SSMS):

SSMS est un environnement intégré d'accès, de configuration, de gestion, d'administration et de développement de tous les composants de SQL Server. SSMS associe un groupe d'outils graphiques à des éditeurs de script performants pour permettre aux développeurs et aux administrateurs de tous niveaux d'avoir accès à SQL Server [43].

Caractéristiques principales de SSMS :

- Possibilité de stocker de gros volumes de données.
- Accès aux données système via des vues, bien plus aisément manipulable que des procédures stockées.
- Un système de droits et mots de passe très souples et sécurisés, qui vérifie aussi les hôtes se connectant. Les mots de passe sont bien protégés, car tous les échanges de mot de passe sont chiffrés, même lors des connexions.

X.2.1.2 SQL Server Integration Services (SSIS) :

SSIS¹ est un composant du logiciel de base de données Microsoft SQL Server qui peut être utilisé pour effectuer un large éventail de tâches de migration de données.

SSIS est une plate-forme pour les applications d'intégration de données et de workflow. Il dispose d'un outil d'entreposage de données rapide et flexible utilisé pour l'extraction de données, transformation et chargement (ETL). L'outil peut également être utilisé pour automatiser la maintenance des bases de données et des mises à jour de SQL Server pour les données du cube multidimensionnel [43].

D'abord publié avec Microsoft SQL Server 2005, SSIS remplacé Data Transformation Services, qui avait été une caractéristique de SQL Server depuis la version 7.0. Contrairement DTS, qui a été inclus dans toutes les versions, SSIS est disponible uniquement dans le "Standard", "Business Intelligence" et éditions "Entreprises".

Cet outil offre plusieurs avantages notamment :

- Un grand nombre de fonctionnalités à travers une gamme très large de composants.
- Une documentation complète, riche et gratuite.
- Compatibilité avec la majorité des SGBD et système d'exploitation.
- Une communauté très réactive et participe dans son développement.

X.2.1.3 SQL Server Analysis Services (SSAS) :

Microsoft SQL Server 2005 Analysis Services (SSAS²) fournit des fonctions OLAP (Online Analytical Processing) et d'exploration de données pour les applications décisionnelles. Analysis Services prend en charge OLAP en permettant de concevoir, de créer et de gérer des structures multidimensionnelles qui contiennent des données agrégées provenant d'autres sources de données, telles que des bases de données relationnelles. Pour les applications d'exploration de données, Analysis Services permet de concevoir, de créer et de visualiser des modèles d'exploration de données créés à partir d'autres sources de données en utilisant un large éventail d'algorithmes d'exploration de données standard [43].

X.2.2 Restitution

Pour la restitution nous avons opté pour une solution web. Cette solution est constituée d'un serveur HTTP IIS, une base de données SQL Server pour les connexions des utilisateurs, et d'une interface créée à partir du Template Gentelella. Pour les interactions avec le serveur (connexion des utilisateurs, lecture des bases des données opérationnelles et décisionnelles, ...) on utilise le langage PHP, et pour la manipulation des cubes OLAP on utilise un API fournit par le toolkit IgniteUI.

X.2.2.1 Serveur Internet Information Services (IIS) :

Internet Information Services (IIS) pour Windows Server est un serveur Web flexible, sécurisée et gérable pour accueillir quoi que ce soit sur le Web. Des médias en streaming aux applications Web. L'architecture évolutive et ouverte d'IIS est prête à gérer les tâches les plus exigeantes.

X.2.2.2 Template Gentelella :

Gentelella est un template d'administration Bootstrap libre à utiliser (open source). Ce template utilise les styles Bootstrap 3 avec une variété de plug-ins et d'outils jQuery pour créer un framework puissant afin de créer des panneaux d'administration ou des tableaux de bord back-end.

Ce template utilise plusieurs bibliothèques pour les graphiques, les calendriers, les formulaires de la validation, les interfaces de style assistant, les menus de navigations off-canvas, le remplissage automatique des formes, barres de progression, les notifications et beaucoup plus.

X.2.2.3 Langage PHP :

PHP est un langage de script utilisé le plus souvent côté serveur : dans cette architecture, le serveur interprète le code PHP des pages web demandées et génère du code (HTML, XHTML, CSS par exemple) et des données (JPEG, GIF, PNG par exemple) pouvant être interprétés et rendues par un navigateur. PHP peut également générer d'autres formats comme le WML, le SVG, le PDF. Il a été conçu pour permettre la création d'applications dynamiques, le plus souvent

développées pour le Web. PHP est le plus souvent couplé à un serveur Apache bien qu'il puisse être installé sur la plupart des serveurs HTTP tel qu'IIS. Ce couplage permet de récupérer des informations issues d'une base de données, d'un système de fichiers (contenu de fichiers et de l'arborescence) ou plus simplement des données envoyées par le navigateur afin d'être interprétées ou stockées pour une utilisation ultérieure.

PHP est libre, gratuit, simple d'utilisation et d'installation, mais nécessite comme tout langage de programmation une bonne compréhension des principales fonctions usuelles ainsi qu'une connaissance aigüe des problèmes de sécurité liés à ce langage [44].

X.2.2.4 IgniteUI toolkit :

IgniteUI est un toolkit HTML et JavaScript complet créé pour réaliser des expériences de navigation modernes sur des différents appareils (PC, tablette ou téléphone), qui assure une haute performance, et une compatibilité avec Asp.net et Bootstrap. Il inclut un ensemble d'APIs et bibliothèque qui permet de naviguer dans des grilles de données, fichiers Excel et Word... Ce toolkit s'appuie sur la bibliothèque JQuery qui est la bibliothèque JavaScript la plus utilisée dans le web ce qui permet de réaliser une interface fluide, interactive et réactive pour l'utilisateur.

IgniteUI est équipé de igOlapXmlaDataSource, un API qui gère la communication entre une application client JavaScript et Microsoft SQL Server Analysis Services (SSAS). Il expose d'une manière amicale pour obtenir des données de Microsoft SQL Server Analysis Services (SASS). L'igOlapXmlaDataSource génère les requêtes MDX nécessaires sur la base des drags and drops réalisés par l'utilisateur. L'igOlapXmlaDataSource est habituellement utilisé avec une ou plusieurs interfaces d'utilisateurs (Ignite widgets) capable de visualiser et d'interagir avec les données OLAP, par exemple igPivotView ou igPivotGrid.

X.2.3 Microsoft Azure

Microsoft Azure est le nom de la plate-forme applicative en nuage de Microsoft. Son nom évoque le concept de « cloud computing » ou informatique en nuage (l'externalisation des ressources informatiques d'une entreprise vers des datacenters distants).

Les outils intégrés, les modèles prédéfinis et les services gérés permettent de créer et de gérer facilement et rapidement des applications d'entreprise, mobiles, web et IoT (objets connectés), et ce en faisant appel aux compétences et technologies existantes.

On a opté pour Microsoft Azure pour profiter des avantages suivants :

- Adapter les capacités de l'hébergement à vos besoins réels.
- Assurer la disponibilité de l'application.
- Vous concentrer sur votre application plutôt que sur l'infrastructure.
- La stabilité et la performance de serveur.

X.3 Réalisation du projet

Après avoir précisé l'ensemble d'outils utilisés dans la phase de réalisation de notre projet. Nous pouvons commencer à implémenter la solution proposée dans ce mémoire.

Dans la première partie, on commence par la construction de l'entrepôt de données, puis nous préparons la zone d'alimentation (ETL), ensuite on crée et charge les cubes dimensionnels. Après on entame la partie d'interface qui inclut la zone de restitution (navigation dans les cubes, reporting, ...). Enfin, on implémente notre système de recommandation en langage PHP.

X.3.1 Construction de l'entrepôt de données

Nous allons créer, en utilisant Microsoft SQL Server Management Studio, une base de données à partir des schémas en étoiles (composés de faits et de dimension) obtenus dans la phase de conception de l'entrepôt de données (Figure VI-10).

Voici une démonstration de la création de la table de fait « FactSales »

```
scripton.sql - DESK...-9QUEUOH/rad (55) * x
CREATE TABLE FactSales(
  ClientID varchar(30) FOREIGN KEY REFERENCES DimClient(ClientID),
  ArticleFinishedID varchar(30) FOREIGN KEY REFERENCES DimArticleFinished(ArticleFinishedID),
  Date date FOREIGN KEY REFERENCES DimTime(Date),
  Quantity float,
  Amount float,
  Temperature float,
  Benefit float,
  PRIMARY KEY (ClientID, ArticleFinishedID, Date)
```

Figure X-1 : requête de la création de la table « FactSales »

X.3.2 Construction de la zone d'alimentation (ETL)

Une fois l'entrepôt de données crée, nous procédons à l'ETL (Extraction, Transformation et Chargement) à l'aide de l'outil Microsoft Business Intelligence Studio 2008 Integration Services.

La première étape consiste à créer les *data flow tasks*, chaque *data flow task* correspond à une dimension ou une table de fait.

Ces *data flow tasks* sont créés pour le remplissage des tables de dimensions et des tables de faits, en s'assurant que le remplissage des tables de dimensions s'exécute avant celui des tables de faits, car les premières référencent les dernières à l'aide des clés étrangères. Comme on peut voir dans le schéma suivant :

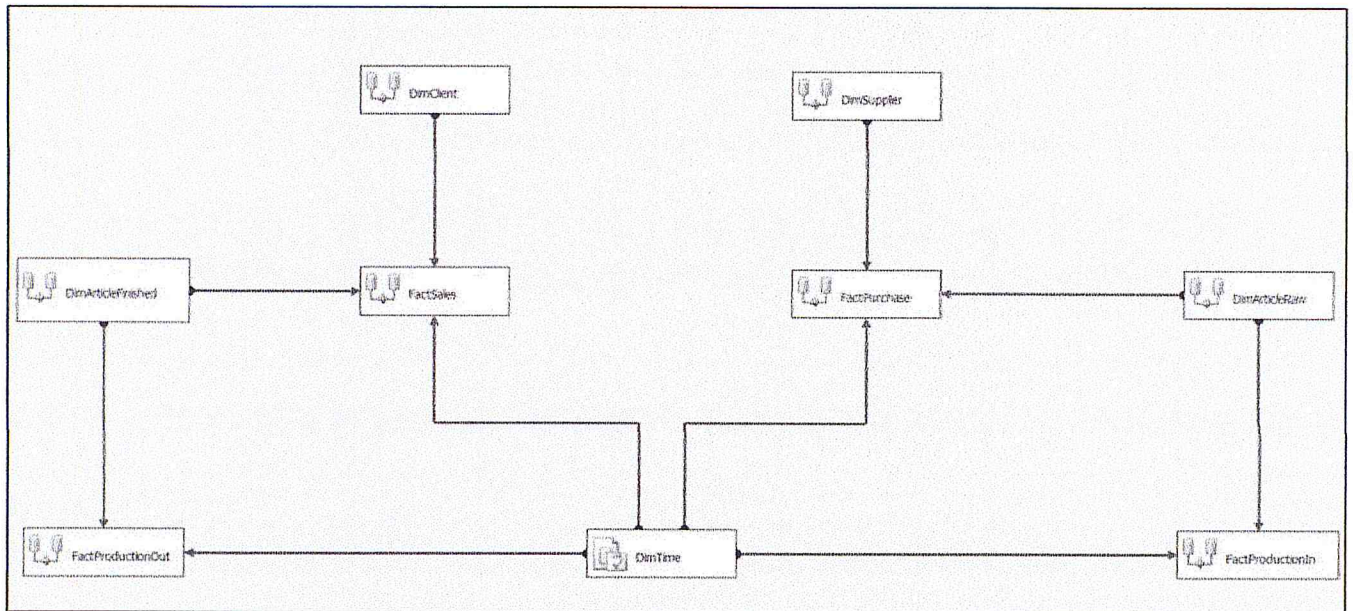


Figure X-2 : schema de remplissage globale

Tout *data flow task* a plusieurs source (ptd2012, ptd2013, ptd2014), nous définissons les connexions aux bases sources ainsi que la base de données de l'entrepôt (paramètres de connexion). Les bases sources seront utilisées dans les composants d'entrée et les tables de l'entrepôt dans les composants de sortie.

Les transformations effectuées sur les données sont généralement :

- des *sorts* pour ordonner les résultats sélectionnés et supprimer les données en double dans certains cas.
- des *lookups* pour mettre à jour les tables.
- des *merges* pour fusionner les différentes sources.
- en plus de *mapping* qui est utilisée pour lier les champs transformés avec les champs des tables résultats (table de fait ou table de dimension)

Voici le schéma de remplissage standard des dimensions et des faits, exemple fait « FactSales »

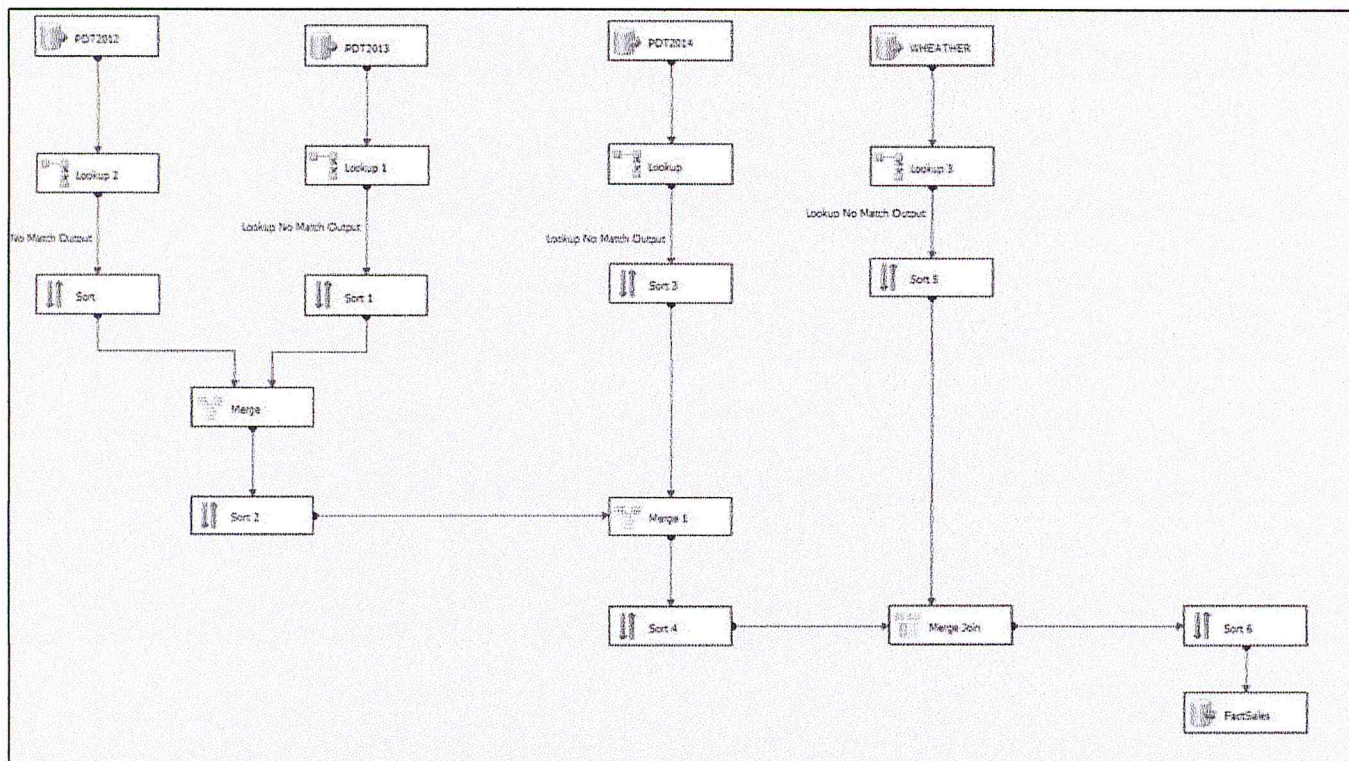


Figure X-3 : remplissage de la table de fait « FactSales »

X.3.3 Construction et chargement des cubes

Pour la construction des cubes on utilise l'outil Microsoft Business Intelligence Studio 2008 Analyses Services. Nous avons suivi les étapes suivantes :

- Création de *Data Source* qui définit la connexion avec l'entrepôt de données.
- Création d'une *Data Source View* qui définit la vue sur l'entrepôt.
- Création de l'ensemble des dimensions, avec les hiérarchies et niveaux correspondant à chacune d'elles.
- Création des cubes à partir des dimensions déjà définies, avec les mesures correspondant à chaque cube.

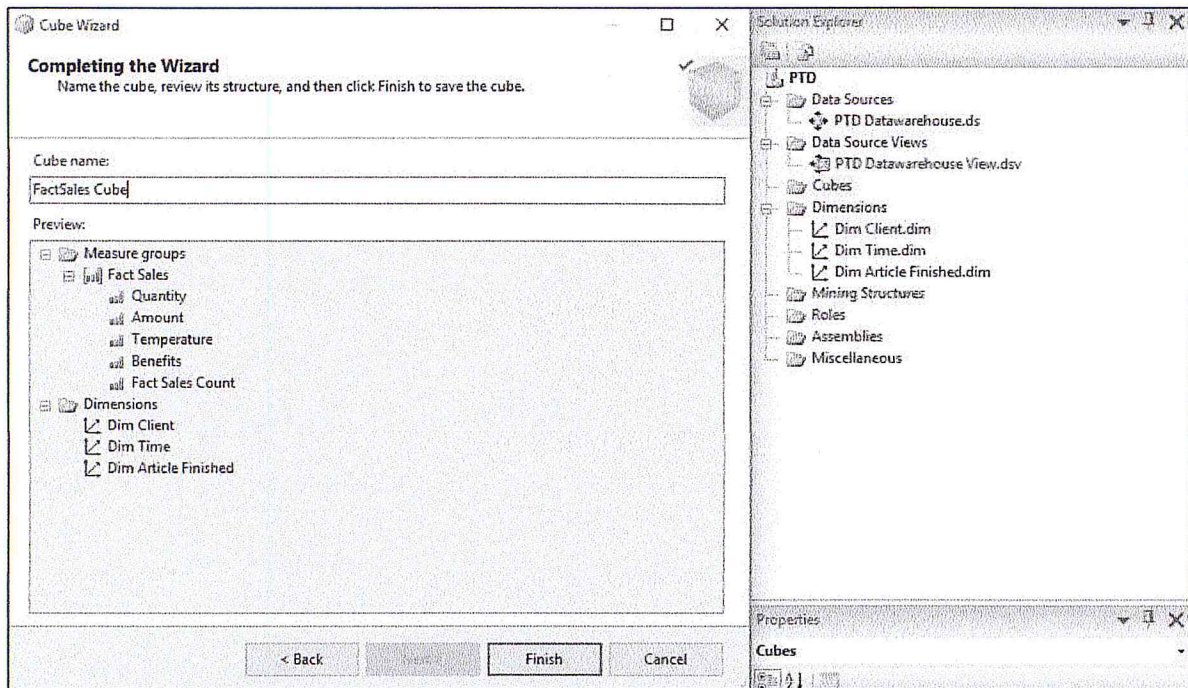


Figure X-4 : construction des cubes

X.3.4 Zone de restitution

La zone de restitution est représentée sous une interface web, elle comprend la visualisation de l'entrepôt de données, des charts prédéfinis (par exemple, les états 80-20), des cubes dimensionnels, et en plus la consultation des bases opérationnelles.

X.3.4.1 Authentification

En premier lieu, les utilisateurs doivent s'authentifier à notre application. Cela se fait à travers une page login présentée dans la figure X-5

The image shows a login form with the following elements:

- Title: Login Form
- Input field: Username
- Input field: Password
- Button: Log in
- Text link: Lost your password?

Figure X-5 : page login

X.3.4.1.1 Méthode de sécurité utilisée :

Pour établir la connexion à l'application, l'utilisateur doit saisir son nom d'utilisateur et son mot de passe. Le système va ensuite vérifier ses informations avec la base données des utilisateurs. Et pour garder l'authentification nous utilisons les sessions de connexion, comme présenté dans la figure X-6

```

44 <?php
45 if(isset($_POST['username']))
46 {
47     require('phpfunc/login_db_connect.php');
48     $_SESSION['logged'] = "";
49     $username = $_POST['username'];
50     $password = md5($_POST['password']);
51     $query = "SELECT * FROM login WHERE username = '$username' AND password = '$password'";
52     $result = sqlsrv_query($db_connection, $query);
53     if(sqlsrv_num_rows($result) == 0)
54     {
55         $_SESSION['logged'] = "no";
56         //header('location: login.html');
57     } else
58     {
59         $_SESSION['logged'] = "yes";
60         $_SESSION['username'] = $username;
61         $_SESSION['password'] = $password;
62         $row = sqlsrv_fetch_array($result);
63         $_SESSION['type'] = $row['type'];
64         header('location: index.html');
65     }
66 }
67 else {
68 }
69 </div class="">

```

Figure X-6 : code source d'authentification

X.3.4.1.2 Droits d'accès :

Il existe plusieurs types d'utilisateur (service) dans l'entreprise, chaque type doit être traité séparément, d'où la nécessité de la création des droits d'accès. Cela permet à chaque type d'avoir une vue adaptée à son domaine. Par exemple, le directeur du service d'achat aura seulement les vues qui concernent les achats. (Le directeur général et les experts analystes peuvent accéder à toutes les vues disponibles dans le système)

Les types d'utilisateurs gérés sont :

- Service vente.
- Service achat.
- Service production.
- Analyste général.

X.3.4.2 Interface de l'application :

Après réussite de l'authentification, l'utilisateur sera orienté vers la page d'accueil.

X.3.4.2.1 Page d'accueil :

Cette page va être adaptée en fonction de son service. On prend par exemple une connexion du directeur de service commercial, comme présenté dans la figure X-7.

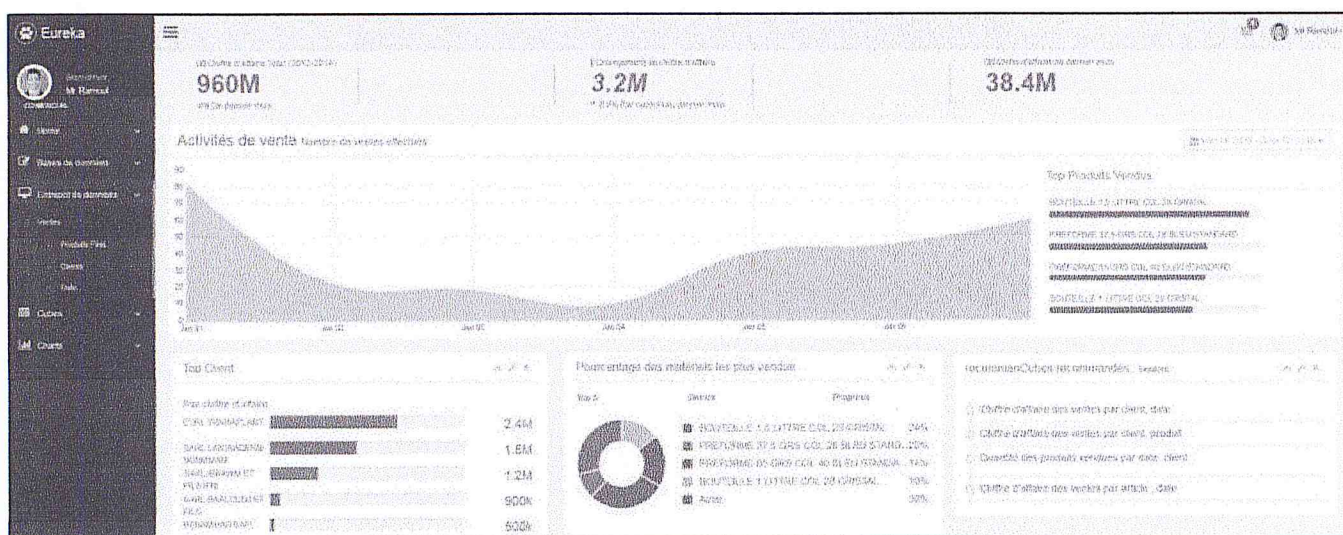


Figure X-7 : la page d'accueil de service commercial

Cette interface est fluide et simple à utiliser, elle donne la possibilité à l'utilisateur de naviguer entre plusieurs aspects (cube, entrepôt, opérationnel, charts), et exploiter les données d'une manière efficace.

X.3.4.2.2 Cubes de données :

L'utilisateur peut accéder au cube relatif à son domaine d'intérêt (par exemple, le service commercial à l'accès au cube des ventes) et faire des opérations de *drag and drop* (pour les dimensions et les mesures) et *slice and dice* (pour les données multidimensionnelles). Pour permettre aux utilisateurs l'exploitation des données OLAP on a utilisé un API de toolkit IgniteUI. L'API en question, est appelé *igOlapXmlDataSource*, va gérer les communications entre l'interface web et le serveur d'analyse SSAS qui héberge les cubes OLAP. Le serveur doit être

configuré avec le fournisseur des données HTTP « msmdpump.dll ». Plus d'informations sur la mise en place du serveur et l'intégration de l'API sont disponibles dans la documentation de toolkit IgniteUI¹.

La figure X-8 présente un cube dimensionnel des chiffres d'affaires par clients et par date.

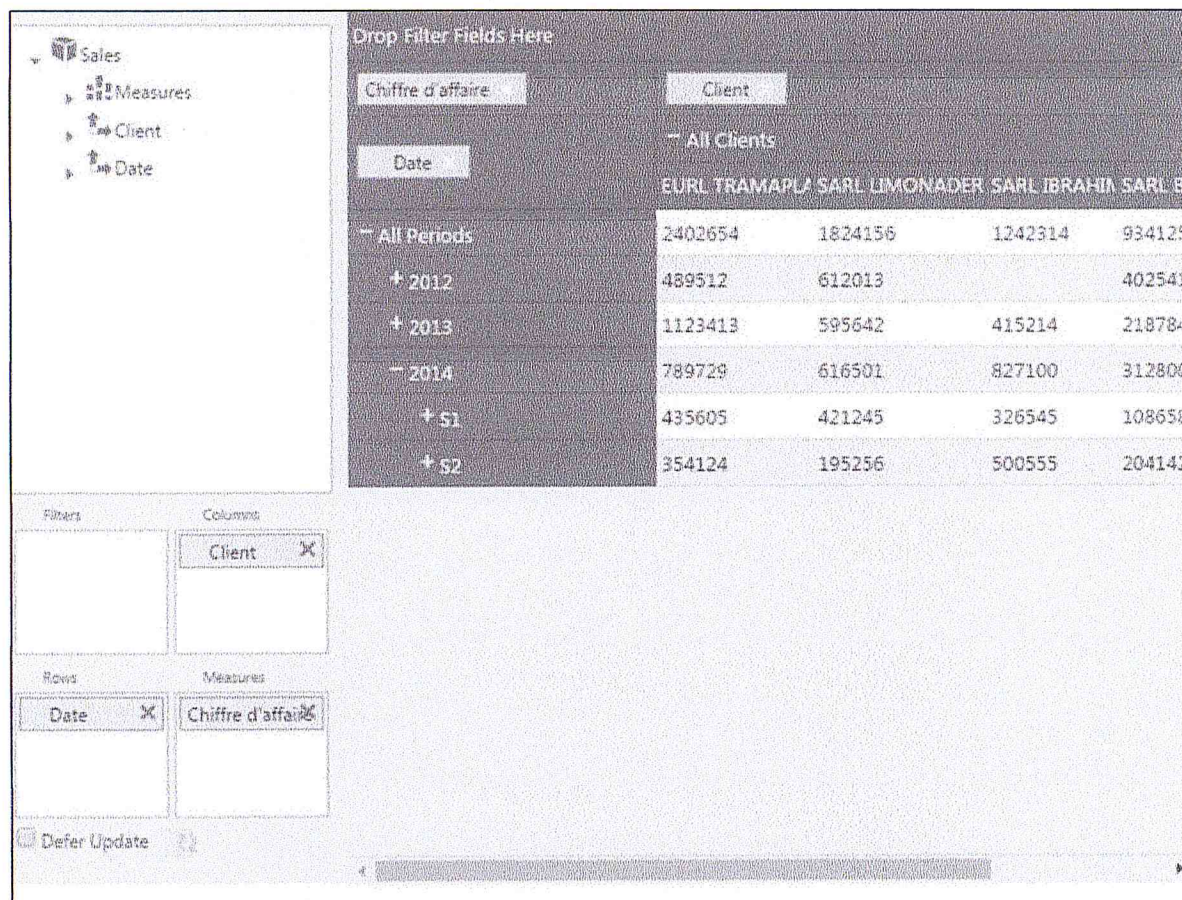


Figure X-8 : interface d'un cube dimensionnel

X.3.4.2.3 Charts et graphes :

L'utilisateur a la possibilité de visualiser les informations ayant des incidences sur la décision d'une manière graphique. Cela lui facilite l'analyse des données numériques où la difficulté de leur interprétation augmente exponentiellement avec leur volume.

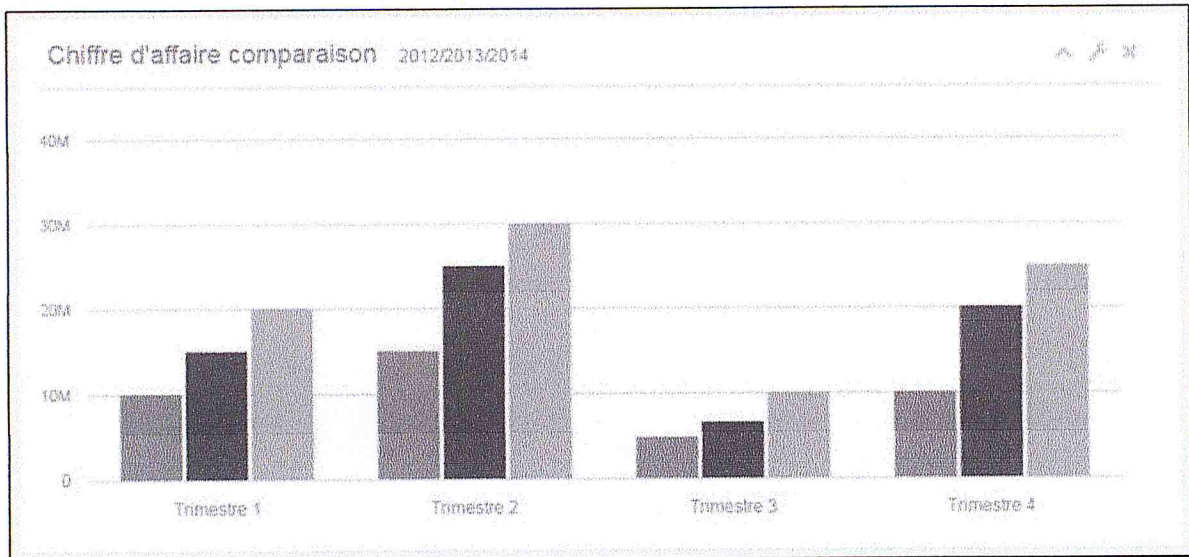


Figure X-9 : Chart de comparaison des chiffres d'affaires des trois années

X.3.5 Système de recommandation

Pour améliorer l'expérience d'analyse des utilisateurs, nous avons ajouté un système de recommandation pour les requêtes MDX. Ce système utilise le filtrage hybride, deux algorithmes pour la collecte d'information et un pour l'apprentissage (Chapitre IX).

Les algorithmes sont implémentés en langage PHP. Voici le code source de l'algorithme de procédure de collecte 1 qui a été présenté dans le chapitre IX.2.1.3.1.

```

1 <?php
2 // #14
3 session_start();
4 $currentUser = $_SESSION['currentUser'];
5 $lastQuery = $_SESSION['lastQuery'];
6 $preQuery = $_SESSION['preQuery'];
7 $timeStamp = $_SESSION['timeStamp'];
8
9 require("historique_db_connect.php");
10
11 $query = "SELECT FROM mdx WHERE query = '$lastQuery'";
12 $result = sqlsrv_query($historique, $query) or die(sqlsrv_error($historique));
13 if (sqlsrv_num_rows($result) == 0)
14 {
15     sqlsrv_query($historique, "INSERT INTO mdx (query) VALUES ('$lastQuery')") or die(sqlsrv_error($historique));
16 }
17
18 sqlsrv_query($historique, "INSERT INTO executes (username, id_mdx, time, durée, finished, id_mdx_prec)
19     VALUES ('$currentUser', '$lastQuery', '$timeStamp', 'NOW() - $timeStamp', false, '$preQuery')");
20
21
22

```

Figure X-10 : Code source de l'algorithme de procédure de collecte 1

Le deuxième algorithme de collecte (Chapitre IX.2.1.3.2) s'exécute quand la session de l'utilisateur est terminée. Voici son code source :

```

1 <?php
2 void session_write_close(void)
3 {
4     session_start();
5     $currentUser = $_SESSION['currentUser'];
6     $lastQuery = $_SESSION['lastQuery'];
7     $preQuery = $_SESSION['preQuery'];
8     $timeStamp = $_SESSION['timeStamp'];
9
10    require("historique_db_connect.php");
11    $query = "SELECT * FROM mdx WHERE query = '$lastQuery'";
12    $result = sqsrv_query($historique, $query) or die(sqsrv_error($historique));
13    if (sqsrv_num_rows($result) == 0)
14    {
15        sqsrv_query($historique, "INSERT INTO mdx (query) VALUES ('$lastQuery')") or die(sqsrv_error($historique));
16    }
17    sqsrv_query($historique, "INSERT INTO executes (username, id_mdx, time, duree, finished, id_mdx_prec)
18    VALUES ('$currentUser', '$lastQuery', '$timeStamp', 'NOW() - $timeStamp', true, '$preQuery')");
19 }
20 ?>

```

Figure X-11 : Code source de l'algorithme de procédure de collecte 2

Le dernier algorithme est celui de l'apprentissage (Chapitre IX.2.2), voici une partie de son code source. La partie d'initialisation de MDX_N :

```

1 <?php
2 $currentType = $_SESSION['currentType'];
3 require("historique_db_connect.php");
4 $query = "SELECT mdx.id as 'id', mdx.query as 'query' FROM mdx, executes, user
5 WHERE mdx.id = executes.id_mdx AND executes.username = user.username AND user.type = $currentType";
6 $result = sqsrv_query($historique, $query) or die(sqsrv_error($historique));
7 $i = 0;
8 while($row = sqsrv_fetch_array($result))
9 {
10     $MDX[$i][0] = $row['id'];
11     $MDX[$i][1] = $row['query'];
12     $MDX[$i][2] = 0;
13     $i++;
14 }
15 ?>

```

Figure X-12 : Code source de l'algorithme d'apprentissage « initialisation MDX_N »

Les recommandations sont affichées de la manière suivante :

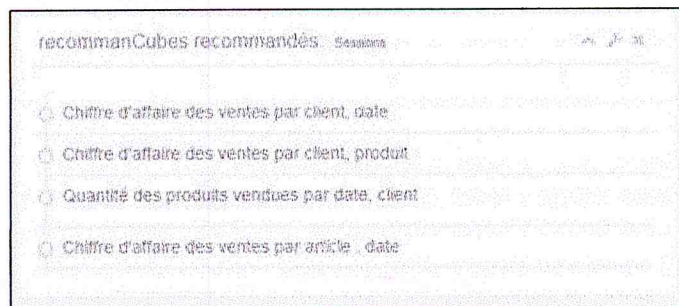


Figure X-13 : Widget des recommandations « exemple service commercial »

X.4 Conclusion

Cette partie a été consacré à l'implémentation de notre solution du système décisionnel accompagné d'un système de recommandation. En premier lieu nous avons établi la présentation des outils utilisés pour le déploiement de la solution. Ensuite nous avons entamé la description des étapes de l'implémentation. Cette dernière consistait de plusieurs étapes : la création de l'entrepôt de données qui va stocker toutes les informations pertinentes à l'analyse souhaitée, la création de la zone d'alimentation avec le processus ETL afin d'alimenter l'entrepôt de données, la construction des cubes dimensionnels sujets d'analyse des décideurs, la création de l'interface web qui va être la zone de restitution de notre système décisionnel, et enfin l'implémentation du système de recommandations.

Nous avons abouti au terme de ce travail à un résultat conforme aux attentes des utilisateurs, et à un système qui reste extensible et évolutif, ce qui permettra aux développeurs d'ajouter facilement de nouveaux axes d'analyses ou de l'adapter à d'autres volets de l'entreprise.

CONCLUSION GENERALE

Conclusion générale

Les entreprises évoluent dans des environnements complexes et compétitifs dans lesquels l'accès à la bonne information est crucial pour leur développement. L'entreprise PTD n'échappe pas à cette règle, c'est pour cela qu'elle décide de mettre en place un système décisionnel qui va la permettre de conforter et d'améliorer sa position dans le domaine de transformation du plastique et d'être mieux préparé face à la concurrence. Parmi les domaines concernés par un système décisionnel sont l'achat, la production et la commercialisation. En effet, les dirigeants ont exprimé un besoin important d'obtenir des informations spécifiques à l'analyse dans ces domaines.

Dans le cadre de notre projet de fin d'études, la société des services informatiques LeaderSoft nous a demandé de concevoir et réaliser un système décisionnel pour les trois volets qui intéressent PTD, les achats, les ventes, et la production.

Ce système décisionnel s'articule autour d'un entrepôt de données alimenté à partir des bases de données sources du logiciel Eureka développé par LeaderSoft. Cela va résulter en plusieurs cubes dimensionnels qui vont être le centre d'analyse des utilisateurs. D'où, pour faciliter la phase d'analyse nous avons opté pour intégrer un système de recommandation qui va accompagner les utilisateurs lors de leur interrogation des cubes.

Ce système décisionnel doit répondre aux attentes des utilisateurs, ce qui fait que de la définition des besoins des utilisateurs est une étape cruciale du projet, une mauvaise définition des besoins rendra le système obsolète.

Nous avons donc procédé, en premier lieu, à une étude du système existant et du domaine métier (le contrôle de gestion au sein de PTD) puis à une collecte d'information auprès des futurs utilisateurs, à travers des questionnaires et notamment des interviews. Nous avons opté, tout au long de cette étape pour la méthode mixte qui considère à la fois les besoins des utilisateurs et les données disponibles. A l'issue de cette étape nous avons défini les besoins des utilisateurs et nous les avons répertoriés en trois volets : suivi des achats, suivi des taux de réalisations de la production et suivi des ventes.

Ensuite nous entamons la conception de la solution, avec la conception de l'entrepôt, à l'aide de la modélisation multidimensionnelle, où nous obtenons des schémas en constellation chaque schéma correspondant à un volet d'analyse. Puis nous abordons la conception de la zone d'alimentation, où nous définissons les détails du processus d'ETL qui assurera que les données

stockées dans l'entrepôt soient fiables et mises à jour. Nous enchaînons avec la conception des cubes dimensionnels, ces cubes offrent une structure adaptée à l'interrogation multidimensionnelle et seront exploités pour l'analyse OLAP. Puis nous nous dirigeons à la conception du système de recommandation qui va aider les utilisateurs et améliorer l'expérience de leurs sessions d'analyse.

Enfin la dernière étape consiste en la réalisation et le déploiement de la solution. L'utilisation de différents outils, nous a permis d'implémenter les étapes sus citées, pour aboutir à un système décisionnel s'articulant autour d'un entrepôt de données pertinentes et fiables, exploitable de manière facile, rapide et intuitive à l'aide d'une interface web qui inclut des cubes, des graphes et des rapports. L'interface intègre un système de recommandation des requêtes multidimensionnelles qui va faciliter et améliorer l'expérience d'analyse.

Au final, les utilisateurs possèdent une solution adaptée à leurs besoins et cela constitue l'objectif que nous nous sommes fixé.

Notre système bien que fonctionnel, il offre ainsi les possibilités suivantes :

- Peut être facilement intégré aux autres entreprises semblables à PTD qui utilise le logiciel Eureka.
- Un des bénéfices de l'API utilisé pour l'OLAP, est qu'il offre la possibilité d'être intégré avec des systèmes distants dans des serveurs différents.
- Aise d'accès avec des différentes périphériques (Ordinateur, mobile, tablette, ...).

A la fin, nous sommes satisfaits avec le thème choisi, car la réalisation de ce projet nous a été bénéfique à tous les niveaux et nous a permis de découvrir le domaine de l'informatique décisionnelle, domaine vaste et en pleine expansion.

Références bibliographique

[1]. Zaraté, P. (2013) *Outils pour la décision coopérative*. Paris : Lavoisier. ISBN 2746295253, 9782746295254

[2]. Fouché, G and L. Langit. 2011. *Foundations of SQL Server 2008 R2 Business Intelligence*. New York City: Apress. ISBN 1430233249, 9781430233244

[3]. Atol. 2015. Présentation des solutions Pentaho. https://www.atolcd.com/uploads/tx_powermail/user_upload/AtolCD-Presentation-Pentaho-2015.pdf

[4]. Ponniah, P. 2001. *Data Warehousing Fundamentals: A Comprehensive Guide for IT Professionals*. The University of Michigan: Wiley. ISBN 0-471-22162-7

[5]. Badard, T and E. Dubé. 2009. *Enabling Geospatial Business Intelligence*. Technology Innovation Management Review. Récupéré Mai 2016 depuis <http://timreview.ca/node/289>

[6]. Kimball, R and J. Caserta. 2011. *The Data Warehouse ETL Toolkit: Practical Techniques for Extracting, Cleaning, Conforming, and Delivering Data*. New Jersey: John Wiley & Sons. ISBN 0-764-57923-1

[7]. Kimball, R. Reeves, L. Ross, M and W Thornthwaite. 1998. *The Data Warehouse Lifecycle Toolkit: Expert Methods for Designing, Developing, and Deploying Data Warehouses*. New Jersey: John Wiley & Sons. ISBN 0471255475, 9780471255475

[8]. Noirault, C. 2006. *Business Intelligence avec Oracle 10g (ETL, Data warehouse, Data mining, rapports...)*. France: Editions ENI. ISBN 2746034298, 9782746034297

[9]. Inmon, W. H. 2002. *Building the Data Warehouse: Third Edition*. New Jersey: John Wiley & Sons. ISBN 0-471-08130-2

[10]. Lecompte, S and T, Boulanger. 2008. *XML par la pratique: bases indispensables, concepts et cas pratiques*. France : Editions ENI. ISBN 978-2-746-4644-3

[11]. Codd, E. F. Codd, S. B and C. T. Salley. 1993. *Providing OLAP to User-Analysts: An IT Mandate*. Computerworld. Récupéré Mai 2016 depuis http://www.minet.uni-jena.de/dbis/lehre/ss2005/sem_dwh/lit/Cod93.pdf

[12]. Fernandez, A. 2002. *La création et la publication de rapport d'activité : Le reporting, qu'est-ce que c'est ?* Récupéré Mai 2016 depuis <http://www.piloter.org/business-intelligence/reporting.htm>

[13]. Ruysen, P. 2010. *Data Mining et Business Intelligence*. Récupéré Mai 2016 depuis <http://dev.af83.com/2010/03/24/data-mining-et-business-intelligence.html>

[14]. Midouni, S. A. D. Darmount, J and F. Bentayeb. 2009. *Approche de modélisation multidimensionnelle des données complexes : Application aux données médicales*. Récupéré Mai 2016 depuis <https://hal.archives-ouvertes.fr/hal-00411237/document>

[15]. Teste, O. 2009. *Modélisation et manipulation des systèmes OLAP : de l'intégration des documents à l'utilisateur*. HDR : Université Paul.

[16]. Negre, E. 2015. *Entrepôts de données : Modélisation multidimensionnelle*. Université Paris-Dauphine. Récupéré Mai 2016 http://www.lamsade.dauphine.fr/~negre/fichiers_joints/BI.pdf

[17]. Gray, J. Brosworth, A. Lyman, A. Reichart, D and H. Pirahesh. 1996. *Data cube: A relational aggregation operator generalizing group-by, cross-tab, and sub-totals*. Dans : New Orleans : s.n. pp. 152–159.

[18]. Teste, O. 2000. *Modélisation et manipulation d'entrepôts de données complexes et historisées*: Université Paul Sabatier - Toulouse III.

[19]. Negre, E. 2014. *Entrepôts de données : Modélisation multidimensionnelle*. Université Paris-Dauphine. Récupéré Mai 2016 <http://www.slideshare.net/ilhemhammouche/bi-40066889>

- [20]. Tournier, R. 2007. *Analyse en ligne (OLAP) de document* : Paul Sabatier - Toulouse III.
- [21]. Jerbi, H. 2012. *Personnalisation d'analyses décisionnelles sur des données multidimensionnelles* [en ligne]. THESE : Université Toulouse 1 Capitole.
- [22]. Laurent, A. 2002. *Bases de données multidimensionnelles floues et leur utilisation pour la fouille de données*. Paris : Paris 6.
- [23]. Ricci, F. Rockach, L. Saphira, B and P. B. Kantor. 2011. *Recommender Systems Handbook*. New York : Springer. ISBN 978-0-387-85820-3
- [24]. Negre, E. 2015. *Systèmes de recommandation*. France : ISTE editions. ISBN (papier) : 978-1-78405-086-3. ISBN (ebook) : 978-1-78406-086-2
- [25]. Melville, P and V. Sindhwani. 2010. *Encyclopedia of Machine Learning, Recommender Systems*. New York : Springer. ISBN : 978-0-387-30768-8
- [26]. Isinkaye, F. O. Folajimi, Y. O and B. A. Ojokoh. *Recommendation systems : Principles, methods and evaluation*. Dans : Egyptian Informatics Journal. Volume 16, Issue 3. 2015. pp. 261-273. doi:10.1016/j.eij.2015.06.005
- [27]. Oard, D. W and J. Kim. *Implicit feedback for recommender systems*. Dans : Proceedings of 5th DELOS workshop on filtering and collaborative filtering. 1998. pp. 31-36. doi:10.1.1.49.3097
- [28]. Buder, J and C. Schwind. *Learning with personalized recommender systems : A psychological view*. Dans : Computers in Human Behavior. Volume 28, Issue 1. 2012. pp. 207-216. doi:10.1016/j.chb.2011.09.002
- [29]. Gadanho, S. C and N. Lhuillier. *Addressing Uncertainty in Implicit Preferences*. Dans : Proceedings of the 2007 ACM conference on Recommender Systems (RecSys '07). 2007. pp. 97-104

- [30]. Jafarkarimi, H. Sim, A. T. H and R. Saadatdoost. *A Naïve Recommendation Model for Large Databases*. Dans : International Journal of Information and Education Technology. Volume 2, Issue 3. 2012. pp. 216-219. DOI:10.7763/IJJET.2012.V2.113
- [31]. Bobadilla, J. Ortega, F. Hernando, A and A. Gutiérrez. *Recommender systems survey*. Dans : Knowledge-Based Systems. Volume 46. 2013. pp. 109-132.
- [32]. Shyong, K. Frankowski, D and Riedl. J. *Do you trust your recommendations ? An exploration of security and privacy issues in recommender systems*. Dans : Emerging trends in information and communication security. 2006. pp. 14-29.
- [33]. Adomavicius, G and A. Tuzhilin. *Toward the next generation of recommender system : A survey of the state-of-the-art and possible extensions*. Dans : IEEE Transactions on Knowledge and Data Engineering. Volume 17, Issue 6. 2005. pp. 734-749. DOI:10.1109/TKDE.2005.99
- [34]. Herlocker, J. L. Konstan, J. A. Terveen, L. G and J. T. Riedl. *Evaluating collaborative filtering recommender systems*. Dans : ACM Transactions on Information Systems. Volume 22, Issue 1. 2004. pp. 5-53
- [35]. Schafer, J. B. Frankowski, D. Herlocker, J and S. Sen. *Collaborative filtering recommender systems*. Dans : The adaptive Web, Methods and Strategies of Web Personalization. 2007. pp. 291-324.
- [36]. Burke, R. *Web recommender systems*. Dans : The adaptive Web, Methods and Strategies of Web Personalization. 2007. pp. 377-408.
- [37]. Park, D. H. Kim, H. K. Choi, I. Y and J. K, Kim. *A literature review and classification of recommender systems research*. Dans : Expert Systems with Applications. 2012. Volume 39, Issue 11. pp. 10059-10072. doi:10.1016/j.eswa.2012.02.038
- [38]. Adomavicius, G and J. Zhang. *Impact of data characteristics on recommender systems performamnce*. Dans : ACM Transactions on Management Information Systems. Volume 3, Issue 1. 2012. DOI:10.1145/2151163.2151166

[39]. Claypool, M et al. *Combining content-based and collaborative filters in an online newspaper*. Dans : Proceedings of ACM SIGIR workshop on recommender systems : algorithms and evaluation. 1991

[40]. Ankunda, R. K. 2011. *The application of the Pareto principle in software engineering*. Récupéré Mai 2016 http://www2.latech.edu/~box/ase/papers2011/Ankunda_termpaper.PDF

[41]. Bunkey, N. *Joseph Juran, 103, Pioneer in Quality Control, Dies*. Dans : New York Times. 2008.

[42]. Kimball, R. 2013. *Kimball Dimensional Modeling Techniques*. Récupéré Mai 2016 depuis <http://www.kimballgroup.com/wp-content/uploads/2013/08/2013.09-Kimball-Dimensional-Modeling-Techniques11.pdf>

[43]. Fouché, G and L. Langit. 2011. *Foundations of SQL Server 2008 R2 Business Intelligence*. New York City: Apress. ISBN 1430233249, 9781430233244

[44]. Achour, M et al. *PHP Manuel*. 2016. Récupéré Juin 2016 depuis <http://php.net/manual/en/>