

## CHAPITRE 5

# Data Warehouse

Depuis l'avènement des premières versions des bases de données, ces dernières ont été centrées sur le traitement opérationnel, généralement transactionnel, répondant aux besoins d'une certaine communauté d'utilisateurs. Pendant ces dernières années, d'autres besoins se sont fait sentir auprès d'une autre communauté d'utilisateurs, généralement des décideurs dans leurs organisations, qui traitent avec l'information et l'analyse. Le traitement analytique examine de grande quantité de données pour détecter les tendances au lieu de regarder un ou deux enregistrements de données comme c'est le cas dans le traitement opérationnel. Dans ce contexte, la notion de Data Warehouse (DW) a émergé dans le sens de satisfaire aux besoins des décideurs dans le processus décisionnel. Dans un DW, toutes les informations, qu'elles que soient et où qu'elles soient, doivent être considérées et exploitées du moment qu'elles soient en relation avec l'organisation en question. C'est pour cela que, tous les modèles de BD (BDR, BDOO, BDRO, BDD) qu'on a déjà vus dans les chapitres précédents peuvent être utilisés à tous les niveaux d'un DW.

## 5.1. Motivation et objectifs des DWs

Beaucoup de besoins relatives à la gestion de l'information se font sentir auprès des acteurs des organisations (entreprises) et pour lesquelles cherchent des solutions. Parmi ces besoins on peut citer les suivants :

- Ils collectent beaucoup de données auxquelles ils ne peuvent pas y accéder.
- Ils ont besoin de voir ce qu'est important.
- Ils perdent beaucoup de temps à savoir qui a les bons chiffres au lieu de concentrer les efforts sur la prise de décision.
- Ils veulent utiliser des informations dans la prise de décision.

Ces besoins mènent tous à exiger des objectifs pour les DWs dont nous citons les plus importants d'entre eux (Kimball 2005) (Kimball 2013):

- **Accessibilité des informations :**

Le contenu du DW doit être compréhensible et la navigation dans son contenu doit être facile et rapide.

- **Cohérence des informations :**

Les informations étant rassemblées à partir de différentes sources, elles doivent être nettoyées et cohérentes dans le sens où si deux unités de mesure ont le même nom elles doivent alors signifier la même chose et si à l'inverse deux unités de mesure n'ont pas la même signification elles doivent porter des noms différents.

- **Adaptation au changement :**

Un DW doit être capable d'assumer un changement continu sans pour autant permettre un bouleversement des données ou des technologies existantes.

- **Présentation des informations en temps opportun :**

Un DW doit être capable de rendre les informations exploitables en un temps raisonnable pour aider les utilisateurs dans la prise de décision.

- **Sécurité des informations :**

Un DW est souvent amené à contenir des informations confidentielles de l'organisation ce qui lui inflige la responsabilité, non seulement, de sécuriser les données mais aussi d'offrir aux gestionnaires d'avoir une visibilité sur leurs utilisations.

- **Fiabilité des informations :**

Afin d'assumer le rôle d'un système d'aide à la décision, un DW doit avoir les bonnes données pour une meilleure décision.

## 5.2. Définition d'un Data Warehouse

Le Data Warehouse est une collection de données orientées sujet, intégrées, non volatiles, historisées, variant dans le temps pour le support d'un processus d'aide à la décision (Inmon 2005).

Nous expliquons dans ce qui suit ces caractéristiques (Vaisman 2014).

**Orientées sujets :** cela veut dire que le DW se concentre sur les besoins analytiques des différents secteurs d'une organisation.

Contrairement aux systèmes opérationnels qui sont orientés programmes, les Data Warehouses sont orientés sujets. Pour une compagnie d'assurance par exemple, les applications peuvent concerner le traitement de l'automobile, de la vie, de la santé et les accidents alors que les principaux sujets peuvent être les clients, les polices d'assurance, les primes et les réclamations.

**Intégrées :** signifie relier les données obtenues à partir de plusieurs systèmes opérationnels en résolvant les problèmes dus aux différences de définition et de contenu des données telles que les différences de format et de codification des données, les synonymes (champs avec des noms différents mais les mêmes données), homonymes (champs avec le même nom mais différentes significations), la multiplicité des occurrences.

**Non volatiles :** les données ne sont ni modifiables ni supprimables ce qui leur garantit une durée de vie beaucoup plus longue par rapport aux données au niveau opérationnel. Le seul changement provient du chargement de nouvelles données

**Variant dans le temps :** signifie que plusieurs valeurs sont enregistrées pour la même information ainsi que la date à laquelle les modifications ont été apportées à ces valeurs. Cela veut dire qu'un DW montre l'évolution à travers le temps et non seulement les données récentes.

## 5.3. L'approche Data Warehouse

L'approche du DW repose sur l'idée d'aller de la donnée à l'information. Répondre par exemple à la requête « Quelle impact pourra avoir la situation familiale d'un employé sur son travail supplémentaire? » va nous amener à chercher toutes les données relatives à la situation familiale pendant une période donnée ainsi que celle relatives au travail réalisé par cet employé pendant la même période dans les applications du système et d'essayer de réunir ces

données qui peuvent être dans différents modèles de BDs (BDR, BDOO, BDRO, BDD). Il faut ensuite intégrer ces données (c'est à dire, les nettoyer, les transformer, les combiner, en supprimer les doublons, les purger, les standardiser, mettre leurs dimensions en conformité) afin de générer des informations qui peuvent constituer une réponse à la requête.

Cependant, l'intégration n'est pas le seul obstacle qui complique le processus de la réponse à la requête. La non-historisation des données au niveau des applications du système constitue un autre obstacle à la satisfaction de la requête.

Afin de pouvoir répondre à une telle requête, il est nécessaire d'ajouter un autre type de données qu'est celui des données dérivées. Cela nous mène à distinguer deux types de données : primitives et dérivées.

Les données primitives représentent les données brutes générées au niveau opérationnel alors que les données dérivées représentent les informations générées par intégration au niveau décisionnel. Le tableau 6 montre quelques différences entre les données primitives et les données dérivées.

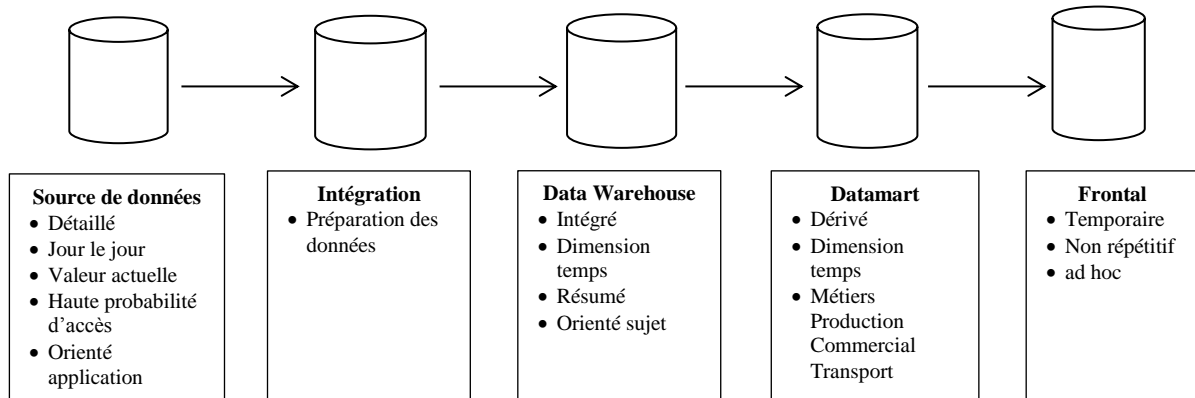
<b>Données primitives</b>	<b>Données dérivées</b>
<ul style="list-style-type: none"> <li>- Orientées application</li> <li>- Détaillées</li> <li>- Destinées à la communauté des exécuteurs</li> <li>- Mises à jour</li> <li>- Accessible à une unité à la fois</li> <li>- Sans redondance</li> <li>- Avec structure statiques et contenu variable</li> <li>- Haute probabilité d'accès</li> <li>- Actuelles</li> </ul>	<ul style="list-style-type: none"> <li>- Orientées sujet</li> <li>- Résumées ou calculées</li> <li>- Destinées à la communauté des décideurs</li> <li>- N'est pas mises à jour</li> <li>- Accessible à un ensemble à la fois</li> <li>- Avec redondance</li> <li>- Avec structure flexible</li> <li>- Faible probabilité d'accès</li> <li>- Historisées</li> </ul>

**Tableau 1** : Différences entre données primitives et données dérivées (Inmon 2005)

Les données primitives et les données dérivées sont tellement différentes (Tableau 6) qu'elles ne peuvent résider dans la même base de données. C'est la raison pour laquelle les données dérivées en plus des données intégrées et historisées sont déposées dans des bases de données séparées des données primitives.

La différence entre les données primitives et les données dérivées mène à considérer des niveaux de base pour les données allant des sources de données

jusqu'à l'exploitation des données. La figure 19 montre comment les données sont reprises à partir de sources de données et ensuite préparées pour être intégrées dans un Data Warehouse. Ce dernier, qui couvre toutes les données, peut être divisé en plusieurs sous-ensembles appelés Datamarts et finalement peut exploiter les données par différents outils tels qu'un système de requêtes.



**Figure 1 :** Niveaux des données d'un DW

La figure 20 montre un exemple sur les niveaux de données d'un employé dans un DW. Le niveau opérationnel contient les données actuelles de l'employé Mohamed Arbaoui dans plusieurs sources. Dans le niveau d'intégration, une préparation des données est réalisée. On peut remarquer par exemple que le prénom de l'employé est écrit en différentes façons (Muhamed, Mohamed et Mohammed) et que l'écriture « Mohamed » est adoptée pour l'intégration. La même chose est faite pour la situation familiale et le nombre d'enfants où respectivement les écritures « SF » et « 3 » sont adoptées. Dans le niveau Data Warehouse, on trouve plusieurs enregistrements sur Mohamed Arbaoui qui montre historiquement sa situation familiale (mariage et nombre d'enfants) sans que les données chevauchent. Si par exemple, la situation familiale de Mohamed Arbaoui change alors un nouvel enregistrement sera créé dans ce niveau. Dans le niveau Datamart (appelé aussi niveau départemental, ou niveau OLAP, ou encore niveau multidimensionnel) on trouve les données relatives à une fonctionnalité spécifique du système telle que les ressources humaines comme le cas de notre exemple où les données sont résumées par mois. Dans le niveau final, en l'occurrence le niveau frontal, les données sont temporaires répondant à une requête qui décèle les tendances sur un aspect donné. Dans notre exemple il est question de mesurer l'impact de la situation familiale sur le travail supplémentaire en heures réalisé par un employé.

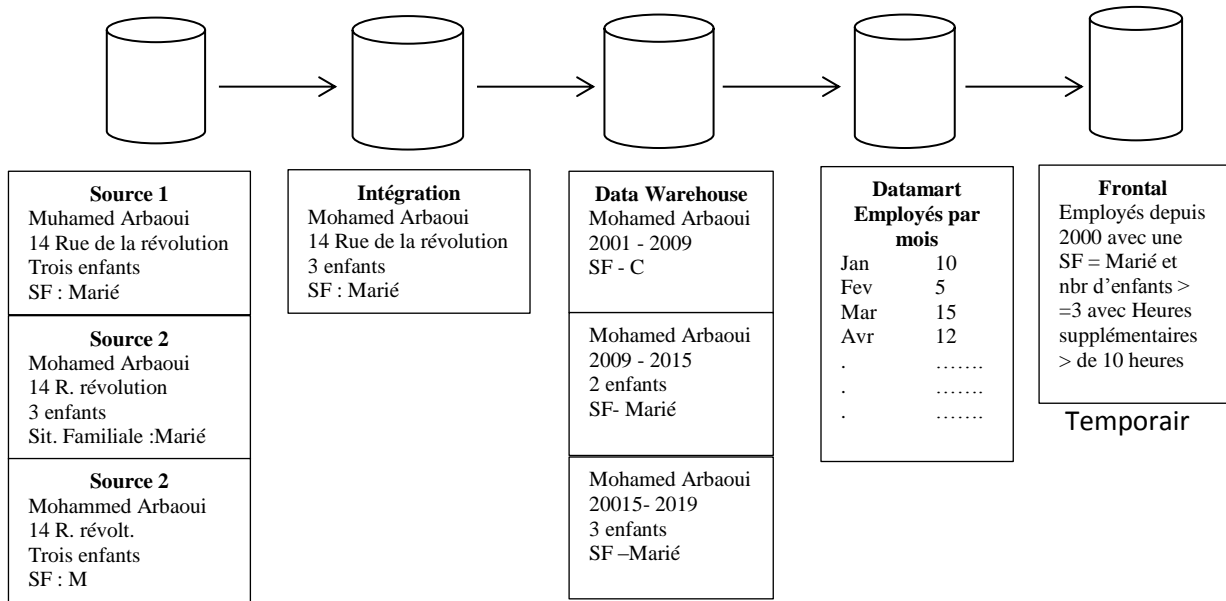


Figure 2 : Les niveaux de données pour un exemple simple : un employé

### 5.4. Architecture des DWs

Nous avons vu dans la section précédente 5.3 que l'approche DW est basée sur des concepts de base en l'occurrence : données primitives, données orientées sujets, historisées et variant dans le temps, intégration des données, données dérivées ainsi que la séparation entre ces derniers et celles dites primitives.

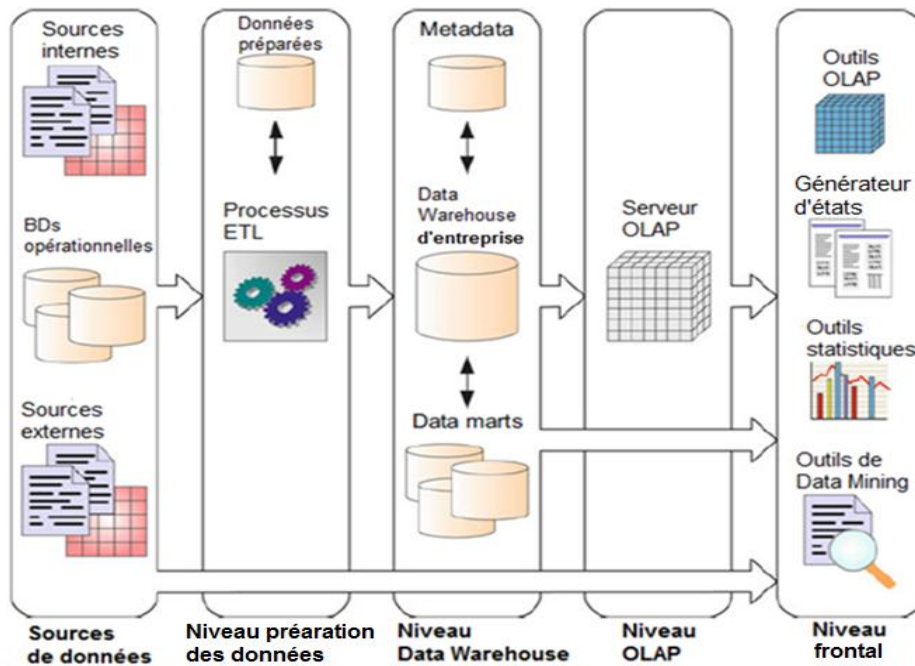


Figure 3: Architecture typique d'un DW (Vaisman 2014)

Sur la base de ces concepts, une architecture globale d'un DW est présentée dans la figure 21. L'architecture est constituée de plusieurs niveaux : le niveau préparation des données, le niveau data Warehouse, le niveau OLAP et finalement le niveau frontal.

#### **5.4.1. Le niveau préparation des données**

A ce niveau, les données sont extraites à partir des BDs opérationnelles qui peuvent être des BDRs, BDOOs, BDROs ou BDDs et d'autres sources de données (internes ou externes de l'organisation). Ces données sont ensuite intégrées dans le data Warehouse. La préparation des données est réalisée par les outils (ETL) : Extraction, Transformation et Loading (chargement). Une base de données intermédiaire, généralement relationnelle (BDR), est dédiée à l'exécution de l'extraction et la transformation des données avant leur chargement dans le Data Warehouse. De telle base de données est appelée opérationnel.

##### ***a) Extraction***

Les données sont extraites à partir de multiple sources de données hétérogènes, internes ou externes à l'organisation. Ces sources peuvent être opérationnelles ou des fichiers avec différentes formats.

##### ***b) Transformation***

Après l'extraction des données, plusieurs transformations peuvent être appliquée telles que le nettoyage (conversion en format standardisé et suppression des erreurs et des incohérences dans les données telles que les erreurs d'orthographe et les conflits de domaine), combinaison des données provenant de plusieurs sources et suppression des doublons.

##### ***c) Loading (chargement)***

Les données transformées sont alors chargées dans le DW. Cela inclus également la propagation des mises à jour des sources de données vers le DW à une fréquence qui peut être, selon la politique de l'organisation, mensuelle, plusieurs fois par jour ou presque en temps réel.

#### **5.4.2. Le niveau Data Warehouse**

A ce niveau, il y a trois composantes importantes à savoir : un Data Warehouse d'entreprise, un ou plusieurs Datamarts et une Metadata.

**Data Warehouse d'entreprise:** il est centralisé et couvre toute l'organisation

**Datamart:** par contre, le Datamart est spécifique à une fonction ou un département dans l'organisation. Le Datamart est chargé par des données (déjà nettoyées et intégrées) du Data Warehouse d'entreprise.

**Metadata:** les métadonnées sont définies comme des « données sur les données » ou « les données relatives aux données » et peuvent être classées en deux types :

(i) **métadonnées de métier** qui décrivent la sémantique des données, les règles organisationnelles et les contraintes relatives à ces données.

(ii) **métadonnées techniques** qui décrivent la manière avec laquelle les données sont structurées et stockées dans le système ainsi que les applications qui les manipulent.

### 5.4.3. Le niveau OLAP

Ce niveau est composé d'un serveur OLAP fournissant aux utilisateurs professionnels des données multidimensionnelles à partir du Data Warehouse ou des Datamarts.

### 5.4.4. Le niveau frontal

Comme montré dans la figure 21, afin de permettre aux clients d'exploiter les données du DW, le niveau interface leurs fournit les outils suivants :

- **Outils OLAP :** facilitent la formulation des requêtes ad hoc complexes impliquant de grandes quantités de données.
- **Générateurs d'états :** produisent des rapports en utilisant des requêtes prédéfinies.
- **Outils statistiques :** utilisent des méthodes statistiques dans l'analyse et la visualisation des données.
- **Outils Data Mining :** permettent d'analyser les données dans le but d'établir un modèle ou déceler les tendances afin d'aider à la prédiction.

## 5.5. Data Warehouse vs Datamart

Un Datamart peut être considéré comme un sous-ensemble d'un Data Warehouse. En effet, un DW couvre entièrement une organisation et contient



des données sur différents sujets alors qu'un Datamart contient des données d'un seul sujet. Un DW est chargé par des données qui proviennent de différentes sources de systèmes opérationnels alors qu'un Datamart est chargé par des données du DW lui-même (Jensen 2010).

Concernant la relation entre les DWs et les Datamart, certains considèrent qu'un DW est une collection de Datamarts ce qui veut dire qu'un DW est construit d'une manière ascendante « Bottom-up » à partir de petits Datamarts alors que le célèbre Inmon proclame l'inverse en considérant que la construction d'un DW est plutôt descendante « Top-down » où les Datamarts sont dérivés à partir des DWs (Inmon 1998).

## 5.6. Modèle multidimensionnel

Le modèle multidimensionnel est fondamental pour plusieurs systèmes d'aide à la décision tels que le Data Warehouse car le modèle multidimensionnel est bien adapté aux requêtes telles que celles citées par (Golfarelli 2009):

- quel est le montant total des recettes enregistrées l'an dernier par état et par catégorie de produit ?
- quelles commandes maximisent les recettes ?

Il est évident que le fait d'utiliser les langages traditionnels tels que SQL pour exprimer ces types de requêtes est une tâche difficile. Il est aussi évident que l'exécution de telles requêtes sur des BDs opérationnels prendra un temps de réponse inacceptable.

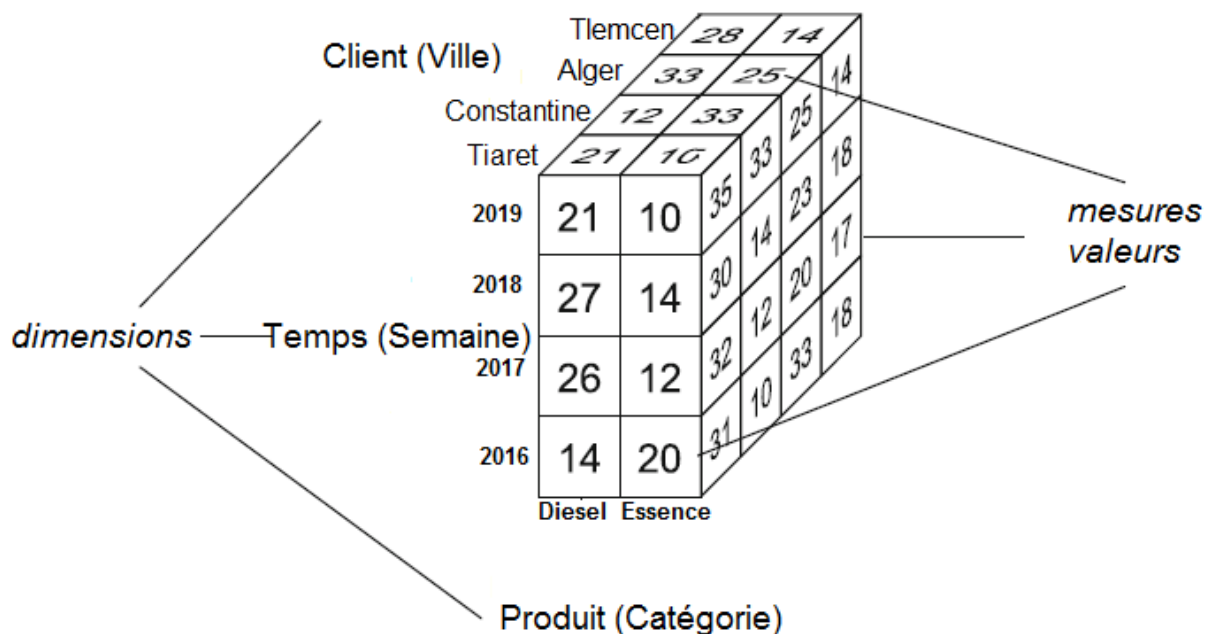
Effectivement, les SGBDs opérationnels ou OLTP ne peuvent assurer une analyse de données basée sur l'exécution de requêtes complexes nécessitant la jointure de beaucoup de tables et l'agrégation de larges volumes de données incluant aussi leurs historisation.

Ces besoins nécessitent ce qu'on appelle les systèmes OLAP qui sont orientés vers l'analyse de données dans le but d'aide à la décision. Les systèmes OLAP se concentrent sur les requêtes analytiques qui exigent l'agrégation, beaucoup de temps, la lecture transversale des enregistrements des tables et de nouvelles techniques d'indexation. Il est évident que le besoin à des solutions qui peuvent supporter ces techniques est plus que pressant. Le Data Warehouse est défini justement dans le but de répondre à ce besoin.

Data Warehouse et OLAP se basent sur le modèle multidimensionnel qui visualise les données dans un espace de n-dimensions appelé les données cube. Ces dernières sont définies par les faits et les dimensions. Dans ce qui suit nous définissons les concepts fondamentaux du modèle multidimensionnel à savoir : les cubes, les dimensions, les faits et les mesures (Jensen 2010) et nous abordons à la fin les principales opérations OLAP.

### 5.7.1. Cubes

Un cube est une structure de données multidimensionnelles définie dans le but de capturer et analyser des données. Un cube généralise la feuille de calcul d'un tableur avec une dimension supérieure à deux. Un cube peut avoir une dimension supérieure à trois contrairement à ce que son nom peut signifier. Pour cette raison le mot hypercube est parfois utilisé à la place de cube. Une collection de cubes constitue une base de données multidimensionnelle ou un Data Warehouse multidimensionnel.



**Figure 4:** Un cube en trois dimensions pour les données de vente avec les dimensions : *Produit, Temps et Client*, et une quantité de mesure

Un cube est composé de cellules où chacune d'elles représente une intersection de dimensions. Une cellule quand elle est non vide est appelée un fait et quand elle est vide cela signifie qu'il n'y a aucune information à enregistrer pour les valeurs de dimensions données. Dans la figure 22, il y a une seule mesure à savoir le nombre de ventes pour chaque fait qui représente une combinaison d'une année, de type d'énergie de voiture, et de ville. Par exemple,

le total de ventes dans la ville de ‘Tiaret’ en 2019 est 31 en additionnant les deux nombres 21 et 10.

### 5.7.2. Dimensions

Les dimensions sont utilisées pour la sélection des données et leur l’agrégation. Le niveau de la dimension est la granularité (le niveau du détail) à laquelle (auquel) les mesures représentent les dimensions. Dans l’exemple de la figure 22, les ventes sont agrégées aux niveaux *Catégorie*, *Semaine* et *Ville* respectivement pour les dimensions *Produit*, *Temps* et *Client*. Les instances d’une dimension sont appelées membres. Par exemple, *Diesel* et *Essence* sont des membres de la dimension *Produit* au niveau *Catégorie*. Les dimensions ont aussi des attributs associés qui les décrivent. Par exemple, la dimension *Produit* doit avoir les attributs *NumProduit* et *Prix* qui ne sont pas visible sur la figure 22.

Les dimensions sont organisées hiérarchiquement en niveaux de détail. Par exemple, pour la dimension *Produit*, la granularité la plus faible est *Produit*, qui est agrégée en *Catégorie* (Figure 23).

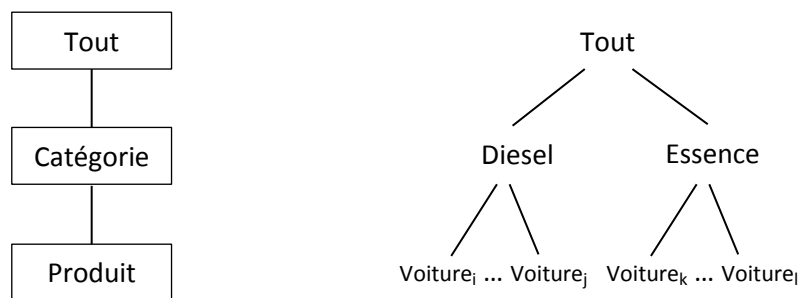


Figure 5: Hiérarchie de la dimension *Client*

La hiérarchie de la dimension est définie dans un Metadata au niveau du cube ou au niveau du modèle multidimensionnel, si la dimension peut être partagée.

### 5.7.3. Faits

Les faits sont les objets des sujets à être analysés dans le but de comprendre leurs comportements tels que les ventes dans l’exemple de la figure 22. Les faits sont définis par leur combinaison de valeurs des dimensions. Donc, les faits existent dans les cellules non vides. Dans la figure 22, chaque nombre affiché dans la cellule représente donc le nombre de voitures vendues par catégorie, par année et par ville du client.

#### 5.7.4. Mesures

Chaque mesure réalise une fonction d'agrégation qui combine plusieurs valeurs de mesures en une seule à un certain détail. Ainsi, une mesure a deux composants : une propriété numérique et une formule telle qu'une simple sommation (SUM). Dans la figure 22, la mesure *Client* fait une agrégation au niveau de la ville. Si on souhaite avoir une mesure au niveau du pays on doit faire une agrégation de toutes les ventes au niveau du pays en utilisant par exemple l'opération de sommation (SUM).

Cependant, il est indispensable de respecter certaines conditions qui régissent l'agrégation afin d'obtenir des résultats cohérents. Dans (Vaisman 2014), les conditions suivantes sont mentionnées:

- **Disjonction des Instances** : Dans un niveau donné il doit y avoir des ensembles d'instances disjoints. Par exemple, dans la figure 23, un produit ne doit pas figurer dans deux catégories différentes sinon la vente de ce produit sera calculée deux fois.
- **Complétude** : Toutes les instances doivent être liées dans la hiérarchie. Par exemple, dans la figure 23, la hiérarchie du *Produit* doit contenir toutes les voitures sinon le résultat d'agrégation sera incorrect.
- **Exactitude** : cette condition réfère à l'utilisation correcte de la fonction d'agrégation du fait que les mesures peuvent être de types différents. Cela mène à distinguer trois types de mesures: **(i) Mesures additives** qui peuvent être résumées de manière significatives en utilisant l'addition. La mesure de quantité dans le cube (Figure 22) est un exemple de mesure additive, **(ii) Mesures semi-additives** qui peuvent être résumées en utilisant l'addition pour quelques dimensions mais pas pour toutes les dimensions telles que l'exemple des quantités d'inventaire qui ne peuvent être agrégées pour deux périodes différentes, **(iii) Mesures non-additives** qui ne peuvent être résumées en utilisant l'addition pour aucune dimension telles que l'exemple du prix.

#### 5.7.5. Opérations OLAP

Le modèle multidimensionnel permet de visualiser les données à partir de plusieurs points de vue et plusieurs niveaux de détail en exploitant les dimensions et leurs hiérarchies ce qui offre un environnement interactif pour l'analyse des données. Nous détaillons dans ce qui suit deux importantes opérations OLAP à savoir Roll-up et Drill-up et résumons le reste dans le tableau 7.

L'opération Roll-up agrège les mesures le long d'une hiérarchie. Par exemple dans la figure 22, si on souhaite calculer la dimension *Client* au niveau du *Pays* on doit appliquer une opération Roll-up puisque le niveau *Pays* est plus élevé que le niveau *Ville*.

Par contre, L'opération Drill-down (l'opposée de l'opération Roll-up) désagrège les mesures le long d'une hiérarchie pour obtenir une granularité plus fine. Par exemple dans la figure 22, si on souhaite calculer la dimension *Temps* au niveau du *Jour* on doit appliquer donc une opération Drill-up puisque le niveau *Jour* est inférieur au niveau *Semaine*.

Opération	Objectif
Add mesure	Ajoute une nouvelle mesure à un cube calculée à partir d'autres mesures ou dimension
Opérations d'agrégation	Agrègent les cellules d'un cube
Dice	Conserve les cellules d'un cube qui satisfait une condition booléenne sur les niveaux de dimension, attributs ou mesures
Différence	Supprime les cellules d'un cube se trouvant dans un autre cube. Les deux cubes doivent avoir le même schéma
Drill-across	Fusionne deux cubes qui ont le même schéma et les mêmes instances à l'aide d'une jointure.
Drill-down	Désagrège des mesures le long d'une hiérarchie pour obtenir une granularité plus fine. Elle est l'opposée de Roll-up
Drill-through	Affiche les données des systèmes opérationnels à partir desquels le cube a été dérivé.
Drop mesure	Supprime des mesures d'un cube.
Pivot	Applique une rotation des axes d'un cube pour fournir une présentation différente de ses données.
Recursive roll-up	Applique récursivement une Roll-up le long d'une hiérarchie jusqu'à atteindre le niveau supérieur.
Rename	Renomme un ou plusieurs éléments de schéma d'un cube.
Roll-up	Agrège des mesures le long d'une hiérarchie. Elle est l'opposée de Drill-up
Roll-up*	Notation abrégée pour une séquence d'opérations de Roll-up.
Slice	Supprime une dimension d'un cube en fixant une valeur unique à un niveau de la dimension.
Sort	Ordonne les membres d'une dimension selon une expression.
Union	Combine les cellules de deux cubes qui ont le même schéma mais leurs membres sont disjoints.

**Tableau 2:** Résumé des opérations OLAP (Vaisman 2014)

## 5.8. ROLAP, MOLAP et HOLAP

Nous avons vu précédemment que l'analyse interactive des données dans un environnement DW est appelée On-line Analytical Processing (OLAP). Lorsque, les données d'un modèle dimensionnel sont stockées dans une base de données relationnelle, l'analyse est appelée ROLAP pour Relational On-Line Analytical Processing. Les systèmes ROLAP étendent le SQL pour prendre en charge le modèle dimensionnel et les fonctions avancées d'OLAP. Le fait de stocker les données dans des BDRs, cela a beaucoup d'avantages puisque les BDRs sont bien standardisées et offre une large capacité de stockage.

Lorsque les cubes de données d'un DW sont stockées dans des tableaux multidimensionnels, l'analyse de ces données est appelée MOLAP pour Multidimensional On-Line Analytical Processing.

Le principal avantage des MOLAPs par rapport aux ROLAPs est que les opérations multidimensionnelles peuvent être réalisées d'une façon facile et naturelle avec les MOLAPs sans aucune nécessité à de complexes opérations de jointure (Golfarelli 2009).

Quant aux systèmes HOLAP (Hybrid On-Line Analytical Processing), ils regroupent les avantages des technologies ROLAP et MOLAP en stockant de grandes volumes de données détaillées dans des serveurs ROLAP et les données agrégées dans des serveurs MOLAP (Liu 2009).

## 5.9. En résumé

Le Data Warehouse a émergé principalement suite à la nécessité des systèmes d'aide à la décision qui s'est manifestée auprès des décideurs des organisations. L'essentielle des objectifs d'un Data Warehouse a été dressé et une définition reflétant l'aspect décisionnel a été donnée. Une architecture en niveaux a été détaillée tout en réservant une attention spéciale au concept très important sur lequel sont basés les Data Warehouse à savoir le modèle dimensionnel.

## 5.10. Exercices

### Exercice 1 :

En reconsidérant l'exemple de la figure 22, répondez aux requêtes suivantes :

- a) Faite une opération de Roll-up au niveau pays de la dimension Client.
- b) Faite une opération de Drill-down au niveau mois de la dimension Temps.

**Exercice 2 :** <sup>(1)</sup>

Un entrepôt de données d'un opérateur téléphonique comprend cinq dimensions: client appelant, client appelé, heure, type d'appel et programme d'appel et trois mesures: nombre d'appels, durée et montant.

Définissez les opérations OLAP à effectuer pour répondre aux requêtes suivantes. Proposez les hiérarchies de dimensions en cas de besoin.

- a) Montant total collecté par chaque programme d'appel en 2019.
- b) Durée totale des appels passés par les clients Algérois en 2019.
- c) Nombre total d'appels en week-end passés par des clients de de la ville Tiaret à des clients de la ville d'Alger en 2019.
- d) Durée totale des appels internationaux lancés par des clients en Algérie en 2019.
- e) Montant total perçu auprès des clients Algérois inscrits au programme d'entreprise en 2019.

**Exercice 3 :**

On suppose que la compagnie aérienne AIR ALGERIE a construit un Data Warehouse contenant des informations sur les vols de ses avions. Ce dernier comprend six dimensions, à savoir l'aéroport de départ, l'aéroport d'arrivée, le vol, l'avion, l'heure d'arrivée et l'heure de départ, et trois mesures, à savoir le nombre de passagers, la durée et la distance en kilomètres.

Définissez les opérations OLAP à effectuer ainsi que les hiérarchies de dimensions nécessaires en répondant aux requêtes suivantes.

---

<sup>1</sup> Cet exercice est adapté à partir d'un exercice proposé dans (Vaisman 2014)

- a) Nombre total de kilomètres parcourus par les vols d’Air Algérie en 2019.
- b) Nombre total de voyages au départ ou à destination de Bruxelles en 2019.
- c) Nombre de passagers des vols Alger-Bruxelles ou Bruxelles-Alger durant 2019.
- d) La durée totale des vols Alger-Bruxelles ou Bruxelles-Alger durant 2019.

### 5.11. Références du chapitre

- [Golfarelli 2009] M. Golfarelli, S. Rezi, Data Warehouse design, Modern principles and methodologies, McGraw-Hill companies, (2009).
- [Inmon 1998] W. H. Inmon, Data Mart Does Not Equal DataWarehouse, DMReview, (1998).
- [Inmon 2005] W.H. Inmon, Building the Data Warehouse, 4ème Edition, Wiley Publishing, Inc, ISBN-13: 978-0-7645-9944-6, (2005).
- [Jensen 2010] C.S. Jensen, T.B. Pedersen, C. Thomsen, Multidimensional Databases and Data Warehousing, Morgan & Claypool Publishers, DOI 10.2200/S00299ED1V01Y201009DTM009, (2010).
- [Kimball 2005] R. Kimball, L. Reeves, M. Ross, W. Thornthwaite, Le data warehouse: Guide de conduite de projet, Edition Eyrolles, ISBN:2-212-11600-4, (2005).
- [Kimball 2013] R. Kimball, M. Ross, The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling, 3ème Edition, John Wiley & Sons, Inc, (2013).
- [Liu 2009] L. Liu, M.T. Özsu (Eds.) Encyclopedia of Database Systems, Springer Science+Business Media, LLC 2009 (USA), ISBN: 978-0-387-35544-3, (2009).
- [Vaisman 2014] A. Vaisman and E. Zimanyi, Data Warehouse Systems, Data-Centric Systems and Applications, DOI:10.1007/978-3-642-54655-6 3, Springer-Verlag Berlin Heidelberg, (2014).



