



MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE
LA RECHERCHE SCIENTIFIQUE
UNIVERSITÉ ABDELHAMID IBN BADIS - MOSTAGANEM

Faculté des Sciences Exactes et de l'Informatique
Département de Mathématiques et d'Informatique
Filière : Informatique

MEMOIRE DE FIN D'ETUDES
Pour l'Obtention du Diplôme de Master en Informatique
Option : **Systemes d'Information Géographique**

THEME :

« Vérification des contraintes d'intégrité dans les entrepôts de données spatiales et SOLAP ». Application : Etude de réseau de distribution de l'eau dans la ville de Relizane

- Etudiantes : **TREICHE RIM IMENE**
SOUCI FATIMA ZOHRA
- Encadrant : **M. BENSALLOUA CHAREF ABDALLAH**

Année Universitaire 2015/2016

Résumé

Les entrepôts de données spatiales (EDS) et les systèmes SOLAP (spatial OLAP) représentent une solution efficace pour stocker, analyser et explorer des données spatiales et non-spatiales dans un objectif d'aide à la décision. Et pour garantir et assurer la cohérence logique de ces données il faut utiliser des règles entre ces données appelées les contraintes d'intégrités.

L'objet du thème est de réaliser une solution géo-décisionnelle basée sur les entrepôts de données spatiales notamment SOLAP, pour la gestion de distribution de l'eau qui est destinée aux décideurs pour l'aide à la décision dans l'ADE de la wilaya de Relizane. Et aussi d'appliquer un mécanisme pour la vérification des contraintes d'intégrité dans ce type d'applications.

Mots clés : Entrepôt de données spatial, SOLAP, contrainte d'intégrité spatiale, réseau de distribution d'eau.

Table de matières

Table de matières	i
Liste d'abréviations	iii
Liste des figures	iv
Liste des tableaux	vi
Dédicace	vii
Dédicace	viii
Remerciement.....	ix
Résumé.....	x
Introduction générale.....	1
Chapitre I : Modélisation multidimensionnelle . Concept de base	3
Introduction	3
I. Les entrepôts de données classiques et OLAP.....	3
I.1. Les classes de données	3
I.2. Alimentation d'un entrepôt de données	4
I.3. L'architecture d'un système basé sur l'entrepôt de données	4
I.4. Concept de base de la modélisation	5
I.5. Les opérations sur les cubes	6
I.6. Schéma conceptuel d'un EDD	8
I.7. OLAP	9
I.7.1. Différents OLAP	9
I.8. Les 12 règles de base	10
II. EDS et SOLAP	10
II.1. Le modèle spatio-multidimensionnel	11
II.2. Implémentation physique d'un serveur SOLAP	12
III. Les outils OLAP Spatial	12
III.1. SIG dominant	13
III.2. OLAP dominant	13
III.3. OLAP-SIG intégrée	13
Conclusion	14
Chapitre II : les contraintes d'intégrités	15
Introduction.....	15
I. Travaux effectués dans le domaine des CI.....	15
II. Les classifications de contraintes d'intégrité d'EDS	16
II.1. Classification orientée implémentation	16
II.1.1. Contraintes ETL	16
II.1.2. Contraintes EDS	16
II.1.3. Contraintes SOLAP	17
II.2. Classification orientée concepts d'EDS	17
II.2.1. Les contraintes intra hyper cube spatial	17

II.2.2.	Les contraintes inter hyper cubes spatiaux	19
II.2.3.	Les CI de versionnement	20
II.3.	Spécification des CI intra-hyper-cube et inter-hyper-cubes	20
III.	Implémentation des contraintes au niveau du tiers EDS.....	22
	Conclusion	23
Chapitre III : Etude de l'existant . Gestion de distribution de l'eau.....		24
	Introduction.....	24
I.	Présentation de la zone d'étude.....	24
I.1.	L'Algérienne Des Eaux (ADE)	24
II.	Ressources en Eau	25
II.1.	Eau De Surface	25
II.1.1.	Les barrages	25
II.1.2.	Les stations de Traitement	26
II.2.	Eau Souterraine	27
	Conclusion	28
Chapitre VI : Modélisation et implémentation de la solution.....		29
	Introduction.....	29
I.	Le modèle décisionnel proposé.....	29
II.	Modélisation	29
II.1.	MagicDraw 18.0	30
III.	Implémentation	31
III.1.	L'organigramme	31
III.2.	Les données utilisées	31
III.3.	Les outils de développements utilisés	32
III.4.	Création de la base de données et la préparation des couches d'information	32
III.4.1.	La création de la base de données	33
III.4.2.	La préparation des couches d'information	38
III.5.	Création du cube de données	39
III.6.	Connexion au serveur OLAP	42
III.7.	Créations de l'application	42
III.7.1.	Connexion à l'interface.....	43
III.7.2.	L'interface utilisateur	44
III.8.	Exploitations de l'application	45
III.8.1.	Les requêtes MDX	45
III.8.2.	L'utilisation de l'interface client	47
III.9.	Vérification des contraintes d'intégrité	49
	Conclusion	50
Conclusion générale		52

Liste d'abréviations

ADE : Algérienne Des Eaux.
API : Interface de Programmation Applicative.
CI : contrainte d'intégrité.
EDD : entrepôt de données.
EDS : entrepôt de données spatiales.
ETL: extract, transform, load.
JDBC: java database connectivity.
JRE : Java Runtime Environment.
HOLAP : Hybrid OLAP.
HTTP: HyperText Transfer Protocol.
IIS : service information internet.
MD: Multidimensionnel.
MDA : architecture dirigée par les modèles.
MDX : Multidimensionnel expression.
MOLAP : Multidimensionnel OLAP.
OCL: Object Constraint Language.
OLAP: on line analytical processing.
OO: Orientée Objet.
ROLAP : Relationnel OLAP.
SDK: Software Development Kit.
SE : système d'exploitation.
SQL : langage de requête structurée (Structured Query Language).
SOAP : Simple object access protocol.
SOLAP : spatiale OLAP.
TCP/IP : Transmission Protocol/Internet Protocol.
UML : langage de modélisation unifié.
USA: United States of America.
XML : langage de balisage extensible (Extensible Markup Language).
XML/A: Xtended Markup Language Analysis.

Liste des figures

Figure 1.1 : L'architecture d'un système basé sur l'EDD.	5
Figure 1.2 : Exemple d'un cube.	6
Figure 1.3 : Exemple d'un schéma en étoile.	8
Figure 1.4 : Exemple d'un schéma en flocon.	8
Figure 1.5 : Exemple d'un schéma en constellation.	9
Figure 1.6 : Présentation d'un exemple des trois types de dimensions spatiales.	11
Figure 2.1 : Types d'implémentation des contraintes d'EDS.	17
Figure 2.2 : Classification orientée concepts d'EDS.	20
Figure 2.3 : L'expression spatial OCL de la contrainte 1.	21
Figure 2.4 : L'expression spatial OCL de la contrainte 2.	21
Figure 2.5 : L'expression spatial OCL de la contrainte 2.	22
Figure 3.1 : L'emplacement de la Wilaya de Relizane.	24
Figure 3.2 : Carte de gestion des communes de la wilaya de Relizane.	25
Figure 3.3 : Carte des ouvrages hydrauliques de la wilaya de Relizane.	28
Figure 4.1 : Le modèle décisionnel proposé.	29
Figure 4.2 : Diagramme de classe de l'entrepôt de donnée spatiale.	30
Figure 4.3 : Organigramme de solution.	31
Figure 4.4 : Fenêtre de SQL Server management studio.	33
Figure 4.5 : Création de la base de données.	33
Figure 4.6 : Fenêtre de création de la base de données.	34
Figure 4.7 : Création de la table.	35
Figure 4.8 : Création des colonnes de la table.	35
Figure 4.9 : Exemple d'une table.	36
Figure 4.10 : La création de schéma de la base de données.	37
Figure 4.11 : Création d'un schéma de la base de données.	37
Figure 4.12 : le géoréférencement.	38
Figure 4.13 : la numérisation de la carte.	38
Figure 4.14 : La création de la geodatabase.	39
Figure 4.15 : Model Multidimensionnel de la base de données.	39
Figure 4.16 : Hiérarchie temps.	40
Figure 4.17 : le schéma du cube de données.	40
Figure 4.18 : le déploiement du cube.	41
Figure 4.19 : le traitement du cube.	41
Figure 4.20 : Module de gestion de connexions.	44
Figure 4.21 : L'interface utilisateur finale.	45
Figure 4.22 : Résultat de la première requête.	46
Figure 4.23 : Résultat de la deuxième requête.	46
Figure 4.24 : Acquisition des requêtes MDX.	47
Figure 4.25 : Exemple d'une requête MDX.	47
Figure 4.26 : Résultat de la requête précédente.	48

Figure 4.27 : Résultat de la même requête.....	48
Figure 4.28 : Résultat de la même requête.....	49
Figure 4.29 : Requête OCL traduite en SQL.....	49

Liste des tableaux

Tableau 1.1 : Les différents opérateurs OLAP.....	7
Tableau 3.1 : les stations de traitement.....	26
Tableau 3.2 : les communes gérées par l'eau de surface.	26
Tableau 3.3 : les communes gérées par l'eau souterraine.	27

Introduction générale

Contexte

L'informatique décisionnelle connue sous le terme Business Intelligence (BI) apporte des solutions nouvelles pour la modélisation, l'interrogation et la visualisation de données dans un objectif d'aide à la décision à l'usage des décideurs et des dirigeants d'entreprises. Dans le cas classique l'informatique décisionnelle considère l'entrepôt de données et OLAP comme une solution effective pour la prise de décision.

Un système d'information géographique (SIG) est un outil d'informatique désigné pour recueillir, stocker, traiter, analyser, gérer et présenter l'information géographique qui est la représentation d'un objet ou d'un phénomène réel ou imaginaire dans le présent, le passé ou le futur localisé dans l'espace à un moment donnée .

Un SIG contient des données spatiales stockées soit sous format vectoriel (gère les points, les lignes et les polygones), soit sous format raster (stockées sous formes de cellules formant une maille) et les deux formats sont complétés par des données alphanumériques et cela nécessite d'analyser d'importants volumes de données géographiques. Les systèmes OLAP, n'offrent aucun instrument pour la gestion des données spatiales donc le couplage de systèmes OLAP et de Systèmes d'Informations Géographiques est un besoin important ce qui donne naissance au concept de Spatial Olap (SOLAP).

Problématique

L'entrepôt de données et les systèmes OLAP peuvent stocker, traiter, analyser et gérer les données non géographiques ce qui nécessite le couplage entre le SIG et les systèmes OLAP pour prendre en compte l'information géographique et leur volume de stockage.

Néanmoins, après le stockage et le chargement de ces données il faut respecter des règles et éviter des situations non désirées; ce qui nécessite aussi d'implémenter des contraintes d'intégrité pour assurer celle-ci. La consistance des manipulations et des résultats.

Contribution

Notre contribution par la présente étude peut se résumer, après avoir effectué un stage pratique dans la direction algérienne des eaux de la wilaya de Relizane, de chercher à développer un système d'aide à la décision dédié aux problématiques liées à quelques activités de l'entreprise ADE-Relizane relatives aux activités de production et de distribution d'eau dans zone concernée, en s'appuyant sur la nouvelle technologie géo-décisionnelle SOLAP.

Ce mémoire est devisé en quatre chapitres, le premier chapitre résume les concepts de bases des entrepôts de données et de l'analyse multidimensionnelle, ces entrepôts sont modélisés selon le modèle spatio-multidimensionnel qui définit les concepts de mesure spatiale

et de dimension spatiale pour prendre en compte la composante spatiale de l'information géographique.

Le second chapitre présente les travaux de recherche, ces recherches portent sur la modélisation des contraintes d'intégrité dans les Systèmes d'Information.

Dans le troisième chapitre nous allons présenter la zone d'étude qui est le réseau couvert par « la direction algérienne des eaux de la wilaya de Relizane (ADE) » et toutes les données concernant notre stage.

Le quatrième chapitre est consacré à décrire les différentes phases de modélisation et de développement de l'application décisionnelle (SOLAP) pour l'ADE de la ville de Relizane.

Enfin, nous allons terminer par une conclusion générale.

Introduction

Une base de données relationnelle est un conteneur permettant de stocker de petites quantités de données en respectant de la 3^{ème} forme normale. Lorsque la quantité des données est importante et les données sont hétérogènes, la gestion de la base sera plus difficile.

Alors comme une solution nous proposons l'entrepôt de données(EDD). Dans ce chapitre nous allons voir les concepts de base des EDD et les types d'applications utilisées dans ce cas.

I. Les entrepôts de données classiques et OLAP

L'entrepôt de données (ou Datawarehouse en anglais) est le lieu de collection, et de stockage de données orientées sujet, intégrées, historisées, et non volatiles utilisées pour le support d'un processus d'aide à la décision [22]. L'entrepôt de données doit respecter de fortes contraintes de sécurité (sauvegarde, intégrité des données, protection contre l'intrusion) et être accessible par les différentes applications et utilisateurs. [23]

Les caractéristiques des données d'EDD sont :

✓ **Données orientées sujet**

Les données sont organisées par sujet et structurées par thèmes. L'avantage de cette représentation est d'éviter aux données concernées par plusieurs sujets d'être dupliquées.

✓ **Données intégrées**

Les données proviennent de diverses sources. Elles sont donc hétérogènes, donc avant d'être intégrées il est nécessaire d'effectuer une forte normalisation pour uniformiser leur format.

✓ **Données historisées**

Les données ne sont jamais mises à jour. Chaque nouvelle donnée est insérée. Un référentiel temps doit être associé aux données afin de permettre la consultation des historiques de changements au cours du temps.

✓ **Données non volatiles**

Un entrepôt de données doit conserver la traçabilité des informations et des décisions prises. Les données ne sont jamais modifiées ou supprimées. [22]

I.1. Les classes de données

Les données d'un ED se structurent selon 2 axes synthétique et historique. L'aspect historique nécessite que chaque nouvelle insertion de données ne détruise pas les anciennes valeurs, mais crée une nouvelle occurrence de la donnée.

L'axe synthétique contient :

- 1) **Les métadonnées** : c'est l'ensemble des données qui décrivent d'autres données (données sur les données).
- 2) **Les données détaillées** : représentent les événements les plus récents (données intégrées dans l'ED).
- 3) **Données agrégées** : Somme de données détaillées. Doivent être facilement accessibles et compréhensibles. [7]

I.2. Alimentation d'un entrepôt de données

L'alimentation est la procédure qui permet de transférer des données du système opérationnel vers l'entrepôt de données en les adaptant. Comment alimenter, donc, l'Entrepôt de Donnée ?

Les processus ETL (Extraction, Transformation et Chargement) est le plus important pour l'alimentation d'un entrepôt de données dont le processus se résume comme suit:

- 1) **Extraction des données** : Cette opération concerne les bases de données opérationnelles, elle sert à extraire les données nécessaires.
- 2) **Transformation** : Cette étape applique un ensemble de règles pour rendre les données cibles homogènes et puissent être traitées de façon cohérente.
- 3) **Chargement** : sert à charger les données nettoyées (élimination des doublés, incomplètes, etc.) dans l'ED .C'est une opération qui risque d'être assez longue et la moins complexe. [16]

I.3. L'architecture d'un système basé sur l'entrepôt de données

L'architecture d'un système basé sur l'EDD est composé de : source de données, serveur EDD, serveur OLAP, client OLAP(GUI). (Voir figure 1.1)

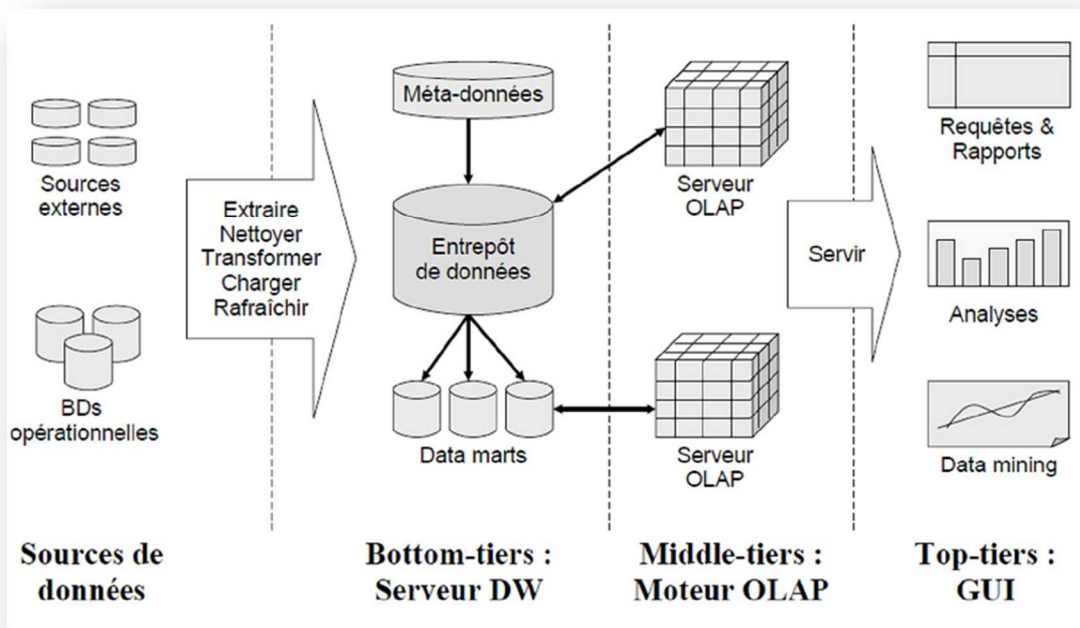


Figure 1.1 : L'architecture d'un système basé sur l'EDD.

1) Sources de données

La couche de source de données contient différentes sources d'information avec différentes structures. C'est à dire des données en général hétérogènes.

2) Serveur DW

Il existe deux approches pour stocker les données dans l'entrepôt de données :

- Normalisation : rendre les données sous la 3^{ème} forme normale.
- Dimensionnelle : par les schémas (étoile, flocon, constellation). [24]

3) Les magasins de données (Data Marts)

Un sous-ensemble de l'entrepôt global concernant un groupe spécifique d'utilisateurs. [25]

4) Moteur d'analyse en ligne (OLAP server)

Moteur de manipulation de données spécifiquement conçu pour soutenir et fonctionner sur des structures de données multidimensionnelles.

5) GUI (Interface utilisateur)

Outils d'aide à la décision (analyse, requêtes, rapports, fouille de données, ... [24]

I.4. Concept de base de la modélisation

Le modèle Multidimensionnel (MD) représente les données décisionnelles dans un espace à n dimensions, communément appelé hyper cube ou cube de données, il se compose de faits contenant les mesures à analyser, de dimensions et les niveaux de hiérarchie.

- 1) **Un cube:** Un cube est un ensemble de données organisées selon des dimensions. On appelle mesure la valeur contenue dans une cellule du cube, associée aux valeurs prises sur les dimensions composant le cube. (Voir figure 1.2)

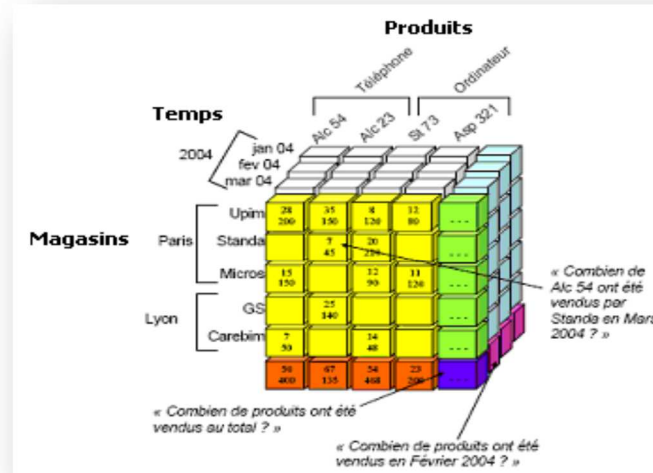


Figure 1.2 : Exemple d'un cube.

- 2) **Dimension:** est un axe d'analyse c'est-à-dire une base sur laquelle seront analysées les données.
- 3) **Fait:** représente la valeur d'une mesure selon un membre de chacune des dimensions.
- 4) **Mesure:** résultat d'une opération d'agrégation des données.
- 5) **Niveau de hiérarchie :** un niveau de hiérarchie se définit au niveau des tables de dimensions. Cela permet d'agréger les données. [11]

I.5. Les opérations sur les cubes

Les outils OLAP utilisent des opérateurs particuliers afin de « naviguer » dans les cubes multidimensionnels :

- 1) **Slice:** consiste une sélection sur une dimension du cube (Consiste à ne travailler que sur une tranche de l'hypercube.)
- 2) **Dice (scoping) :** définition d'un sous cube par sélection sur deux ou plusieurs dimensions. On s'intéressera alors seulement à une partie des données.
- 3) **Pivot :** permet une présentation alternative du cube, transformation en une série de plans 2D Reversement du cube sur un ou plusieurs axes pour une vision alternative.

4) **Roll up/Drill down** : Cela permet d'aller vers les informations détaillées dans une hiérarchie ou au contraire de remonter d'un niveau de granularité. Il s'agit donc de « zoomer ou de dézoomer » sur une dimension. [15]

Le nom	L'opération
Slice	
Dice	
Pivot	
Roll up / drill down	

Table 1.1. Les différents opérateurs OLAP.

I.6. Schéma conceptuel d'un EDD

- 1) **Modèle en étoile** : Le centre est la table des faits, et les branches sont les dimensions. (Voir figure 1.3)

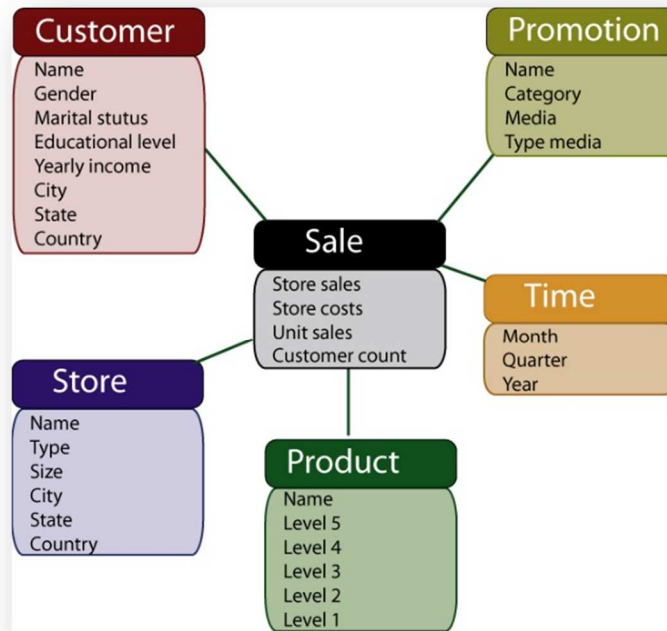


Figure 1.3 : Exemple d'un schéma en étoile.

- 2) **Modèle en flocon** : Le principe est le même que pour le modèle en étoile, mais en plus les dimensions sont décomposées. (Voir figure 1.4) [17]

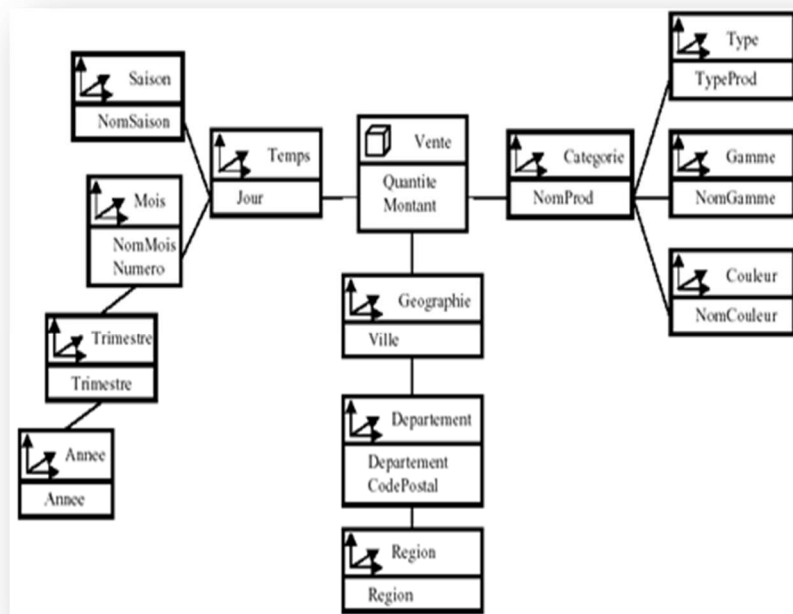


Figure 1.4 : Exemple d'un schéma en flocon.

- 3) **Modèle en constellation** : Il est encore basé sur le modèle en étoile. Mais on rassemble plusieurs tables des faits qui utilisent sur les mêmes dimensions. (Voir figure 1.5) [26]

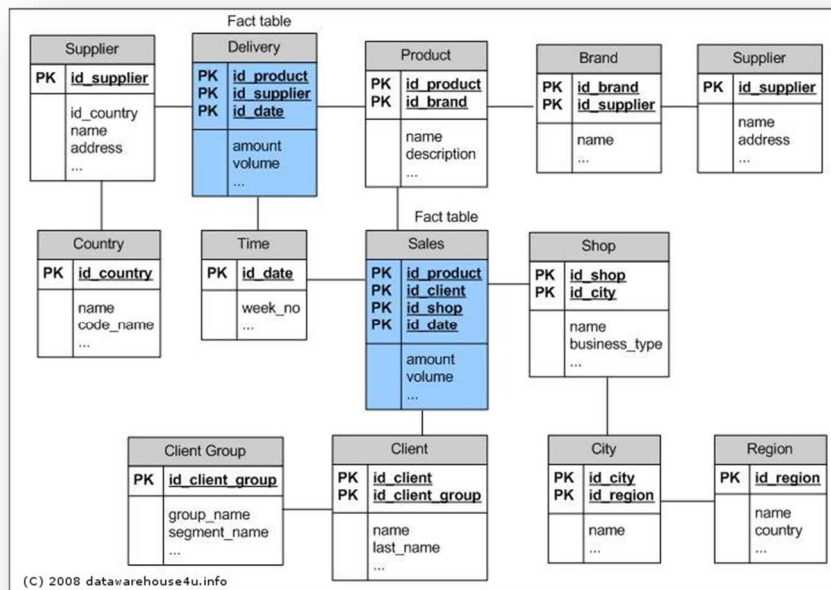


Figure 1.5 : Exemple d'un schéma en constellation.

I.7. OLAP

Les entrepôts de données reposent sur le système OLAP (On Line Analytical Processing). Ce système travaille en lecture seulement. Les programmes consultent d'importantes quantités de données pour procéder à des analyses. Les objectifs principaux sont regrouper, organiser des informations provenant de sources diverses, les intégrer et les stocker pour donner à l'utilisateur une vue orientée métier, retrouver et analyser l'information facilement et rapidement. Cela nécessite de consulter des versions historiques de la base et peut se permettre d'ignorer temporairement les dernières mises à jour. [27]

I.7.1. Différents OLAP

La modélisation de solution OLAP s'effectue en différentes techniques :

- 1) **ROLAP** : Dans le Relationnel OLAP les données sont stockées dans une base de données relationnelle. Un moteur OLAP permet de simuler le fonctionnement d'un hyper cube. Cela permet une facilité dans la mise à jour des données.
- 2) **MOLAP** : Le Multidimensionnel OLAP consiste à utiliser un système multidimensionnel pur, qui gère des structures multidimensionnelles natives. Elles utilisent des tableaux à n dimensions. L'accès aux données se fait directement dans le cube. Cela permet une rapidité d'accès à l'information mais augmente le temps de mise à jour.
- 3) **HOLAP** : (Hybrid OLAP) est un hybride entre ROLAP ET MOLAP. Les parties tables de faits et tables de dimensions sont stockées dans une base

relationnelle standard tandis que le reste des données (les calculs) sont stockées dans une base multidimensionnelle. [28]

I.8. Les 12 règles de base

On a 12 règles de base permettant de qualifier le concept global nommé OLAP (On Line Analytical Processing).

- 1) **Transparence** : Le système doit être transparent pour l'utilisateur qui doit pouvoir accéder à la base par l'intermédiaire d'outils standards tels que des tableurs ou des formulaires HTML.
- 2) **Accessibilité** : Les sources multiples servant à alimenter l'entrepôt doivent être facilement accessibles à travers la structure logique de l'entrepôt.
- 3) **Manipulation des données** : La navigation doit pouvoir s'effectuer simplement et intuitivement à travers des interfaces graphiques à forte ergonomie.
- 4) **Souplesse d'affichage** : Le système doit pouvoir retourner les résultats des requêtes sous une forme graphique. Ces résultats doivent pouvoir être manipulés et servir de base à d'autres requêtes.
- 5) **Multidimensionalité** : Le système doit permettre une manipulation multidimensionnelle de l'ensemble des données de l'entrepôt.
- 6) **Dimensionnalité générique** : Toutes les dimensions d'un hypercube doivent être accessibles à travers un même protocole.
- 7) **Client/serveur** : L'approche doit respecter l'architecture client-serveur afin que plusieurs utilisateurs puissent manipuler l'entrepôt sans difficulté.
- 8) **Multi-utilisateur** : Des requêtes sur l'entrepôt de données doivent pouvoir être réalisées en simultanées par plusieurs utilisateurs.
- 9) **Accès stable** : Le nombre de dimensions et/ou le nombre de niveaux d'agrégation doivent pouvoir changer sans perturber le fonctionnement de l'entrepôt.
- 10) **Gestion des matrices creuses** : Les cellules " vides " d'un hypercube doivent être gérées de façon efficace afin de limiter les capacités de stockage nécessaire et les temps d'accès aux données de l'hypercube.
- 11) **Croisement des dimensions** : Les dimensions dans les différents entrepôts métiers (datamarts) doivent être accessibles et croisées. Toutes les règles de gestion doivent pouvoir s'appliquer à l'ensemble des tranches des hyper-cubes correspondant.
- 12) **Nombre illimité de dimension et de niveaux d'agrégation** : Il ne doit pas y avoir de limites imposées au nombre de dimension et de niveaux d'agrégation manipulable. [9]

II. EDS et SOLAP

Un Entrepôt de Données Spatiales (EDS) est une collection de données spatiales et non spatiales : orientées-sujet, intégrées, variantes dans le temps, et non volatiles dédiées à la prise de décision spatiale [20]. Un EDS est une reformulation d'un ED conventionnel à la seule différence d'intégrer des données de nature spatiale. [10]

Le concept SOLAP s'appuie sur l'ajout de la dimension spatiale au concept de l'OLAP avec adaptation des opérations d'analyses, enrichissement de l'interface et affichage cartographique.

Il est défini par Yvan Bédard comme Une plate-forme visuelle supportant l'exploration et l'analyse spatio-temporelle faciles et rapides des données selon une approche multidimensionnelle à plusieurs niveaux d'agrégation via un affichage cartographique, tabulaire ou en diagramme statistique. [20]

II.1. Le modèle spatio-multidimensionnel

Les systèmes d'EDS et SOLAP se basent sur le modèle spatio-multidimensionnel. Ce modèle étend le modèle multidimensionnel des systèmes d'ED et de l'OLAP avec de nouveaux concepts spatio-multidimensionnels.

Ces nouveaux concepts concernent la reformulation des concepts classiques de dimensions et de mesures pour prendre en compte la composante spatiale.

1) Les Dimensions spatiales

La dimension spatiale est une dimension qui contient au moins un niveau d'agrégation spatial et Permet la représentation de l'information spatiale relative aux localisations géographiques et non géographiques des faits en axes d'analyse.

Il existe 3 types de dimensions spatiales : (Voir figure 1.6)

- Descriptive : Les données n'ont que la valeur nominale de la référence spatiale (nom du pays, nom de l'état, nom de la ville).
- Géométrique : chaque élément de la dimension est associé à un objet géométrique.
- Mixte : qui contiennent en même temps des données géométriques et non géométriques.

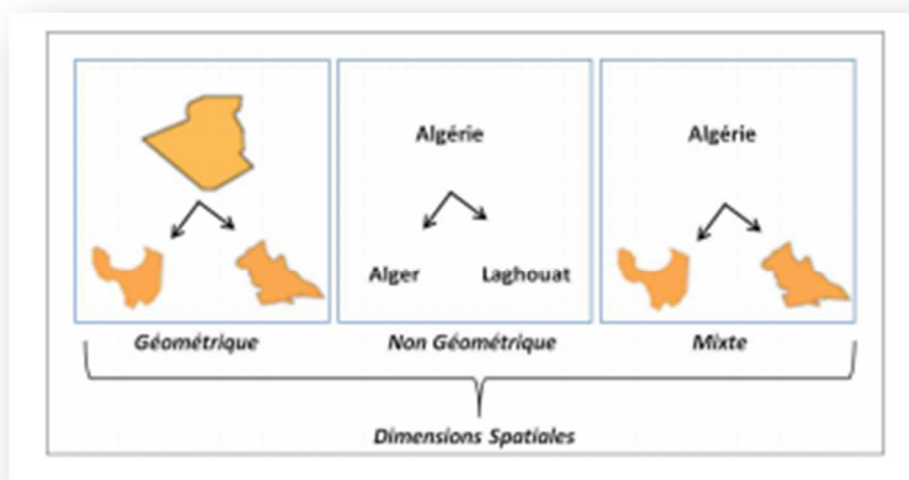


Figure 1.6 : Présentation d'un exemple des trois types de dimensions spatiales.

2) Les Mesures spatiales

Une mesure spatiale est une mesure dont le type de données est spatial. Sa représentation cartographique représente un intérêt pour l'analyse décisionnelle et représente l'information spatiale relative aux localisations géographiques et non géographiques des faits dans les cellules de l'hypercube.

Il existe 2 types de mesures :

- Descriptive numérique.
- Spatiale :
 - Ensemble de coordonnées.
 - Résultat d'une opération de mesure spatiale ou topologique.
 - Ensemble de pointeurs vers une structure géométrique externe.

3) Les Faits spatiaux

Un fait spatial est défini comme étant un fait géoréférencé à une position qui est pertinente à analyser; sa localisation est représentée par une mesure spatiale. Un fait est dit spatial s'il contient au moins une mesure spatiale ou un niveau d'agrégation spatial. [8]

Les opérateurs SOLAP sont les mêmes opérateurs OLAP : slice, Dice, drill down, roll up, pivot. [29]

II.2. Implémentation physique d'un serveur SOLAP

Les approches principales pour l'implémentation de serveurs SOLAP sont MOLAP, HOLAP, ROLAP :

- 1) **MOLAP** : extraient les données de l'entrepôt de données Spatiale et les mémorisent en utilisant des structures de données particulières et ils appliquent des techniques d'indexation et de hachage pour localiser les données lors de l'exécution des requêtes multidimensionnelles.
- 2) **ROLAP** : utilisent la technologie des bases de données relationnelles pour mémoriser les données. Pour obtenir de performances acceptables, ces systèmes utilisent des structures d'indexation particulières comme l'index bitmap, et les vues matérialisées.
- 3) **HOLAP** : sont plus rapides et nécessitent moins d'espace de stockage, car les données sont stockées par le serveur SOLAP de façon à garantir ces aspects. Les serveurs HOLAP permettent le passage à l'échelle et sont plus adaptés pour des mises à jour très fréquentes. [29]

III. Les outils OLAP Spatial

Un outil SOLAP repose sur l'intégration des fonctionnalités SIG et OLAP. La composante cartographique est utilisée pour visualiser les membres de dimensions et/ou les mesures avec une composante spatiale, pour représenter les mesures alphanumériques, grâce à des cartes thématiques, et pour accéder aux opérations de navigation multidimensionnelle.

Différents systèmes SOLAP, qui peuvent être classifiés en trois différentes typologies, ont été développés.

Les solutions SOLAP sont regroupées en trois grandes classes: Olap dominant, SIG dominant et OLAP-SIG intégrée.

III.1. SIG dominant

Dans les solutions SIG dominantes, le serveur OLAP est simulé grâce à une base de données relationnelle modélisée sous forme d'étoile. Les solutions SIG dominantes, offrent toutes les fonctionnalités d'un outil SIG : stockage, analyse et visualisation des données spatiales. Par contre, elles doivent inclure, dans la base de données, des éléments permettant d'implémenter les opérations OLAP de forage et de coupe, puisqu'il n'existe pas de serveur OLAP pour gérer ces opérations. De plus, toutes les fonctionnalités avancées OLAP comme l'utilisation de mesures dérivées, ne sont pas présentes dans ce type d'outil, ce qui limite ses capacités d'analyse multidimensionnelles.

III.2. OLAP dominant

Les outils OLAP dominants utilisent un système OLAP et offrent toutes les fonctionnalités classiques pour l'analyse multidimensionnelle. Par contre, les fonctionnalités SIG sont limitées à une simple représentation cartographique des mesures et des dimensions spatiales, à la navigation cartographique et à la sélection d'objets géographiques. Ces solutions ne présentent aucun instrument pour l'analyse spatiale ou d'autres fonctionnalités avancées SIG. Ceux-ci, comme montré par plusieurs travaux sont nécessaires et complémentaires à l'analyse spatio-multidimensionnelle. De plus, dans ces solutions les opérateurs de forage sur la dimension spatiale sont inexistantes ou limités. Il n'est pas possible par exemple, d'obtenir une carte avec des membres spatiaux de différents niveaux, grâce à une simple opération de forage sur un seul membre spatial. Par conséquent, dans les solutions OLAP dominant, l'information géographique n'est pas exploitée, ce qui implique que ce type de solution présente quelques limitations pour une analyse spatio-multidimensionnelle concrète. L'utilisateur ne peut pas comparer visuellement les mesures associées aux membres de différents niveaux spatiaux, ou encore il n'a pas accès aux outils SIG pour analyser les données géographiques.

III.3. OLAP-SIG intégrée

Les solutions OLAP-SIG intégrées comme JMap fusionnent toutes les fonctionnalités des deux différents systèmes dans un seul environnement. Les fonctionnalités SIG d'analyse et de visualisation sont nécessaires pour l'analyse spatio-multidimensionnelle et elles complètent les fonctionnalités purement OLAP. Les solutions OLAP-SIG intégrées sont alors les plus adaptées pour une analyse spatio-multidimensionnelle réelle et efficace. Cette intégration peut être vue comme une reformulation des trois niveaux d'une architecture OLAP classique, en utilisant et/ou en ajoutant des fonctionnalités SIG. Le premier niveau est un entrepôt de données spatiales, qui doit permettre de modéliser les complexes structures de données associées aux dimensions et aux mesures spatiales. Le deuxième niveau est un serveur OLAP capable de gérer des requêtes spatio-multidimensionnelles.

Dans le meilleur des cas, il devrait :

- gérer les hiérarchies complexes et traiter symétriquement mesures et dimensions.
- permettre la définition des mesures spatiales comme objets spatiaux et résultats d'opérations spatiales, et de mesures dérivées à partir des membres et/ou mesures classiques et/ou spatiales.
- considérer les relations spatiales entre les membres de dimensions et éviter de prendre en compte plusieurs fois la même mesure pendant le processus d'agrégation.
- prendre en compte la composante spatiale des données géographiques lors d'une opération de forage ou de coupe.
- fournir un mécanisme pour calculer les vues matérialisées. [19]

Conclusion

Dans ce chapitre, nous avons présenté les concepts de base concernant l'EDD. Dans le deuxième chapitre on va présenter les CI (contrainte d'intégrité) dans l'EDDS, leurs utilisations et objectifs.

Introduction

Les contraintes d'intégrité permettent d'assurer la cohérence des bases de données et de renseigner sur leur niveau de qualité. Dans les systèmes d'information géographique, la qualité des données provient également de la qualité de la localisation des objets et de leur niveau de cohérence spatiale.

I. Travaux effectués dans le domaine des CI

Les contraintes d'intégrité sont des règles assurant la cohérence logique de données. Les travaux liés à la modélisation des CI dans l'entrepôt de données (ED), l'entrepôt de données spatiales (EDS) sont nombreux et quelques travaux sont cités ci-dessous :

Dans ce chapitre nous allons voir les recherches faites par quelques auteurs en se concentrant sur deux classifications qui facilitent l'identification et l'implémentation des contraintes d'EDS : **1.** Classification orientée implémentation, **2.** Classification orientée concepts d'EDS.

Dans le cas d'entrepôt de données classique :

- L'auteur dans [3] propose un modèle multidimensionnel supportant des contraintes d'intégrité intra et inter-membres, inter-niveaux de dimension et inter-dimensions exprimées avec un langage logique.
- L'auteur dans [4] propose un modèle multidimensionnel en constellation. Ils introduisent les contraintes intra et inter-dimensions, qui définissent des inclusions et exclusions entre hiérarchies d'une ou plusieurs dimensions. Ces contraintes sont définies au niveau conceptuel en utilisant deux langages, logique et visuel (graphique).
- L'auteur dans [6] introduit de nouvelles contraintes, appelées contraintes de dimension, pour vérifier l'agrégabilité de dimensions hétérogènes. Ces contraintes définissent les chemins d'agrégation possibles des membres le long des hiérarchies de dimension. Elles sont définies avec un langage logique.

Dans le cas d'entrepôt de données spatiales :

- L'auteur dans [14] propose un modèle conceptuel spatio-multidimensionnel, nommé Spatial Multi Dim, étendant le modèle E/R. Ils utilisent des pictogrammes pour exprimer des CI spatiales simples entre deux niveaux de dimension spatiaux reliés par une relation roll-up et entre les niveaux spatiaux feuilles des dimensions.
- L'auteur dans [5] introduit les premières idées d'un profil UML pour modéliser les EDS selon une approche MDA. Ils ne considèrent que des CI spatiales de typage portant sur les attributs de dimension et mesures spatiaux. Ces contraintes sont représentées avec des pictogrammes.
- L'auteur dans [1] propose un modèle logique pour les EDS étendu par de nouvelles contraintes. Ces contraintes définissent des dépendances sémantiques entre fonctions d'agrégation spatiales et fonctions d'agrégation numériques.

- Enfin, par rapport aux travaux ci-dessus, L'auteur dans [18] propose la classification de CI la plus complète de toutes mais qui peut être encore étendue.

Il propose également les spécifications d'un langage hybride (langage enrichi par des pictogrammes) pour la modélisation des CI spatiales naturel contrôlé et temporelles dans les cubes de données spatiales).

II. Les classifications de contraintes d'intégrité d'EDS

Dans cette section, nous proposons deux nouvelles classifications des contraintes d'intégrité pour les entrepôts de données spatiales.

L'auteur dans [18] propose deux nouvelles classifications de CI d'EDS qui sont interdépendantes et complémentaires :

- La première est une classification orientée implémentation qui regroupe les contraintes selon les niveaux d'implémentation possibles.
- La seconde est une classification orientée concepts d'EDS (Fait, Dimension, Agrégation, etc.) qui se base sur la nature des éléments sur lesquels portent les CI.

Nous montrons comment spécifier les contraintes avec les langages OCL et OCL Spatial. Avant de détailler ces deux classifications, il est important de préciser la définition de contrainte d'intégrité, les CI d'entrepôt de données spatiales sont des règles définissant les relations spatiales permises, interdites ou obligatoires entre deux classes d'objets d'une base de données spatiales.

II.1. Classification orientée implémentation

A présent, nous allons décrire la première classification qui groupe les CI sur la base de leurs niveaux d'implémentation dans l'architecture SOLAP, Pour cette classification nous distinguons trois catégories de contraintes : contraintes ETL, contraintes EDS et contraintes SOLAP.

II.1.1. Contraintes ETL

Ces contraintes sont implémentées au niveau ETL pour prévenir le chargement de données erronées dans l'entrepôt de données. Elles peuvent être définies dans les programmes ETL par des transformateurs, composants dédiés aux traitements de données extraites de plusieurs sources ou par de composants dédiés à la qualité de données.

II.1.2. Contraintes EDS

Ces contraintes sont implémentées au niveau EDS, sous forme de vues et de requêtes SQL pour les architectures ROLAP. Ces contraintes peuvent être utilisées pour vérifier la validité des données entreposées mais plus particulièrement leur cohérence topologique.

II.1.3. Contraintes SOLAP

Ces contraintes sont utilisées pour garantir une agrégation correcte des mesures et une exploration correcte des hyper-cubes. Elles peuvent être implémentées soit dans le schéma de l'hyper-cube du Serveur SOLAP, soit dans le parseur des requêtes du serveur SOLAP, ou dans le client SOLAP. (Voir figure 2.1).

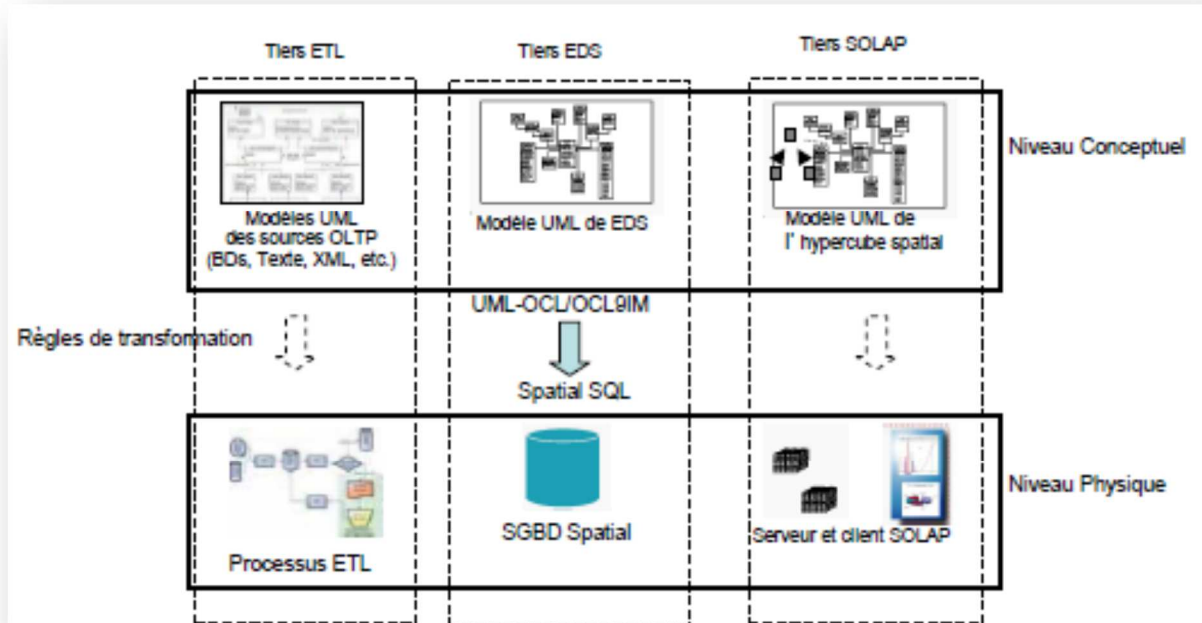


Figure 2.1 : Types d'implémentation des contraintes d'EDS. [13]

- Note : Il est important de noter qu'il est possible qu'une même contrainte puisse être implémentée et vérifiée dans les trois niveaux.

II.2. Classification orientée concepts d'EDS

Cette nouvelle classification distingue les CI suivant les concepts d'EDS (métadonnées, membres, faits, agrégation, etc.) sur lesquels elles portent. Notons que cette classification est très expressive car elle couvre toutes les classifications existantes.

En effet, ils étendent la classification la plus complète proposée dans [18] par d'autres catégories de CI : CI de schéma, de métadonnées, d'exhaustivité et CI inter-hyper-cubes.

II.2.1. Les contraintes intra hyper cube spatial

Elles portent sur des membres et des faits d'un seul hyper cube spatial. Nous distinguons quatre sous-classes :

A) Les contraintes sur les membres

Ces contraintes peuvent être utilisées pour identifier les membres incorrects et les relations entre les membres incorrectes. Elles portent sur un ou plusieurs attributs appartenant à un ou plusieurs niveaux de dimension. Nous distinguons donc les CI intra-niveau et les CI inter-niveaux.

1) CI Intra Niveau

Elles portent sur les membres et les relations entre les membres d'un seul niveau de dimension. Elles peuvent être aussi intra-attribut (portent sur un seul attribut) ou inter-attributs (Impliquent deux attributs ou plus).

Contrainte 1 : Les états américains situés dans la région "ouest" doivent être disjoints des états américains situés dans la région "Nord-est". Cette contrainte définit une relation spatiale topologique entre les membres (états) d'un seul niveau.

2) CI Inter Niveau

Elles portent sur les membres et relations entre membres de plusieurs niveaux de dimensions. Elles sont également inter-attributs.

Contrainte 2 : seuls les états situés dans les régions "ouest", "sud-ouest" ou "Nord-est" peuvent avoir des frontières avec des provinces. Cette contrainte porte région du niveau State et l'attribut location du niveau Province.

B) Les contraintes sur les faits

Ces contraintes impliquent un ou plusieurs faits d'un seul hyper cube spatial et peuvent être donc intra fait ou inter faits. Elles peuvent être utilisées soit pour identifier les faits incorrects et relations entre faits incorrectes, ou bien pour interdire l'exécution de requêtes spatio-multidimensionnelles sémantiquement incorrectes, portant sur un ou plusieurs faits.

1) **Contraintes intra fait :** elles portent sur une ou plusieurs mesures d'un seul fait d'un seul hyper cube.

Contrainte 3 : les feux qui se produisent aux USA (United States of America) ne peuvent pas être liés à des classes de feux canadiennes. Inversement, les feux se produisant au Canada, ne peuvent pas être liés à des classes de feux américaines.

2) **Contraintes inter faits:** elles portent sur les mesures de plusieurs faits d'un seul hyper cube spatial.

Contrainte 4 : aux États-Unis, la somme des surfaces détruites par les feux de classe "A" est supérieure à la somme des surfaces détruites par les feux de classe E" Cette contrainte porte sur plusieurs faits.

C) **Les contraintes d'agrégation :** elles garantissent des agrégations correctes des mesures de l'hyper cube. Pour cette classe de contraintes, nous distinguons quatre sous classes :

- 1) **Les contraintes d'agrégation utilisateurs :** Ces contraintes définissent les combinaisons correctes ainsi que leur ordre d'exécution.
- 2) **Les contraintes d'exhaustivité de données :** Elles vérifient la présence de toutes les données (membres et faits) dont l'analyse décisionnelle a besoin. Elles correspondent en partie à la condition de complétude de [12].

- 3) **Les contraintes de schéma** : Elles sont des conditions qui doivent être respectées lors de la définition des hiérarchies des dimensions et du choix de la granularité des faits. Elles portent sur les multiplicités des relations entre niveaux de dimension et celles des relations entre faits et dimensions.
- 4) **Les contraintes de calculabilité**: Elles garantissent une utilisation correcte des fonctions d'agrégation pour agréger les mesures le long des dimensions. Plus précisément, elles définissent les combinaisons (dimension, niveau à partir duquel on agrège, niveau vers lequel on agrège, mesure, fonction d'agrégation) correctes et incorrectes. Elles peuvent aussi définir les relations de dépendance entre plusieurs de ces combinaisons, en considérant par exemple, les dépendances entre les fonctions d'agrégation Spatiales et les fonctions d'agrégation numériques. Notons que ces CI doivent être valables pour tous les ED et EDS.

D) CI inter-hiérarchies

Ces contraintes définissent des exclusions et inclusions entre hiérarchies de dimension. Deux hiérarchies sont exclusives si leurs membres ne peuvent pas être combinés pour visualiser les mesures de l'hyper cube.

II.2.2. Les contraintes inter hyper cubes spatiaux

Elles portent sur des membres et des faits appartenant à plusieurs hyper cubes spatiaux. Elles peuvent être:

- A. **intra niveau** : Portent sur les membres d'un niveau partagé entre plusieurs hyper cubes spatiaux (schéma en constellation). [2]
- B. **inter niveaux** : Portent sur des membres de plusieurs niveaux appartenant à des hyper cubes spatiaux différents.
- C. **inter faits** : Impliquent plusieurs faits de plusieurs hyper cubes spatiaux.

Contrainte 5 : considérons un deuxième hyper cube pour l'analyse des inondations selon Les dimensions temps, localisations et types d'inondation. Ce hyper cube partage les dimensions temps et localisations avec l'hyper cube "incendies". Il a aussi deux mesures zone inondée (spatiale) et niveau de précipitations (numérique). Un exemple simple de contrainte (zone inondée) des incendies et inondations, se produisant à la même date, doivent être disjointes. (Voir figure 2.2)

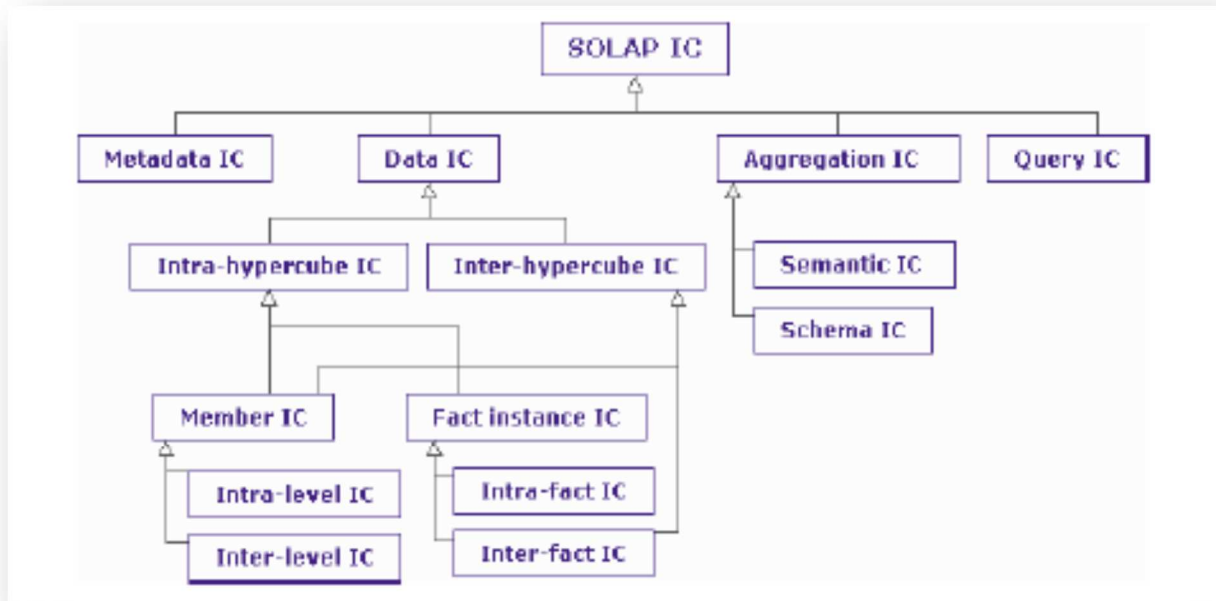


Figure 2.2 : Classification orientée concepts d'EDS. [18]

II.2.3. Les CI de versionnement

Ces contraintes vérifient la cohérence entre les différentes versions d'un EDS ; deux types de cohérences doivent être garantis :

- ✓ **cohérence entre schémas** : à vérifier dans le cas d'une évolution du schéma de l'EDS.
- ✓ **cohérence entre instances** : à vérifier dans le cas d'un changement au niveau des données.

II.3. Spécification des CI intra-hyper-cube et inter-hyper-cubes

Dans cette section, Les auteurs présentent la spécification en OCL et OCL spatial des exemples de contraintes intra hyper cube spatial, leur contribution est de montrer que ces contraintes peuvent facilement s'exprimer en utilisant les standards UML et OCL. Ils utilisent pour cela, les expressions OCL de type invariant (Mot clef inv) et les diagrammes objets UML. Un invariant OCL exprime une condition qui doit être en permanence vérifiée par un ensemble d'objets (OMG, 2010c). Nous citons donc ce qui suit deux exemples des CI traités dans le travail. [18]

1. Les contraintes sur les membres

Contrainte 1: C'est une contrainte OCL de type invariant (mot clef inv.) qui s'applique à une instance "State" (Contextes State). Un invariant OCL exprime une contrainte sur un objet ou un groupe d'objets qui doit être respectée en permanence. En utilisant la syntaxe d'OCL, (géoA).opération topo (géoB), elle définit une relation spatiale topologique (areDisjoint) entre l'instance courante de "State" (self) et chaque instance "State" (s) de la

collection des instances "State" retournée par l'expression "State.allInstances". Le mot clef self est utilisé dans OCL pour référencer l'objet courant de la classe spécifiée dans le contexte. (Voir figure 2.3).

```
Context State inv:  
State.allInstances -> forAll(s |  
  (self.region = 'West' and s.region = 'Northeast') implies  
  self.location.areDisjoint (s.location))
```

Figure 2.3 : L'expression spatial OCL de la contrainte 1. [18]

Contrainte 2: Cet invariant vérifie pour chaque objet courant "State" (self), s'il existe un objet "Province" p, dans la collection des objets "Province", tel que les géométries des deux. (Voir figure 2.4).

```
context State inv:  
Province.allInstances -> exists(p |  
  self.location.areAdjacent(p.location)) implies  
  ((self.region = 'West') or (self.region = 'Midwest') or  
  (self.region = 'Northeast'))
```

Figure 2.4 : L'expression spatial OCL de la contrainte 2. [18]

2. Les contraintes sur les faits

Contrainte 3: cette contrainte porte sur un fait (instance de la classe FireDisaster). Elle vérifie que le fait courant (self) n'est pas, au même temps, lié à une ville américaine et une classe de feux canadienne et inversement. "State.allInstances.county.city.fireDisaster" retourne la collection des faits (instances de la classe FireDisaster) liés à la hiérarchie "localisations USA". De même, "Canadian Class.allInstances.fireClass.fireDisaster" retourne la collection des

faits liés à la hiérarchie des classes de feux canadiennes. L'expression "collection ->excludes (self)" retourne vrai si l'objet "self" n'appartient pas à collection. (Voir figure 2.5).

```
context FireDisaster inv :
(State.allInstances.county.city.firedisaster ->
excludes(self) xor
CanadianClass.allInstances.fireclass.firedisaster ->
excludes(self) ) and
(Province.allInstances.county.city.firedisaster ->
excludes(self) xor
USAClass.allInstances.fireclass.firedisaster ->
excludes(self) )
```

Figure 2.5 : L'expression spatial OCL de la contrainte 2. [18]

III. Implémentation des contraintes au niveau du tiers EDS

Pour implémenter le modèle conceptuel UML d'EDS et les contraintes de données intra hyper cube spatial spécifiées en OCL et OCL 9IM (sections 5.1 et 5.2) sous Oracle spatial, nous avons utilisé l'extension spatiale d'OCL2SQL. OCL2SQL est un outil open source, entièrement écrit en java. Il permet de générer à partir d'une contrainte OCL *c*, une requête SQL qui sélectionne toutes les données qui ne satisfont pas *c*. Le principe de génération de code est basé sur des patrons de conversion d'OCL vers SQL pour Oracle. Il intègre les extensions spatiales d'OCL appelées OCL 9IM et OCL ADV et Il implémente les règles pour leurs transformations vers le SQL d'Oracle Spatial. Pour implémenter les contraintes et l'entrepôt de données spatiales.

Nous avons procédé en 5 étapes :

- ✓ **Définition du schéma XMI :** Le modèle multidimensionnel UML de l'EDS (figure 3) est traduit en schéma XMI 1.0.
- ✓ **Définition des contraintes OCL :** La définition de chaque CI doit tenir compte de la syntaxe supportée par l'outil Spatial OCL2SQL et des éléments définis dans le schéma XMI.
- ✓ **Définition du fichier de métadonnées géométriques :** Il est utilisé pour mettre à jour la table des métadonnées spatiales d'oracle.
- ✓ **Génération des scripts SQL pour Oracle Spatial :** Cette étape consiste à exécuter le projet pour générer les scripts SQL pour Oracle Spatial.
- ✓ **Implémentation des scripts sous Oracle Spatial :** La dernière étape consiste à exécuter les scripts générés sous Oracle Spatial pour implémenter les contraintes et l'EDS. [18]

Conclusion

Nous avons vu dans ce chapitre que La qualité des analyses OLAP et SOLAP dépend fortement de la qualité et l'exploration de ces données. Dans ce contexte, différents travaux se sont intéressés à la définition de contraintes d'intégrité dans les ED et EDS. Ces Différents travaux proposent différentes approches : contraintes d'intégrité, dépendances fonctionnelles, formes normales multidimensionnelles, règles de production, etc. En effet, les CI peuvent être utilisés pour identifier les données erronées, définir les règles d'agrégation correctes et aussi identifier les requêtes multidimensionnelles incorrectes. Ensuite nous avons présenté le domaine ciblé par notre solution d'aide à la décision, ainsi que la zone d'étude concernée qui est l'ADE « la direction algérienne des eaux » de la wilaya de RELIZANE.

Introduction

Dans ce chapitre, nous allons donner une vue sur la zone étudiée et de ses spécifications. Par la suite nous parlerons de l'entreprise où nous avons fait notre stage en essayant d'exploiter les données recueillies de cette entreprise.

I. Présentation de la zone d'étude

Pour réaliser notre projet d'aide à la décision, nous avons choisi la Wilaya de Relizane qui est située au nord-ouest de l'Algérie et s'étend sur une superficie globale de 484.000 hectares soit 4851,21 km². (Voir figure 3.1)

- Nombre de communes de la Wilaya : 38
- Population de la Wilaya : 803 278hab.



Figure 3.1 : L'emplacement de la Wilaya de Relizane.

I.1. L'Algérienne Des Eaux (ADE)

L'ADE Est un établissement public national à caractère industriel et commercial doté de la personnalité morale et de l'autonomie financière. Il a été créé par le décret exécutif n° 01-101 du 27 Moharrem 1422 correspondant au 21 Avril 2001. L'établissement est placé sous la tutelle du ministre chargé des ressources en eau, et son siège social est fixé à Alger.

I.2. L'Algérienne Des Eaux (ADE) de la wilaya de Relizane

L'entreprise des Eaux de RELIZANE gère actuellement 30 communes dont 04 en gros, représentant une population desservie actuelle de 609 426 hab. sur un Total de 803 278hab. soit un taux de 75% de la population totale de la wilaya. (Voir figure 3.2)



Figure 3.2. Carte de gestion des communes de la wilaya de Relizane.

II. Ressources en Eau

II. 1 Eau De Surface : on a 3 barrages et 07 stations de traitement.

II. 1.1 Les barrages

- Barrage Sidi M'hamed Ben Ali desservi la commune de Relizane et Bendaoued.
- Barrage Gargar : desservi les communes suivant Hamri, Beni Zenthis, O/Rhiou, Lahlaf, O/Yaiche, El hassi, Beni Dergoun, Zemmoura, O/El Djemaa, Mendes, Oued Esselam, Dar Ben Abdellah et Sidi Lazreg.
- Barrage kramis: desservi la commune de Mediouna, Beni Zenthis et Sidi M'hamed Ben Ali.

II. 1.2 Les stations de Traitement

La wilaya de Relizane dispose de sept Stations de traitement (03 Relizane, 01 à Ben Daoud, 01 à Gargar, 01 Hamri et 01 Souk El had). Ils sont répartis comme suit : (Voir tableau 3.1)

Station de Traitement	Date de Mise en Service	Capacité de Production	Production Actuelle
Relizane A	1945	6048 m ³ /j (70 l/s)	5184 m ³ /j (60 l/s)
Relizane B	1985	17 280m ³ /j (200l/s)	15984m ³ /j (185 l/s)
Khachab(Bendaoued)	2007	1 728 m ³ /j (20 l/s)	1000 m ³ /j (10 l/s)
Hamri	2004	4320 m ³ /j (50 l/s)	1 600 m ³ /j
Gargar	2010	34560m ³ /j (400 l/s)	17000 m ³ /j
Bormadia	2012	1728 m ³ /j (20 l/s)	1500 m ³ /j
Sidi Yaakoub	2008	1728 m ³ /j (20 l/s)	/

Tableau 3.2 : les stations de traitement.

Le tableau ci-dessous présente les communes alimentées par eau de surface. (Voir tableau 3.2)

Communes	Population Desservie	Mode de Gestion	Stations de Traitement	Production M ³ /J	Dotation hydrique
Relizane	126 322	En Détail	St A & B de Relizane	19000	150
	13 599	Mixte		1400	103
Bendaoued	1000	En Gros	Khacheb	500	500
Hamri	3 377	Mixte	Hamri	750	220
Ben Zenthis	1 963	En Gros		450	230
Zemmoura	29 205	En Détail	Gargar	3 500	120
O/Djemaa	10 662			2 700	250
Beni Dergoun	6 923			800	115
O/Yaiche	4 445			500	115

Tableau 3.2 : les communes gérées par l'eau de surface.

II.2 Eau Souterraine

La wilaya de Relizane dispose de deux sources et 64 forages avec 50 en services et un seul puis. (Voir tableau 3.3)

Communes	Population Desservie	Mode de gestion	Nombre de Forages	Production m ³ /J	Dotation hydrique
O/Rhiou	64 378	En Détail	07	16 128	210
Djdiouia	28 819	En Détail	05	4 392	140
A/Moussa	29 412	En Détail	03	6500	140
A/Tarik	12 350	En Détail	03	6 500	140
El Ouldja	3 734	En Gros	03	6 500	140
Mendes	11 318	En Détail	04	504	55
El Matmar	15 359	En Détail	04	2 000	130
Yellel	20 339	En Détail	07	1 800	90
Mazouna	24 938	En Détail	06	8500	100
S.M. Ben Ali	20 276	En Détail	06	8 500	100
Mediouna	17 524	En Gros	06	8500	100
El Guettar	15 361	En Détail	04	1152	100

Tableau 3.3 : les communes gérées par l'eau souterraine.

La wilaya de Relizane contient 30 station de pompages ou 04 à Relizane ; 01 à Bendaoued ; 01 à Hamri, 03 à Bas Chélif, 03 à El Ouanchariss, 10 à Gargar, 03 à Kramis, 03 à Béni Zenthis et 02 à Souk El Had. Et 115 réservoirs avec une capacité de 95550 m³. (Voir figure 3.3)

Introduction

Dans ce chapitre, on a commencé par définir le modèle multidimensionnel utilisé. Puis les étapes suivies pour réaliser le système d'aide à la décision. Ce dernier est basé sur la technologie SOLAP pour l'ADE de la wilaya de Relizane. Nous avons aussi présenté les différentes techniques et logiciels utilisés pour finaliser notre travail.

I. Le modèle décisionnel proposé

La figure ci-dessous montre un modèle décisionnel qui va nous aider à développer notre application tel que la première partie (Source de données+ETL+EDDS) est pour la création de notre base de données et la deuxième pour la création d'un moteur SOLAP. (Voir figure 4.1)

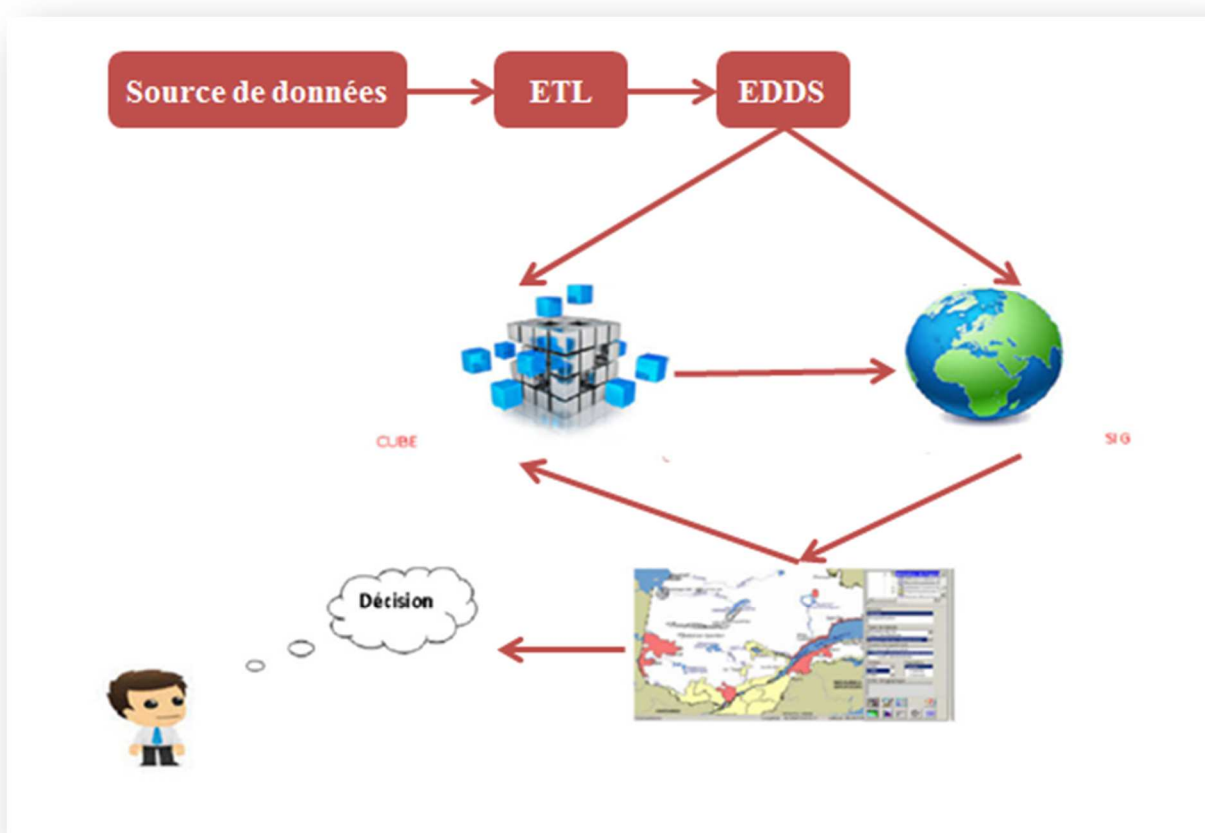


Figure 4.1 : Le modèle décisionnel proposé.

II. Modélisation

Le modèle multidimensionnel utilisé dans ce travail est un modèle en constellation qui est basé sur le modèle en étoile (le centre est la table des faits, et les branches en sont les dimensions). Mais on rassemble plusieurs tables des faits qui partagent quelques dimensions.

II.1 MagicDraw 18.0

MagicDraw est un outil de modélisation. Conçu pour les analystes d'entreprise, Les analystes de logiciels, programmeurs, ingénieurs et rédacteurs de documentation. Cet outil de développement dynamique et polyvalent facilite l'analyse et la conception orientée objet (OO), systèmes et bases de données.

Le schéma suivant représente le diagramme de classes de l'entrepôt de donnée spatiales de notre solution décisionnelle. Ce diagramme est créé sous MagicDraw. [21] (Voir figure 4.2)

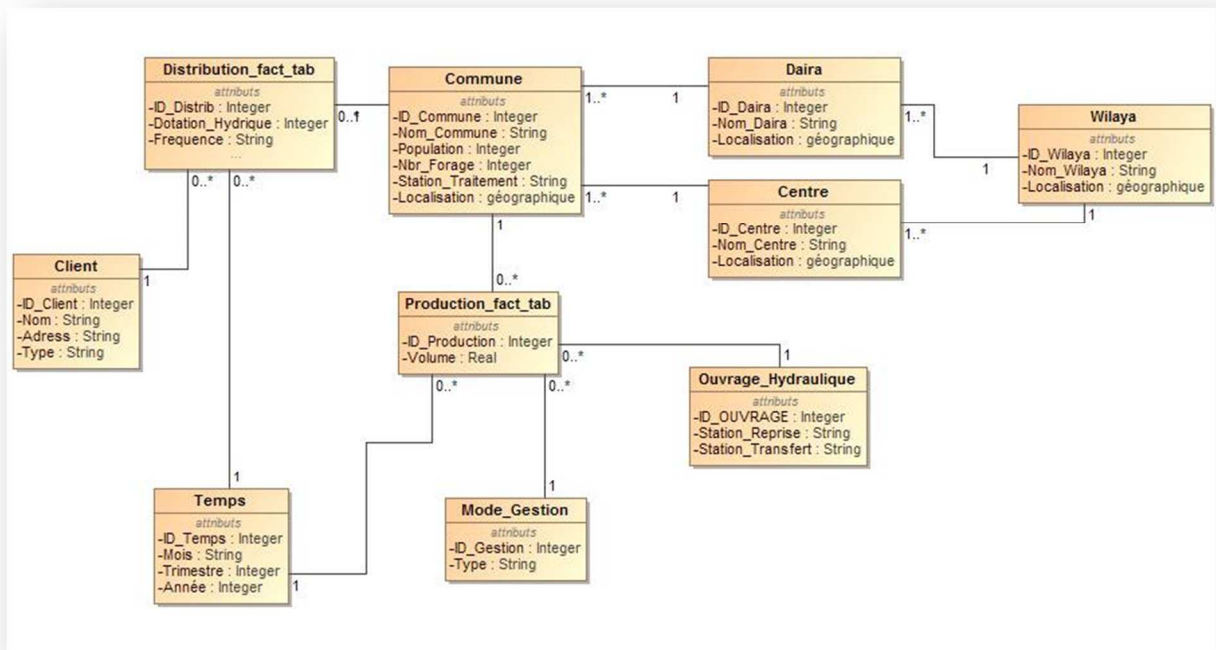


Figure 4.2 : Diagramme de classe de l'entrepôt de donnée spatiale.

Dans ce diagramme, nous avons dix classes : chaque classe a ses attributs, et entre chaque deux classes nous avons une relation de type association.

Nous avons 2 types de classes :

1-tables de fait : la table distribution contient la mesure : Dotation Hydrique et la table production contient la mesure Volume.

2-niveau d'agrégation pour les dimensions

- Dimension temps : c'est une dimension temporelle définie par la classe temps.
- Dimension client : c'est une dimension thématique définie par la classe client.
- Dimension ouvrage hydraulique : c'est une dimension thématique définie par classe ouvrage hydraulique.
- Dimension mode de gestion : c'est une dimension thématique définie par classe mode de gestion.

- Dimension géographique : c'est une dimension géographique définie par les classes (commune, daïra, centre, wilaya).

Cette dernière dimension contient 2 hiérarchies:

- la hiérarchie commune->daïra->wilaya
- et la hiérarchie commune->centre->wilaya.

III. Implémentation

Notre solution de développement d'outil d'aide à la décision peut être schématisée selon l'organigramme suivant (Voir figure 4.3):

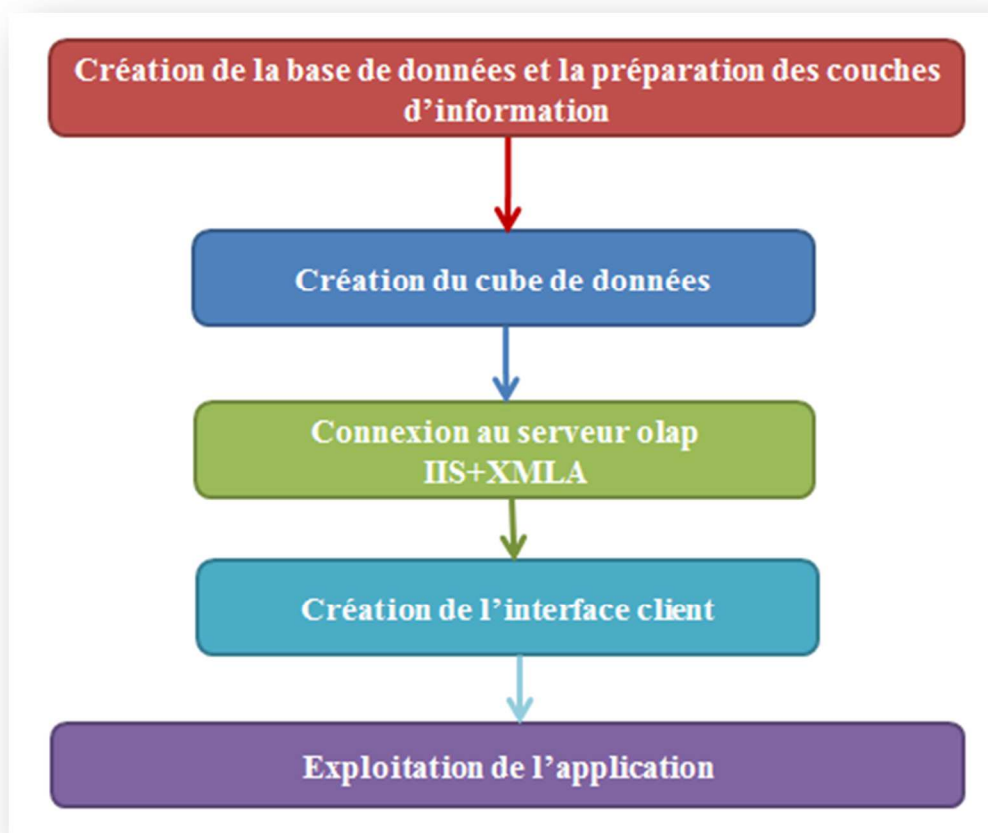


Figure 4.3 : organigramme de solution.

III.2 Les données utilisées

Pour notre projet, on a besoin de deux types de données :

- Données alphanumériques : qui représentent la totalité des statistiques (Nom de commune, Daïra, Centre, Volume de Production...).
- Données géographiques : La carte de la Wilaya de Relizane et la carte de l'Algérie ainsi que les différentes couches utilisées (Des polygones qui représentent les Communes, Daïra, Centre. Des points qui représentent les

Stations de Traitement, Réservoirs, Clients. Des lignes qui représentent le réseau de Transfert par Barrage, Le Réseau de Transfert par Champ Captant).

III.3 Les outils de développements utilisés

Notre choix était porté sur l'outil Microsoft SQL server 2012 pour implémenter notre base de données, l'outil Arc Gis pour le développement de notre système d'information géographique et Eclipse pour le développement de notre interface.

- **Microsoft SQL Server 2012**

Microsoft SQL Server est une application utilisée pour créer des bases de données informatiques pour la famille des systèmes d'exploitation de Microsoft Windows. Il fournit un environnement utilisé pour produire des bases de données accessibles à partir des postes de travail, du web ou d'autres média tels qu'un assistant numérique personnel. SQL Server 2012, la plate-forme de gestion et d'analyse des données la plus complète. [32]

- **Eclipse**

Eclipse est un projet, décliné et organisé en un ensemble de sous-projets de développements logiciels, de la Fondation Eclipse, visant à développer un environnement de production de logiciels libres qui soit extensible, universel et polyvalent, en s'appuyant principalement sur Java. [32]

- **Microsoft Visual Studio 2010**

Visual Studio est un ensemble complet d'outils de développement permettant de générer des applications Web ASP.NET, des Services Web XML, des applications bureautiques et des applications mobiles. Ces outils permettent de créer notre application. [33]

- **ArcGIS 10.2**

ArcGIS est une suite de logiciels d'information géographique (ou logiciels SIG) développés par la société américaine Esri (Environmental Systems Research Institute, Inc.). L'utilisation de ces outils nous permet de développer notre système d'information géographique. [34]

III.4 Création de la base de données et la préparation des couches d'information

Dans cette étapes on va créer notre base de données en utilisant SQL Server 2012 plus précisément SQL Server Management Studio. Les déférentes couches d'information et la geodatabase sont créés sous ArcGIS 10.2.

La figure ci-dessus représente Fenêtre de SQL Server management studio (Voir figure 4.4):

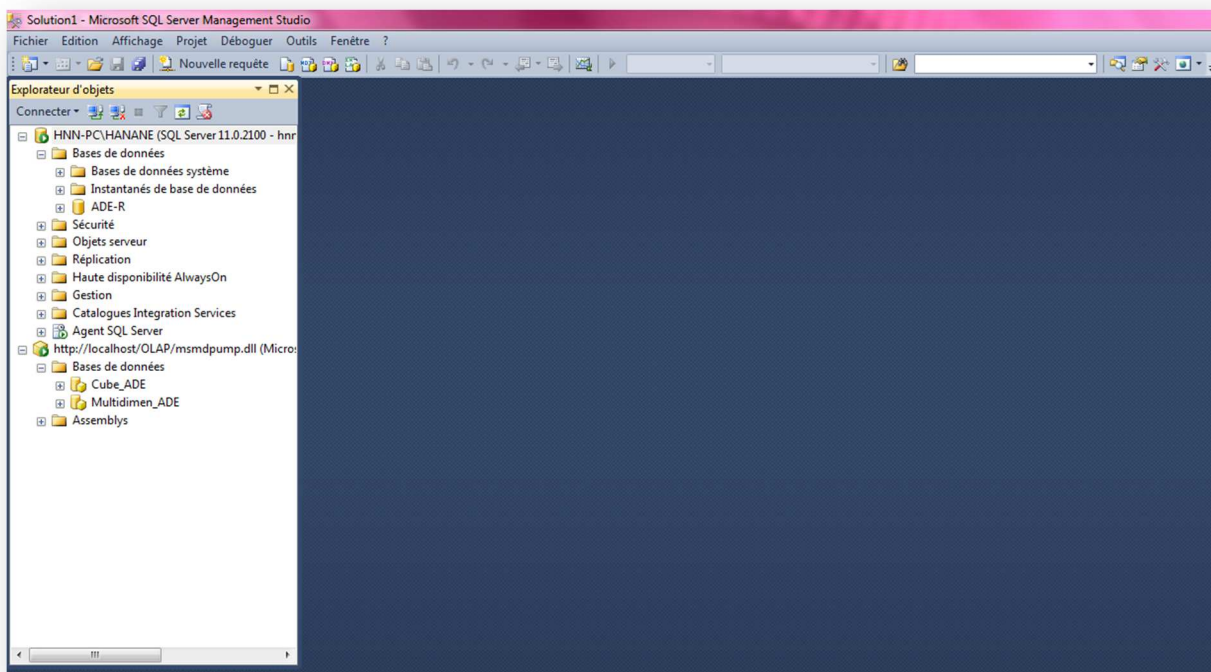


Figure 4.4 : Fenêtre de SQL Server management studio.

III.4.1 La création de la base de données

La création de la base de données se fait à travers SQL Server Management Studio. (Voir figure 4.5)

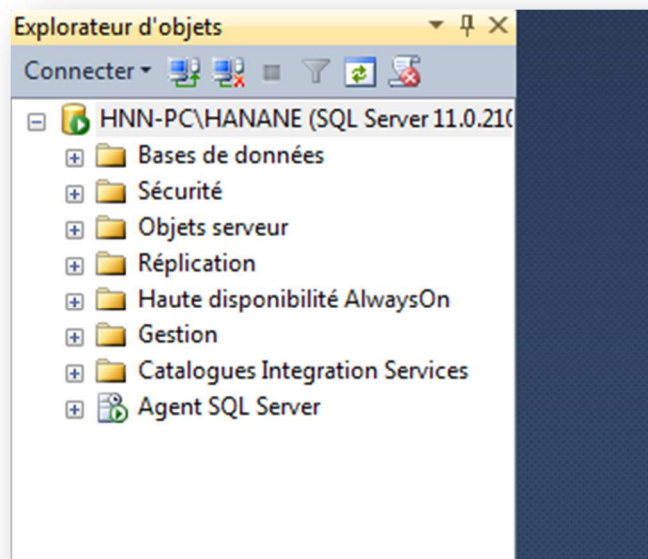


Figure 4.5 : Création de la base de données.

Une fenêtre s'ouvre où nous allons préciser le nom de la base de données : ADE_R et le nom serveur : hnn-PC\hnn. (Voir figure 4.6)

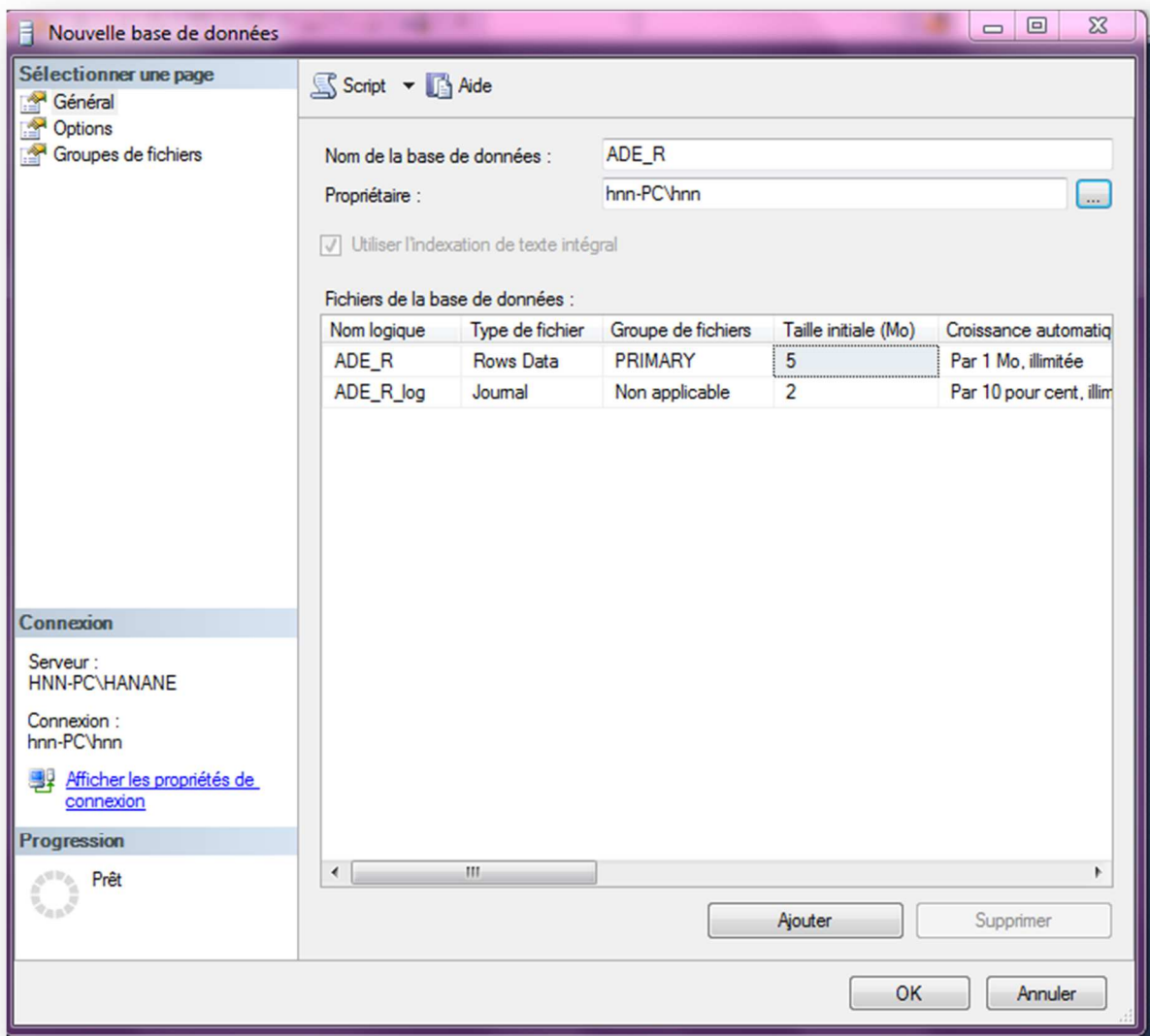


Figure 4.6 : Fenêtre de création de la base de données.

Après la validation de la base de données, on accède à une autre section où nous pouvons créer les tables que nous avons besoin dans notre base. (Voir figure 4.7)

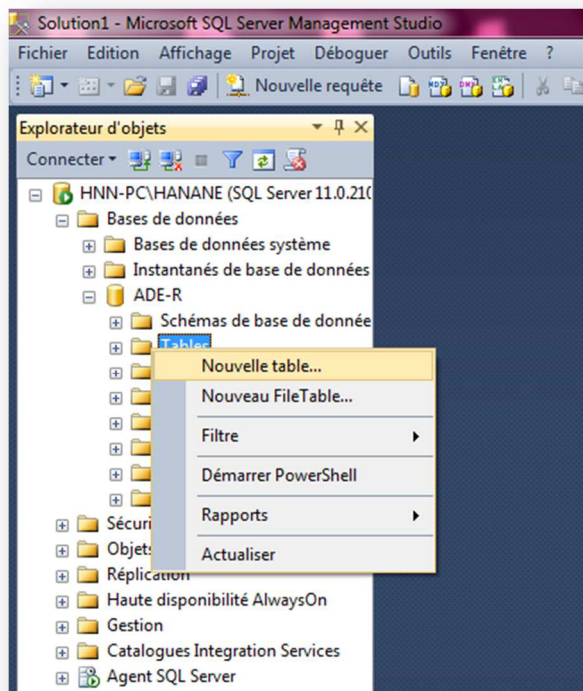


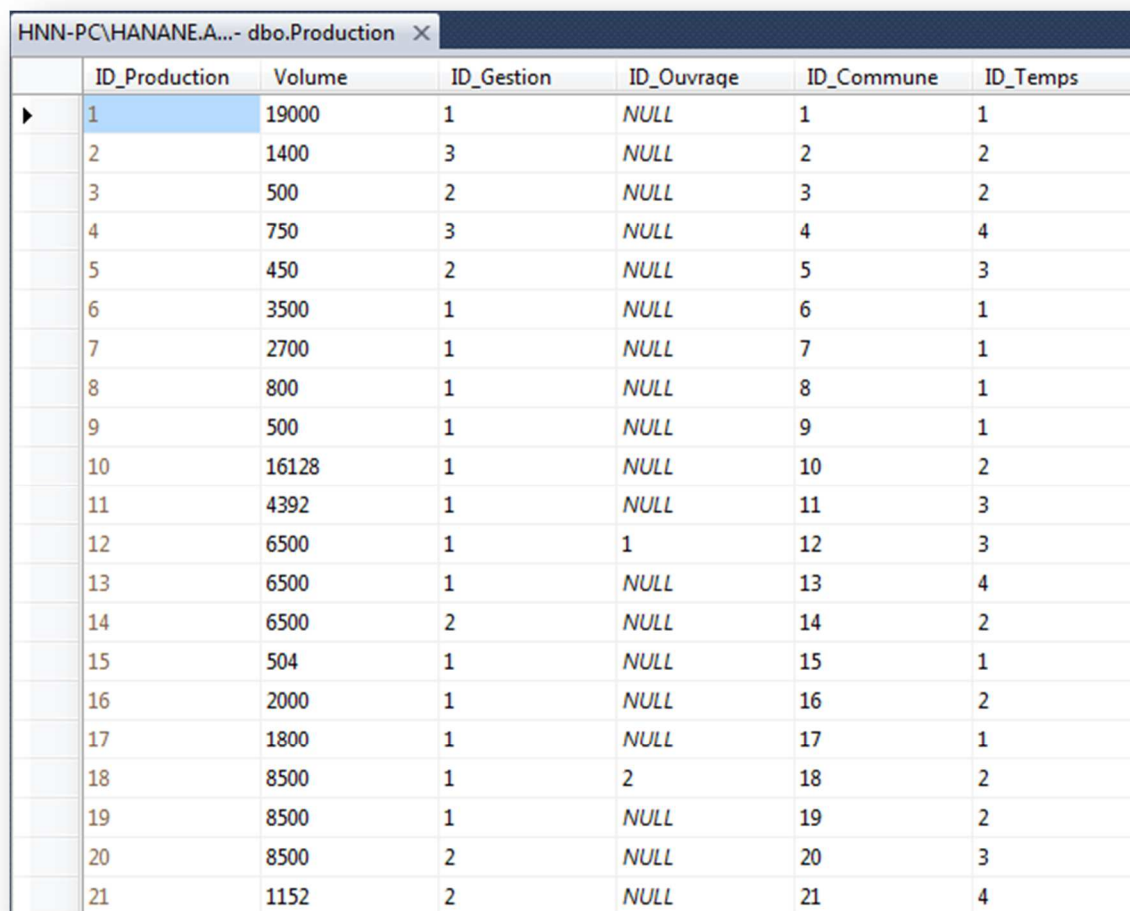
Figure 4.7 : Création de la table.

Depuis cette fenêtre, nous pouvons créer des colonnes pour la table, en spécifiant le type de donnée et la clé primaire de chaque table. (Voir figure 4.8)

	Nom de la colonne	Type de données	Autoriser le...
🔑	ID_Production	int	<input type="checkbox"/>
	Volume	float	<input checked="" type="checkbox"/>
	ID_Gestion	int	<input checked="" type="checkbox"/>
	ID_Ouvrage	int	<input checked="" type="checkbox"/>
	ID_Commune	int	<input checked="" type="checkbox"/>
	ID_Temps	int	<input checked="" type="checkbox"/>
▶	ID_tr		<input type="checkbox"/>

Figure 4.8 : Création des colonnes pour la table.

Après la création des tables nous allons les remplir, la fenêtre ci-dessus montre comment le faire. (Voir figure 4.9)



	ID_Production	Volume	ID_Gestion	ID_Ouvrage	ID_Commune	ID_Temps
▶	1	19000	1	NULL	1	1
	2	1400	3	NULL	2	2
	3	500	2	NULL	3	2
	4	750	3	NULL	4	4
	5	450	2	NULL	5	3
	6	3500	1	NULL	6	1
	7	2700	1	NULL	7	1
	8	800	1	NULL	8	1
	9	500	1	NULL	9	1
	10	16128	1	NULL	10	2
	11	4392	1	NULL	11	3
	12	6500	1	1	12	3
	13	6500	1	NULL	13	4
	14	6500	2	NULL	14	2
	15	504	1	NULL	15	1
	16	2000	1	NULL	16	2
	17	1800	1	NULL	17	1
	18	8500	1	2	18	2
	19	8500	1	NULL	19	2
	20	8500	2	NULL	20	3
	21	1152	2	NULL	21	4

Figure 4.9 : Exemple d'une table.

Après la construction de toutes les tables de notre projet, on doit relier entre les tables par les clés primaires et les clés étrangères. Pour cela, on doit créer un schéma de base de données pour établir les jointures entre les tables. (Voir figure 4.10)

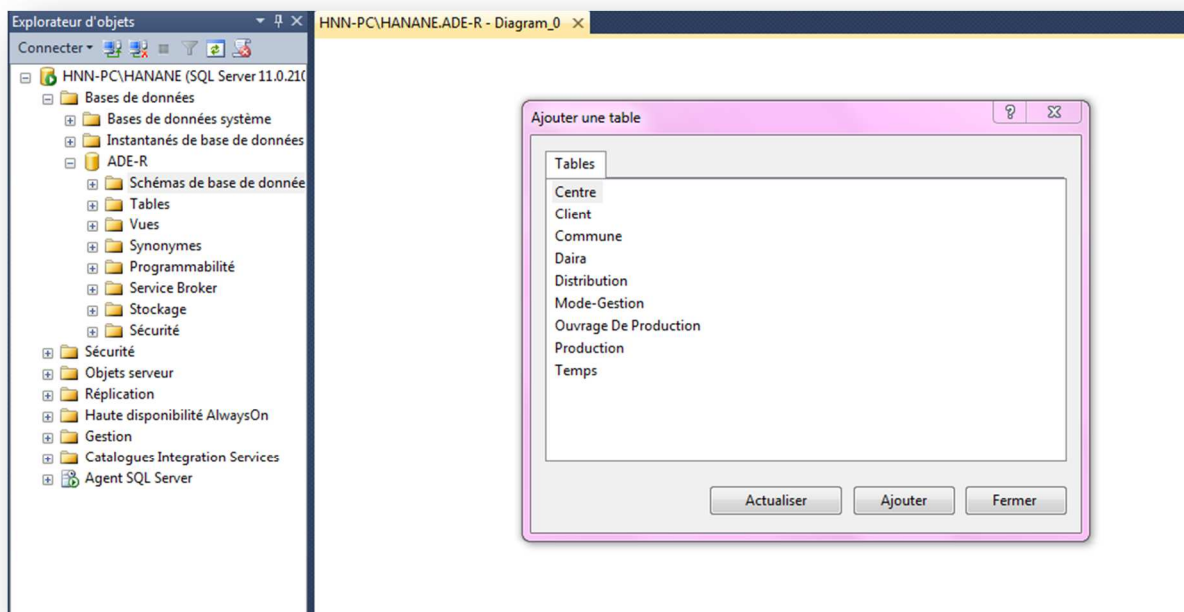


Figure 4.10: La création de schéma de la base de données.

Le schéma suivant représente le diagramme de la base de données ADE (Voir figure 4.11)

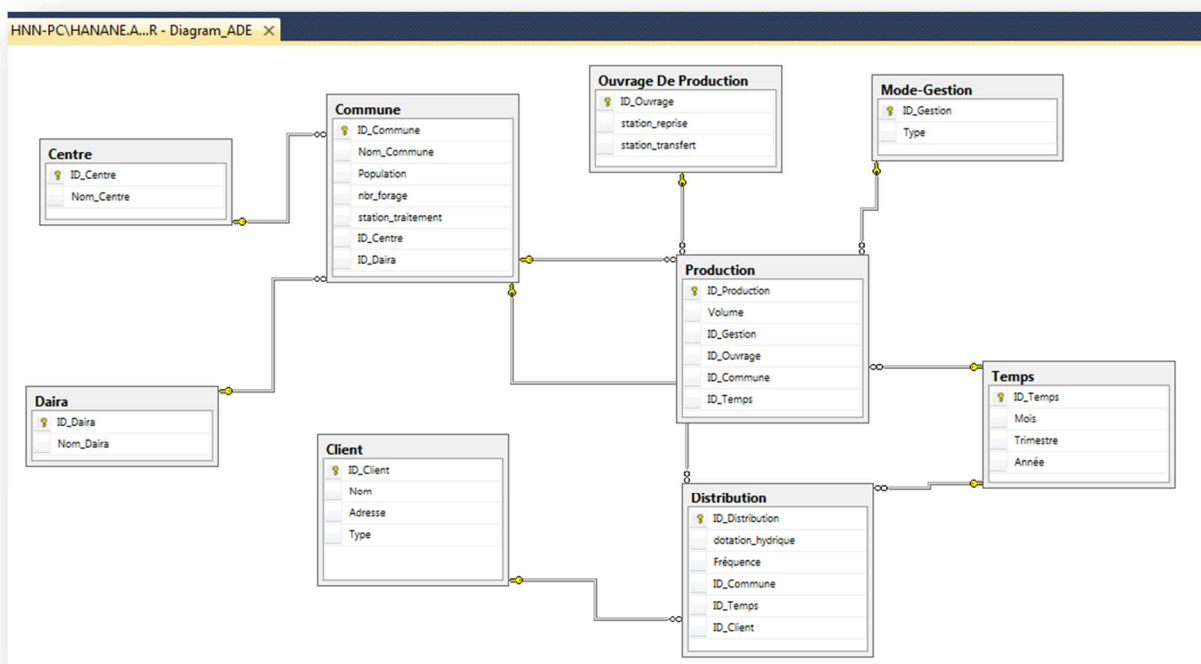


Figure 4.11 : Création d'un schéma de la base de données.

III.4.2 La préparation des couches d'information

Nous avons fait le calage de la carte de la wilaya de Relizane avec la projection « WGS_1984_UTM_Zone_31N ; Projection : Transverse Mercator ». Le calage se fait sur la carte de l'Algérie. (Voir figure 4.12)

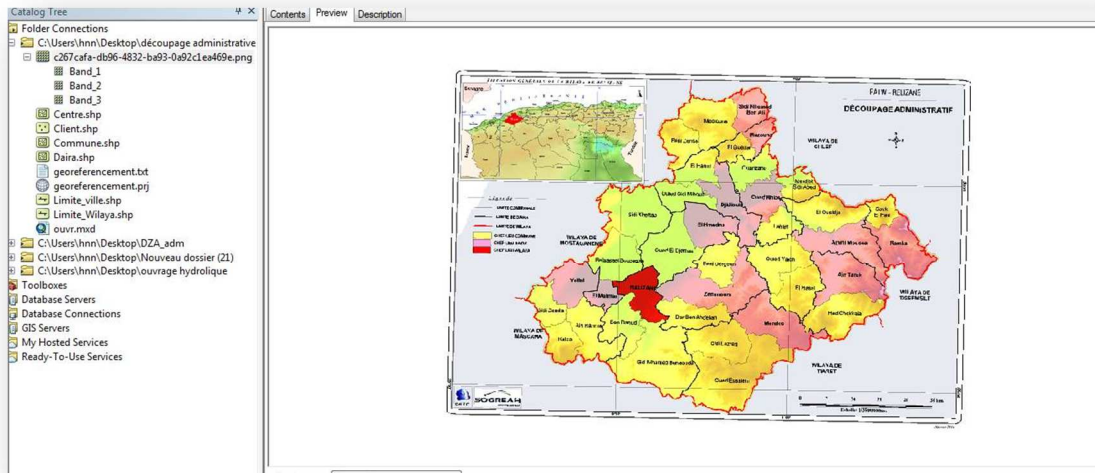


Figure 4.12 : le géoréférencement.

Après, nous allons créer les différents fichiers Shape qu'on a besoin en ajoutant les champs nécessaires dans les tables. (Commune, Daïra, Centre, Client, Stations de traitement, Réservoirs, Réseau de transfert).

Ensuite, on passe à la numérisation des couches d'information et le remplissage de ses tables. (Voir figure 4.13)

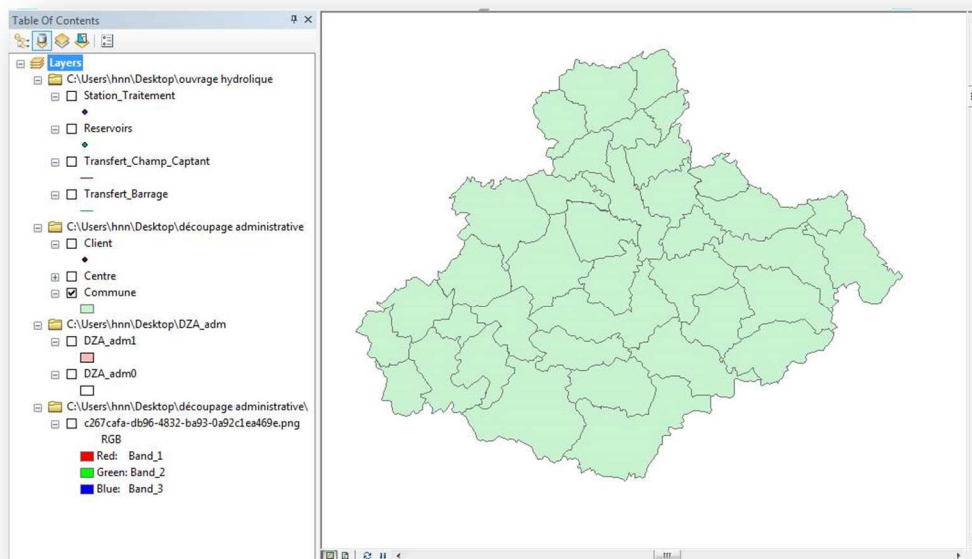


Figure 4.13 : la numérisation de la carte.

FIN	Shape #	Id	Nom	Population	Nbr Fra	Station Tr
0	Polygon	2	S.M. Ben Ali	20276	6	
1	Polygon	1	Medouna	17524	6	
2	Polygon	3	Ben Zerthia	1963	0	Hamri
3	Polygon	6	Mazouna	24938	6	
4	Polygon	5	El Guettar	15361	4	
5	Polygon	4	El Hamri	3377	0	Hamri
6	Polygon	9	Ouarzane	0	0	
7	Polygon	13	O'Rhou	64378	7	
8	Polygon	14	Merdjet Sidi Albed	0	0	
9	Polygon	37	El Ouldja	3734	3	
10	Polygon	38	Souk El Had	0	0	
11	Polygon	20	Ramka	0	0	
12	Polygon	19	A/Moussa	29412	3	
13	Polygon	26	A/Tank	12550	3	
14	Polygon	33	Had Chelkala	0	0	
15	Polygon	36	Oued Essakala	0	0	
16	Polygon	32	Mendes	11318	4	
17	Polygon	14	Lahaf	0	0	
18	Polygon	18	O'Yacine	4445	0	Gargar
19	Polygon	25	El Hassi	0	0	
20	Polygon	8	Djouia	28819	5	
21	Polygon	7	Ouled Sidi Mhoub	0	0	
22	Polygon	10	Sidi Khatab	0	0	
23	Polygon	12	El Hmadna	0	0	
24	Polygon	17	Ben Dergoun	8923	0	Gargar
25	Polygon	24	Zemmoura	26205	0	Gargar
26	Polygon	16	O'Djema	10682	0	Gargar
27	Polygon	15	Belassel Bouzegza	0	0	
28	Polygon	31	Dar Ben Abdelah	0	0	
29	Polygon	35	Sidi Lazreg	0	0	
30	Polygon	23	Reizane	126322	0	SH/SIB
31	Polygon	30	Sidi Mhamed Benaouda	0	0	
32	Polygon	29	Ben Darou	1000	0	Khacheb

Figure 4.14 : La création de la godatabase.

III.5 Création du cube de données

Pour la création du cube de données nous allons suivre les étapes :

- Création de la source de données.
- Création de la vue de source de données. (Voir figure 4.15)

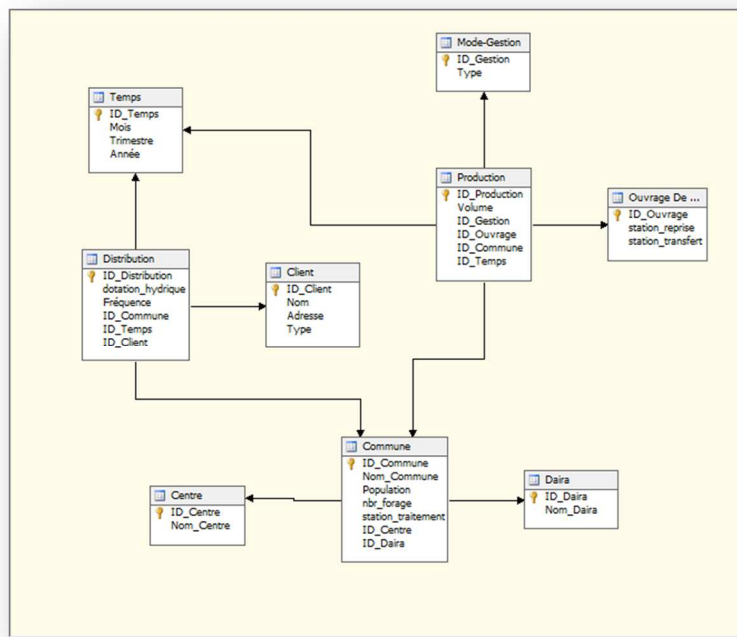


Figure 4.15 : Model Multidimensionnel de la base de données.

- Création des dimensions.
- Création des hiérarchies. (Voir figure 4.16)

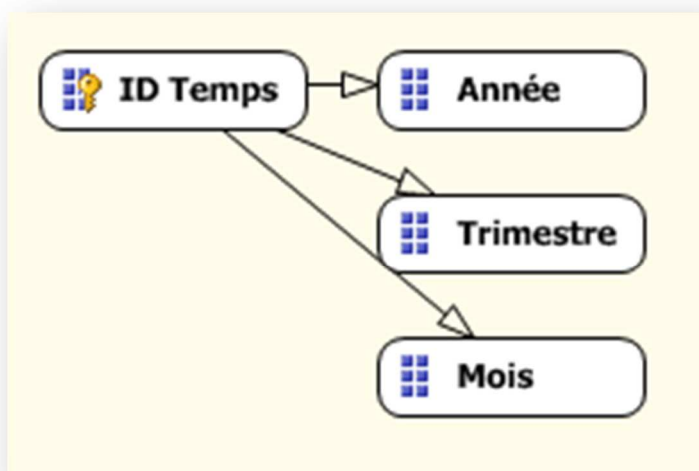


Figure 4.16 : Hiérarchie temps.

- Création de cube de données.

Le schéma ci-dessous représente le cube de données (Voir figure 4.17):

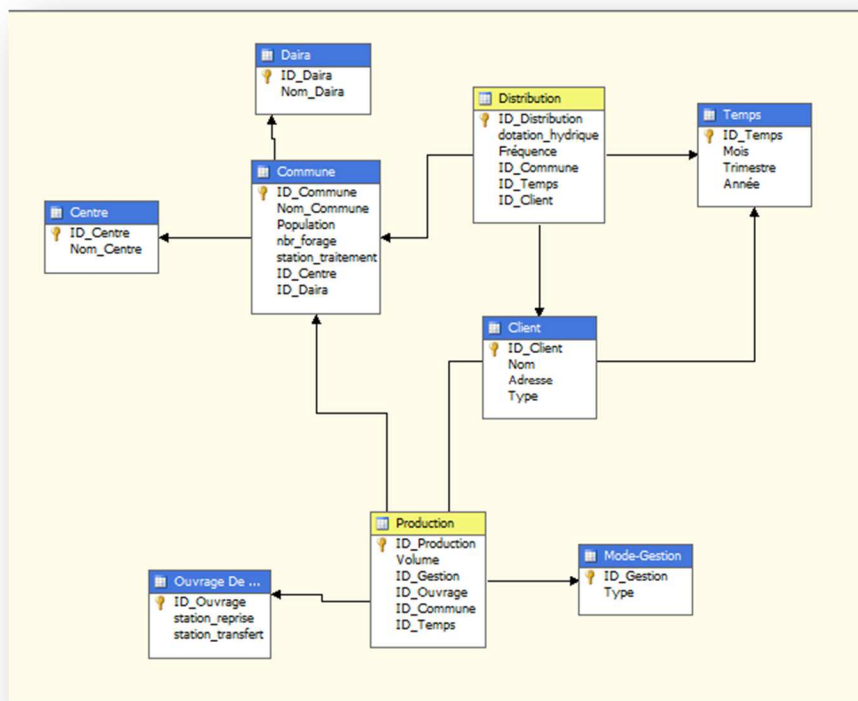


Figure 4.17 : le schéma du cube de données.

Modélisation et implémentation de la solution

Nous avons deux tables de faits (Distribution, Production) qui contiennent les clés des autres tables de dimensions ainsi que les mesures : volume pour production et Dotation Hydrique pour Distribution. Ainsi que huit tables de dimensions (Commune, Daira, Centre, Mode de Gestion, Ouvrages Hydraulique, Temps, Client) qui contiennent les enregistrements nécessaires pour l'agrégation des requêtes analytiques.

La réalisation de ces étapes est suivie par le déploiement et le traitement pour vérifier que le cube est correctement créé. (Voir figure 4.18 et 4.19)

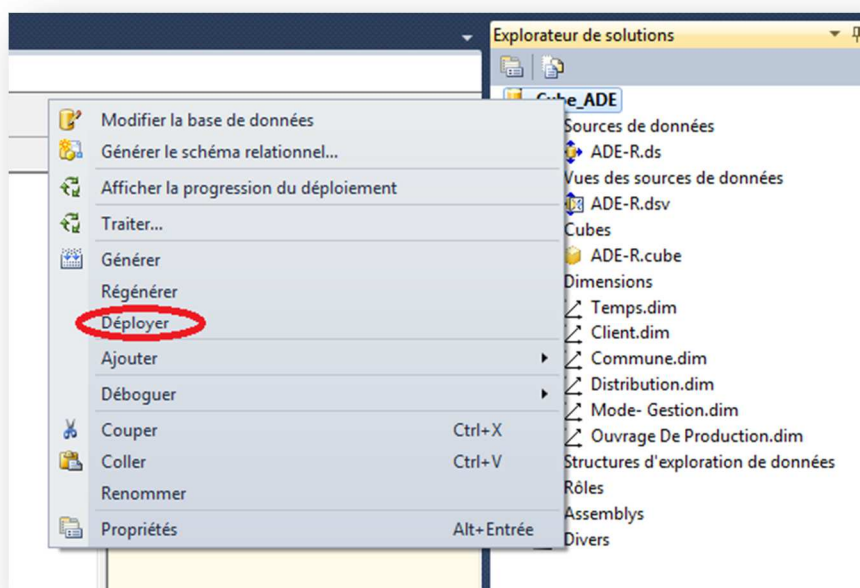


Figure 4.18 : le déploiement du cube.

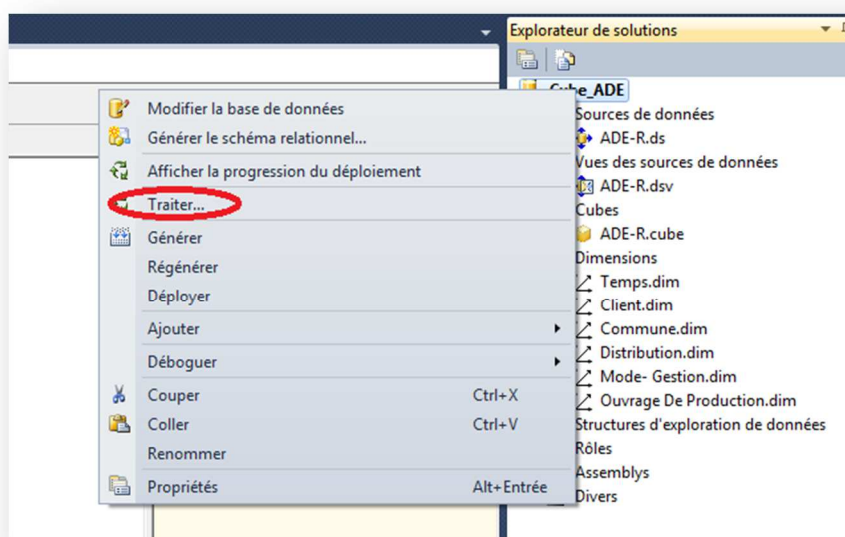


Figure 4.19 : le traitement du cube.

III.6 Connexion au serveur OLAP

Pour accéder à une instance Analysis Services il faut configurer un point de terminaison HTTP. On peut activer l'accès HTTP en configurant MSMDPUMP.dll, une extension ISAPI qui s'exécute dans Internet Information Services (IIS) et qui pompe des données entre des applications clientes et un serveur Analysis Services. Cette approche constitue une alternative à la connexion à Analysis Services lorsqu'une solution OLAP nécessite les capacités suivantes :

- Un accès client via une connexion Internet ou extranet.
- Une application cliente qui s'exécute dans un environnement réseau qui permet des connexions HTTP mais pas TCP/IP.
- Des méthodes d'authentification autres que la sécurité intégrée de Windows sont requises. IIS prend en charge les connexions anonymes et l'authentification de base.
- Des applications clientes qui ne peuvent pas utiliser les bibliothèques clientes Analysis Services (par exemple, une application Java dans notre cas). Si on ne peut pas utiliser les bibliothèques clientes Analysis Services pour accéder aux données, on peut utiliser SOAP et XML/A sur une connexion HTTP directe à une instance Analysis Services.

III.7 Créations de l'application

Cette étape consiste à créer un projet sur l'environnement java en utilisant Eclipse Luna. Pour des raisons d'utilisation des outils spécifiques à l'affichage de carte et des outils de « Business Intelligence », on aura besoin d'ajouter quelques JRE.

- **OLAP4j**

L'API, connu sous le nom olap4j, est l'équivalent OLAP de Java Data base Connectivité (JDBC) pour les données relationnelles. Plus précisément, olap4j étend classes de base de JDBC spécifications 3 et 4 afin d'apporter des sources de données OLAP pour la plate-forme Java. Les connexions peuvent être obtenues par le gestionnaire de connexion JDBC. L'API utilise des instructions, qui sont fournis par les connexions. Les requêtes, formulées en utilisant le langage MDX, peuvent être envoyés textuellement à la connexion.

L'API d'olap4j comprend également un système capable de représenter toute grammaire de requête MDX spécifique au serveur comme un modèle d'objet métier de type. Résultats de requêtes multidimensionnelles sont représentés par un objet. Cellset formable pour une meilleure représentation des résultats. [35]

- **ArcGIS Runtime SDK pour Java**

Le SDK ArcGIS Runtime pour Java fournit un ensemble d'outils de développement qui vous permettent de créer des applications ciblées de carte 2D pour répartition sur les plates-formes Windows et Linux. Le SDK vous permet

d'intégrer des cartes, localisateurs, et des modèles de géo-traitement en ligne et locales pour créer des applications riches et des SIG a haute fonctionnalités s qui tirent pleinement parti de la plate-forme Java Desktop.

Ce que le SDK nous donne :

- Un riche API Java SE fournie par une suite de fichiers jar.
- Un plug-in pour Eclipse, y compris les modèles de projets pour créer facilement de nouvelles applications.
- Un ensemble de composants d'exécution. Déployez uniquement les composants dont vous avez besoin l'aide du Générateur de déploiement.
- Une application de visualisation de l'échantillon interactif qui vous permet de visualiser les capacités du SDK et de voir code de l'application que vous pouvez utiliser pour créer vos propres applications.
- Une bibliothèque d'outils open-source qui comprend un ensemble de composants pour aider au développement rapide d'applications. La bibliothèque d'outils comprend les commandes suivantes:
 - Légende.
 - La barre d'échelle.
 - Navigator, y compris rotation de la carte.
 - Superpositions de gérer l'affichage des pop-ups, dessiner des graphiques, sélectionnant graphiques, et plus...
 - Curseur de temps pour voir les couches temporelles courant.
 - Composants d'édition d'entité, y compris un éditeur dispositif de fixation.
 - Carte générale.
 - Étendue signets.
 - Contrôles de visibilité des calques. [36]

III.7.1 Connexion à l'interface

Elle s'établie via une Connexion « http », une fois connecté au serveur, le gestionnaire nous montre toutes les bases de données disponibles sur le serveur et les cube des données dans chaque base comme montre la figure : (Voir figure 4.20)

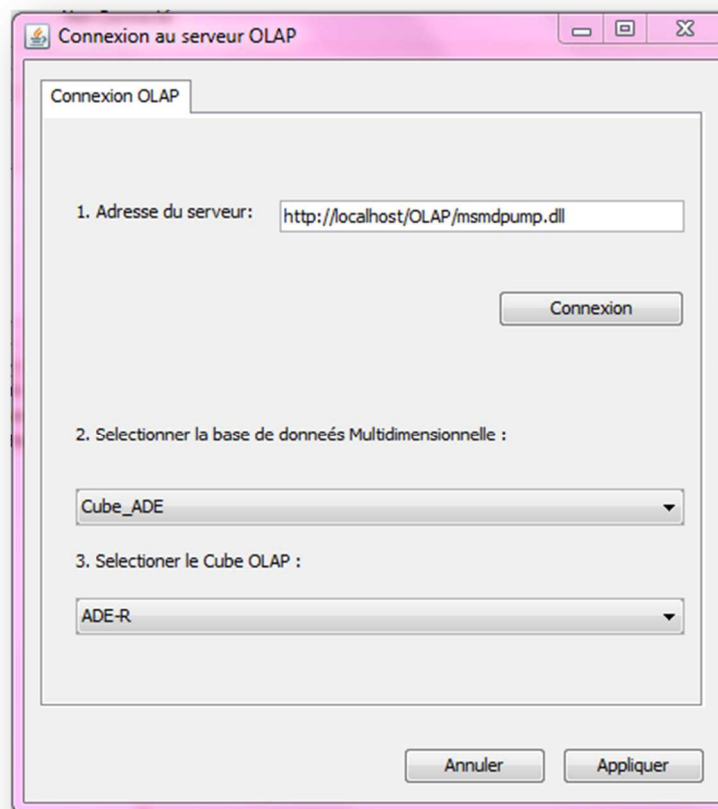


Figure 4.20 : Module de gestion de connexions.

III.7.2 L'interface utilisateur

Cette application a été créée pour être facile à utiliser et manipuler des différentes opérations, elle est composée de :

- ✓ Une zone affiche la carte et autre affiche des résultats.
- ✓ des différents composants qui permettent de choisir les paramètres des requêtes MDX tels que :
 - Choix de mesure (Indicateur).
 - Choix de dimension.

Elle offre aussi un contrôle de tous ces modules :

- ✓ Gestion de connexion.
- ✓ Supprimer les requêtes et faire des nouvelles.
- ✓ Ajouter des nouvelles couches graphiques.
- ✓ Gestion de visualisation des couches.
- ✓ Requêtes personnalisées.
- ✓ Différents modes de visualisation des résultats.

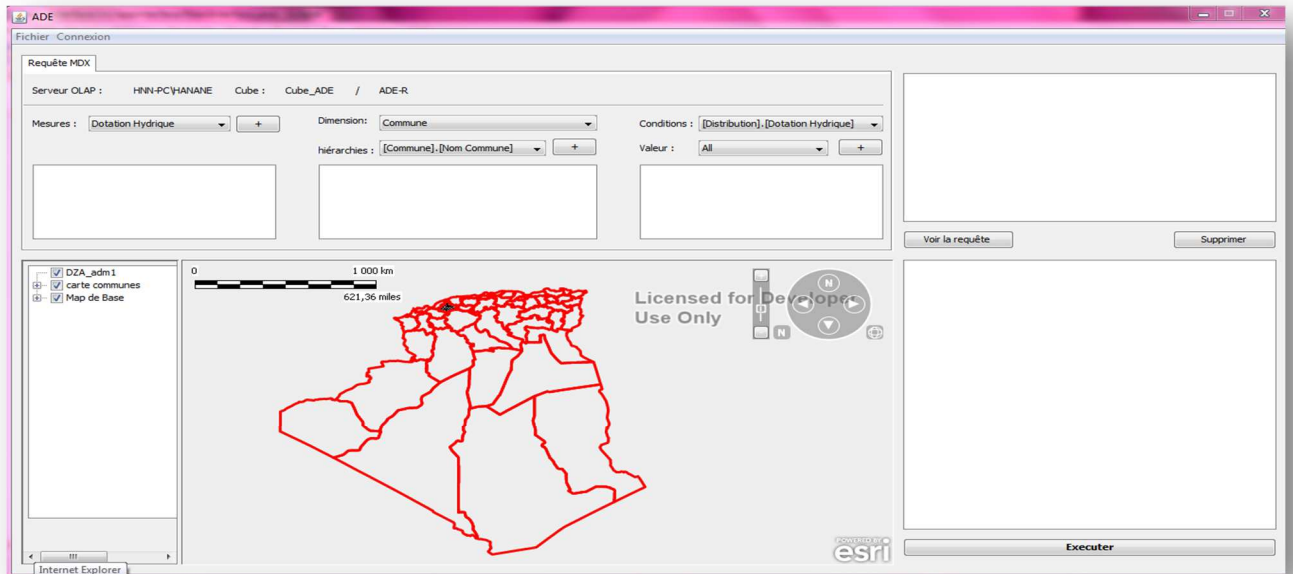


Figure 4.21 : L'interface utilisateur finale.

III.8 Exploitations de l'application

L'exploitation de l'application peut se faire en 2 manières :

- ✓ En utilisant les requêtes MDX à partir de SQL Server.
- ✓ En utilisant l'interface client.

III.8.1 Les requête MDX

Le langage MDX est un langage qui utilise des requêtes multidimensionnelles appliquées sur les cubes et remplace le langage SQL dans les bases de données relationnelles. Bien qu'il soit différent, il y a de fortes ressemblances avec le langage SQL.

Dans ce qui va suivre on va montrer quelques exemples des requêtes MDX et leurs résultats et on commençant par pointer sur notre cube puis on utilise la fonctionnalité (Requête MDX) de l'outil SSAS (SQL Server Analysis Services). Une fenêtre de saisie de la requête doit apparaître tout en assurant que notre cube (Cube_ADE) est sélectionné.

Exemple1

```
SELECT    {[Measures].[Dotation Hydrique]}    ON    COLUMNS,
          {[Temps].[Mois].Members} ON ROWS FROM [ADE-R].
```

The screenshot shows a SQL query window with the following query:

```
SELECT {[Measures].[Dotation Hydrique]} ON COLUMNS, {[Temps].[Mois].Members} ON ROWS FROM [ADE-R]
```

The results pane shows a table with the following data:

	Dotation Hydrique
All	3591
Decembre	770
Juin	813
MARS	725
Septembre	1283
Unknown	(NULL)

Figure 4.22 : Résultat de la première requête.

Exemple 2

SELECT {[Measures].[Volume], [Measures].[Dotation Hydrique] } ON COLUMNS, { [Commune].[ID Commune].members} ON ROWS FROM [ADE-R] WHERE ([Commune].[Nom Commune].&[Relizane]).

The screenshot shows a SQL query window with the following query:

```
SELECT {[Measures].[Volume], [Measures].[Dotation Hydrique] } ON COLUMNS, { [Commune].[ID Commune].members} ON ROWS FROM [ADE-R] WHERE ([Commune].[Nom Commune].&[Relizane])
```

The results pane shows a table with the following data:

	Volume	Dotation Hydrique
All	20400	334
1	19000	334
2	1400	(NULL)

Figure 4.23 : Résultat de la deuxième requête.

III.8.2 l'utilisation de l'interface client

L'utilisation de l'interface contient deux étapes principales :

- a) **Lancement des requêtes MDX:** la figure ci-dessous représente l'implémentation des requêtes MDX en choisissant les mesures, les dimensions nécessaires et les conditions s'ils existent : (Voir figure 4.24)

Figure 4.24 : Acquisition des requêtes MDX.

La requête montrée dans la figure précédente sera après affichée sous la forme suivante : (Voir figure 4.25)

```
SELECT NON EMPTY { [Measures].[Dotation Hydrique]
NON EMPTY { [Commune].[Nom Commune].members} ON F
WHERE ([Temps].[Trimestre].&[1er Trimestre]);
```

Voir la requête Supprimer

Figure 4.25 : Exemple d'une requête MDX.

b) Visualisation des résultats

Après l'exécution de la requête les résultats sont présentés sous deux formes :

1-Un tableau comme montre la figure suivante : (Voir figure 4.26)

		Dotation Hydrique
All		885
	A/Moussa	140
	Beni Dergoun	115
	El Matmar	130
	O/Djemaa	250
	Relizane	150
	S.M.Ben Ali	100

Executer

Figure 4.26 : Résultat de la requête précédente.

2-Une carte comme représente la figure ci-dessous : (Voir figure 4.27)

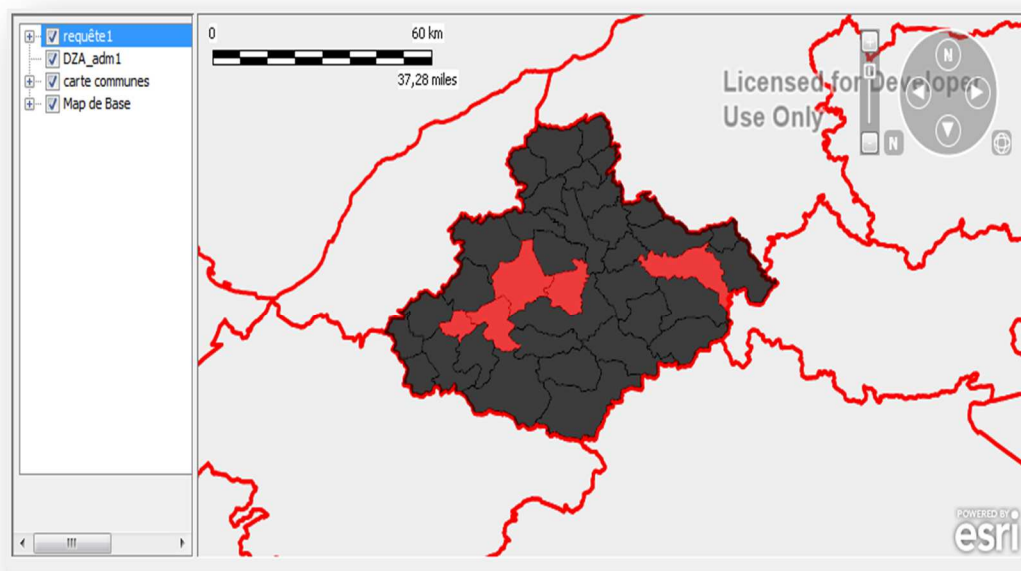


Figure 4.27 : Résultat de la même requête.

III.9 Vérification des contraintes d'intégrité

La Vérification des contraintes d'intégrité se fait grâce à l'outil OCL2SQL. C'est une solution libre (open source) qui permet de traduire les requêtes exprimées en OCL (Object Constraint Language) comme il est illustré dans la figure suivante : (Voir figure 4.28)

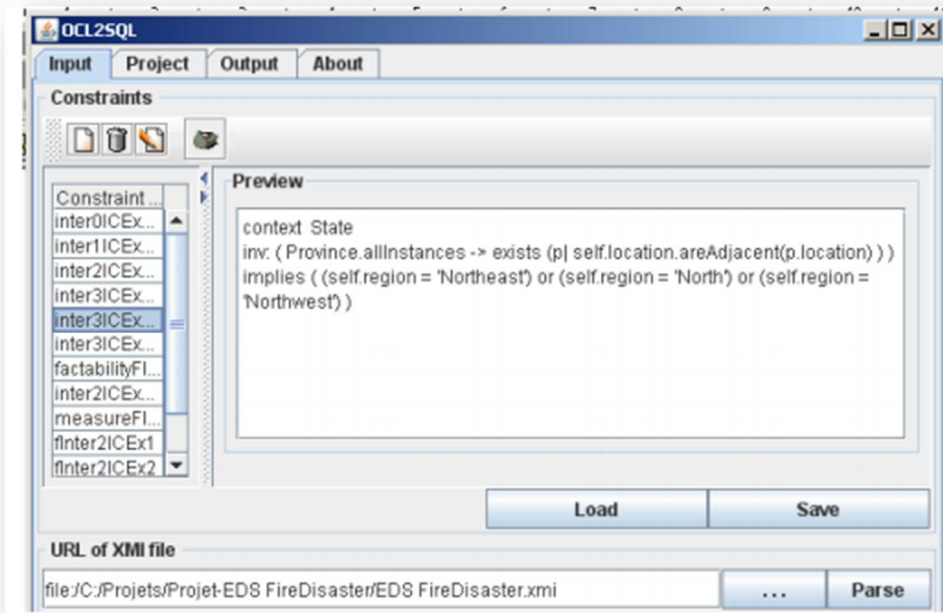


Figure 4.28 : fenêtre montre une requête OCL.

Vers un script SQL comme il est montré dans la figure suivante : (Voir figure 4.29)

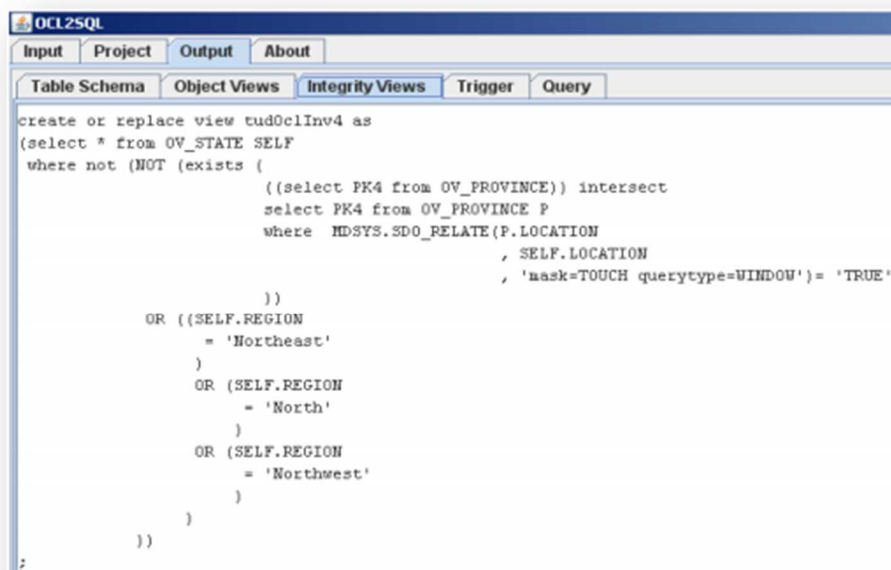


Figure 4.29 : requête OCL traduite en SQL.

En effet, la requête se traduit sous forme de déclencheur (trigger) qui s'exécute automatiquement lors de présence d'une erreur de manipulation dans le cube de donnée principalement dans l'application des opérateurs d'agrégation ou de mesures non consistante (non additivité par exemple).

Conclusion

Ce chapitre était consacré pour la modélisation et le développement de l'application décisionnelle SOLAP en se basant sur l'intégration du serveur OLAP fourni par SQL Server 2012 et le SIG réalisé sous ArcGIS. L'application cliente est développée sous l'environnement java (éclipse)

Conclusion générale

Les systèmes d'aide à la décision sont présents dans de nombreux domaines et destinés à faciliter la prise de décision au niveau de l'entreprise. Ces systèmes apportent des solutions nouvelles pour la modélisation, l'interrogation et la visualisation de données brutes, de documents, de connaissances personnelles et de modèles métier afin d'identifier et résoudre des incidents et prendre des décisions.

L'intégration des données spatiales dans l'entrepôt de données est un enjeu majeur nécessite le couplage de systèmes OLAP et de Systèmes d'Informations Géographiques au sein de systèmes OLAP Spatial (SOLAP) qui représente une solution efficace pour l'analyse spatiaux-multidimensionnelle d'informations géographiques.

Tout au long de notre travail nous avons pu nous introduire dans le domaine de l'informatique décisionnelle et aussi dans les entrepôts des données spatiales (EDS) et les systèmes SOLAP (Spatial OLAP), et nous avons vu tous les différents outils et concepts pour réaliser notre travail.

Ce travail nous a permis d'enrichir et d'approfondir nos connaissances à travers un stage que nous avons effectué au sein de la direction des eaux. Ça nous a aidés de collecter les informations nécessaires à la modélisation multidimensionnelle.

En effet, nous avons modélisé une solution SOLAP pour la direction des eaux de la ville de RELIZANE. Cette modélisation a été basée sur de différents logiciels pour développer une application décisionnelle tout en se focalisant sur la modélisation des CI des données spatiales.

Annexe

❖ Normalisation d'une base de données

On a 3 formes normales :

1) La première forme normale

Une relation est en première forme normale si tous ses attributs sont atomiques (Multivalué, composé).

2) La deuxième forme normale

Une relation est en deuxième forme normale si :

- ✓ Elle est en première forme normale.
- ✓ Tout attribut non clé dépend de la totalité de toutes les clés.

3) La troisième forme normale

Une relation est en troisième forme normale si :

- ✓ Elle est en deuxième forme normale.
- ✓ Il n'existe aucune DF entre attributs non clé.

❖ Le diagramme de classe

Le diagramme de Classe va permettre de représenter une vue statique du système d'information. Pas de dynamisme ici puisqu'on n'évoque pas les stimuli qui font réagir le SI, il s'agit plutôt des relations entre les Classes, des services rendus et utilisés par chacune d'elles et de l'articulation de l'ensemble. Ce diagramme sera souvent utilisé pour vous présenter des design pattern, car il montre bien la structure de la solution logicielle.