Agrégations multiples différentiées dans les bases de données multidimensionnelles

Ali Hassan*, Franck Ravat*, **Olivier Teste****, **Ronan Tournie***, **Gilles Zurfluh***

* *Université Toulouse 1 Capitole – IRIT (UMR 5505)*
** *Université Toulouse 3 Paul Sabatier – IRIT (UMR 5505)*
118, Route de Narbonne – 31062 Toulouse cedex 9
{hassan, ravat, teste, tournier, zurfluh}@ irit.fr

RÉSUMÉ. *De nombreux modèles ont été proposés pour la modélisation de données multidimensionnelles dans les entrepôts. Ces propositions considèrent une même fonction d'agrégation pour déterminer les valeurs d'une mesure aux différents niveaux de granularité de l'espace multidimensionnel. Nous proposons un nouveau modèle conceptuel plus flexible supportant des agrégations multiples différentiées. L'agrégation multiple permet d'associer à une même mesure, une fonction d'agrégation différente pour chaque axe d'analyse. L'agrégation différentiée autorise des agrégations spécifiques à chaque niveau de granularité. Le modèle proposé repose sur des formalismes graphiques suffisamment expressifs pour contrôler la validité des fonctions d'agrégation qui peuvent être distributives, algébriques ou holistiques. Nous montrons également comment la modélisation conceptuelle peut être exploitée au niveau logique R-OLAP pour construire efficacement des treillis de pré-agrégats.*

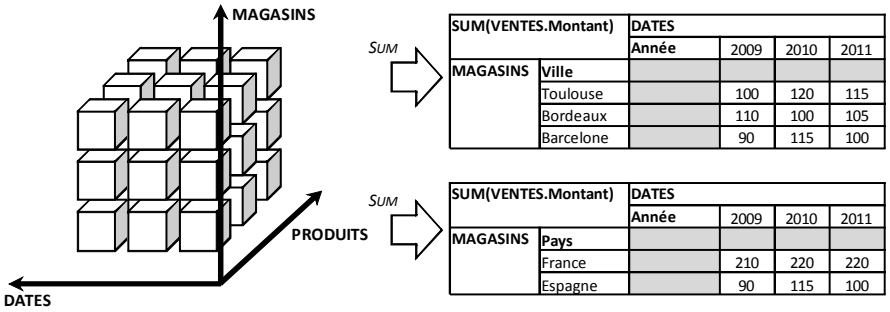
ABSTRACT. *Many models have been proposed for multidimensional data warehouses modeling. These propositions consider the same aggregate function to determine the values of a measure with different levels of granularity of the multidimensional space. We propose a new conceptual model for multidimensional representation of data supporting multiple differentiated aggregations. Multiple aggregation allows to associate to the same measure, a different aggregation function for each axis of analysis. Differentiated aggregation allows specific aggregations at each level of granularity. The proposed model is based on graphical formalisms expressive enough to control the validity of aggregate functions that can be distributive, algebraic and holistic. We also show how conceptual modeling can be exploited in logic R-OLAP to build effectively lattices of pre-aggregates.*

MOTS-CLÉS : *système décisionnels, base de données multidimensionnelles, modélisation conceptuelle d'entrepôt de données, mécanismes d'agrégations multiples dans les treillis multidimensionnels.*

KEYWORDS: *decision-making system, multidimensional database, conceptual modeling of data warehouse, mechanisms of multiple aggregations in multidimensional lattices.*

1. Introduction

Les systèmes d'information d'aide à la prise de décision ont montré leur capacité à intégrer de larges volumes de données tout en supportant efficacement des analyses sur les données entreposées. Ces systèmes décisionnels sont élaborés à partir de sources de données, généralement provenant du système opérationnel d'une organisation ; les données identifiées pertinentes dans les sources sont extraites, transformées, puis chargées (Vassiliadis *et al.,* 2002) dans un espace de stockage appelé entrepôt de données (« data warehouse »). Afin de rendre efficace l'interrogation et l'analyse de ces données entreposées, des techniques d'organisation des données spécifiques ont été développées (Kimball, 1996) reposant sur des bases de données multidimensionnelles (BDM). Ce type de modélisation considère la donnée à analyser comme un point dans un espace à plusieurs dimensions, formant ainsi un cube de données (Gray *et al.,* 1996). Les décideurs visualisent un extrait des cubes de données, généralement une tranche à deux dimensions (Gyssens *et al.,* 1997). A partir de cette structure, appelée table multidimensionnelle (TM), le décideur peut interagir au travers d'opérations de manipulation (Ravat *et al.,* 2007). Les opérations les plus emblématiques sont les forages qui consistent à modifier le niveau de granularité des données observées et les opérations de rotation qui consistent à changer de tranche du cube manipulé. On parle d'analyse en ligne ou encore de processus OLAP (« On-Line Analytic Processing »).



SUM(VENTES.Montant)		DATES			
		Année	2009	2010	2011
MAGASINS	Ville				
	Toulouse		100	120	115
	Bordeaux		110	100	105
	Barcelone		90	115	100

SUM(VENTES.Montant)		DATES			
		Année	2009	2010	2011
MAGASINS	Pays				
	France		210	220	220
	Espagne		90	115	100

Figure 1. *Agrégation uniforme dans les tranches de cube.*

Cet environnement offre un cadre adéquat aux analyses des décideurs, cependant les structures de données imposées peuvent s'avérer imparfaites. En particulier, lors d'une analyse, une BDM classique supporte des calculs d'agrégation uniforme réalisés à partir de la même fonction d'agrégation dans les différentes tranches de cube. Par exemple, si l'on considère des montants de ventes, ces derniers peuvent être calculés en effectuant la somme des produits vendus en fonction des villes et des années. Le calcul de ces mêmes montants de vente en fonction des pays est généralement réalisé avec la même fonction d'agrégation (la somme SUM) comme

l'illustre la figure 1. Dès lors que l'utilisateur souhaite changer les fonctions d'agrégation entre deux tranches de cube manipulées, les BDM classiques ne garantissent plus la validité des données calculées, voire ne supportent pas ce type de manipulation.

L'objectif de cette recherche est de rendre possible des agrégations non uniformes lors de la manipulation et de garantir leur validité en proposant un modèle multidimensionnel permettant de supporter des ***agrégations qualifiées de multiples différentiées***. Notre proposition vise à développer un modèle multidimensionnel suffisamment flexible pour autoriser la conception des cubes intégrant différentes fonctions d'agrégation suivant les niveaux d'agrégation.

1.1. *Illustration du problème*

Pour illustrer nos propos, nous utiliserons le cas d'un jury de délibération des diplômes. Dans cet exemple simple, les décideurs (enseignants membres du jury) délivrent les diplômes en analysant les moyennes des étudiants. Nous considérons que l'année universitaire se compose de deux semestres. Chaque semestre comprend des unités d'enseignement (UE). Chaque UE se compose de matières. Chaque matière est associée à un coefficient qui représente l'importance de la matière dans l'UE, qui elle-même est liée à une valeur de crédit (ECTS). Chaque semestre accumule le même total d'ECTS. Les étudiants sont organisés dans des groupes.

Une BDM est mise en place par des processus d'extraction, de transformation et de chargement des données issues du système opérationnel que nous ne détaillerons pas dans cet article. La figure suivante décrit conceptuellement le schéma en étoile de cette BDM (Ravat *et al.*, 2008).

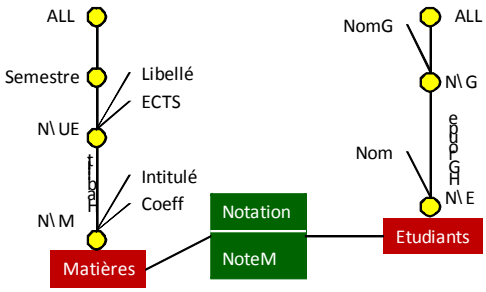


Figure 2. *La BDM de l'exemple du jury des diplômes.*

Cette BDM vise à analyser les notes (mesure) en fonction de chaque matière et de chaque étudiant (dimensions). Une matière est caractérisée par un numéro de

matière (N°M), un numéro d'unité d'enseignement (N°UE) et un Semestre. Chaque étudiant est caractérisé par un numéro (N°E), un Nom et un groupe (N°G et NomG).

Ce schéma de BDM permet par exemple d'obtenir directement la moyenne d'un étudiant par matière.

AVG(NOTATION.NoteM)		MATIERES			
		Semetsre	S1		
		N\ UE	U1		U2
		N\ M	M1	M2	M3
ETUDIANTS	N\ E (Nom)				
	E1 (Martin)		14	10	12
	E2 (Duval)		8	10	9

Figure 3. *TM visualisant la moyenne des étudiants par matière.*

Pour obtenir la moyenne par UE dans cet environnement multidimensionnel classique, il suffit d'agréger les moyennes par matières conformément à la fonction d'agrégation associée à la mesure NoteM (AVG). Or une telle opération donne un résultat incorrect compte tenu des modalités d'examens. En effet, la moyenne par UE est calculée à partir des notes des matières en tenant compte du coefficient de chaque matière (1). De manière analogue, pour obtenir la moyenne par semestre, l'application de la fonction d'agrégation prévue est inappropriée puisque cette moyenne est calculée en prenant en compte les crédits (ECTS) de chaque UE (2).

$$Moyenne\_UE = \frac{\sum Note * Coeff}{\sum Coeff} \quad (1)$$

$$Moyenne\_Semestre = \frac{\sum Moyenne\_UE * ECTS}{\sum ECTS} = \frac{\sum \left(\frac{\sum Note * Coeff}{\sum Coeff} \right) * ECTS}{\sum ECTS} \quad (2)$$

Les approches classiques qui considèrent une fonction d'agrégation unique pour tous les niveaux d'agrégation modélisés dans le schéma en étoile souffrent donc de plusieurs limites :

– **la variabilité de la fonction d'agrégation.** Le modèle traditionnel ne donne pas la possibilité d'utiliser des fonctions d'agrégation évoluant avec les niveaux des hiérarchies ou avec les dimensions. Dans l'exemple du jury de diplômes, la fonction d'agrégation des moyennes change avec les niveaux de hiérarchie entre N°M, N°UE et Semestre.

– **les lacunes des fonctions de base.** Nous remarquons que, pour agréger les données entre les niveaux de hiérarchie, nous utilisons des fonctions d'agrégation non-standards qui utilisent des données autres que les valeurs de la mesure (coefficients *Coeff*, poids *ECTS*).

– **les contraintes des agrégations.** Dans la littérature (Gray *et al.*, 1996), les fonctions d'agrégation appartiennent à trois catégories différentes. La première correspond aux fonctions **distributives** qui calculent les valeurs agrégées à un niveau de granularité à partir des valeurs déjà agrégées au niveau de granularité directement inférieur (par exemple, la somme - SUM - d'un montant par année peut se calculer à partir de la somme des montants par semestre). La deuxième correspond aux fonctions **algébriques** qui calculent les valeurs agrégées à partir de résultats intermédiaires stockés (par exemple, la moyenne - AVG - d'un montant par année peut se calculer à partir de la somme - SUM - des montants et du nombre - COUNT - des occurrences). Enfin, la troisième correspond aux fonctions **holistiques** qui ne peuvent pas être calculées à partir de résultats intermédiaires. Dans ce cas, il faut calculer les valeurs agrégées à partir des valeurs élémentaires correspondant au niveau de granularité le plus bas (par exemple, RANK). Outre ces catégories de fonctions, des contraintes sur la manière d'opérer le calcul de l'agrégation distributive ou algébrique peuvent exister. Dans notre exemple, la moyenne par semestre est nécessairement calculée à partir du calcul de la moyenne par UE comme l'illustre la première expression de la formule (2).

Notre objectif est donc de proposer un modèle multidimensionnel suffisamment expressif pour supporter ce type d'agrégations.

1.2. *Positionnement et contributions des travaux*

Il existe classiquement deux approches pour la modélisation des BDM : une approche reposant sur la métaphore du cube de données suivant laquelle la BDM est représentée par des cubes, et une approche dite de modélisation multidimensionnelle où la BDM est décrite par un schéma en étoile ou en constellation (Kimball, 1996). Nos travaux s'inscrivent dans cette seconde approche. En effet, la métaphore du cube repose sur une séparation équivoque entre les éléments de structure et les valeurs (Torlone, 2003) : modélisation des axes de l'analyse peu expressive notamment en raison de la difficulté à représenter l'organisation hiérarchique des données. Elle s'avère également limitée lorsqu'il s'agit de représenter des constellations de faits et de dimensions partagées.

Plusieurs synthèses du domaine (Chaudhuri *et al.,* 1997), (Vassiliadis *et al.,* 1999), (Mazón *et al.,* 2009) et d'études comparatives (Pedersen *et al.,* 2001), (Abelló *et al*., 2006), (Lujàn-Mora *et al.,* 2006), (Ravat *et al.*, 2008), (Oliveira *et al*., 2011), (Jaecksch *et al.,* 2011) sont disponibles dans la littérature scientifique. La plupart des propositions existantes considèrent qu'une mesure est associée à une fonction d'agrégation qui sera utilisée à tous les niveaux d'agrégation. Cette fonction calcule la même agrégation pour toutes les combinaisons de tous les paramètres modélisés.

Parmi les modèles existants, le modèle YAM2 (Abelló *et al*., 2006) est le seul modèle qui permet d'utiliser une fonction d'agrégation différente pour chaque dimension. Néanmoins ce modèle ne donne pas la possibilité de faire évoluer la

fonction avec les niveaux de hiérarchies. Dans les travaux de (Pedersen *et al.,* 2001), on peut lier à une seule mesure plusieurs fonctions d'agrégation mais chaque fonction est utilisée pour toutes les dimensions et tous les niveaux des hiérarchies.

En ce qui concerne les outils commerciaux, « Business Objects » utilise une seule fonction d'agrégation pour une mesure. En revanche, l'outil « Analysis Services de Microsoft » offre la possibilité d'appliquer un « forage personnalisé » à une hiérarchie de plusieurs façons (Harinath *et al.,* 2009) :

– par l'utilisation des opérateurs unaires qui sont utilisés pour résoudre le problème de l'agrégation sur un type particulier de hiérarchie (une hiérarchie d'attribut parent-enfant). Les hiérarchies parent-enfant sont construites à partir d'un seul attribut parent. Un attribut parent décrit une relation de jointure réflexive dans une table de dimension principale.

– par l'utilisation de scripts MDX.

– par l'utilisation d'une propriété CustomRollupColumn qui indique à une colonne où sont stockés les scripts MDX.

Les trois façons représentent des fonctions d'agrégation mais elles ne sont liées ni à une dimension ni à un niveau d'agrégation. Elles sont liées à un membre (une instance) d'un niveau d'agrégation d'une hiérarchie, c'est-à-dire, à une ligne dans la table de la dimension. Donc, pour appliquer ce « forage personnalisé » à un seul niveau d'agrégation il faut le répéter pour toutes les instances de ce niveau. Cela pose un problème de stockage et diminue la performance (Harinath *et al.,* 2009). D'un autre coté, la liaison de « forage personnalisé » avec une instance spécifique peut entraîner des difficultés en ce qui concerne la mise-à-jour des données.

Notre objectif est de lever ces limites en développant un modèle conceptuel de représentation des agrégations multidimensionnelles multiples différentiées. Par *multiples* nous signifions qu'une même mesure peut être agrégée selon plusieurs fonctions d'agrégation et par *différentiées* nous indiquons que ces agrégations peuvent varier en fonction du niveau d'agrégation choisi.

En plus de la classification des fonctions d'agrégation (Gray *et al.*, 1996) distributives, algébriques et holistiques, il existe d'autres classifications :

– du point de vue de la « Summarizability », les fonctions d'agrégation sont classifiées selon deux groupes (Abelló *et al*., 2006), (1) « Transitive » qui garantit la « Summarizability », (2) « Non-Transitive » qui implique que l'agrégation doit toujours se calculer à partir du niveau de base.

– du point de vue de la mesure (données), les fonctions d'agrégation sont de trois types (Pedersen *et al.,* 2001) : (1) applicables aux données additives, (2) applicables aux données qui peuvent être utilisées pour les calculs de moyenne, (3) applicables aux données constantes, c'est-à-dire qu'elles ne peuvent être que dénombrées.

Toutes ces propositions et les classifications des fonctions d'agrégation existantes estiment que l'on peut calculer l'agrégation d'une mesure à partir du

niveau de base. Notre but est d'ajouter le moyen de traiter le cas contraire, quand on ne peut pas agréger la mesure à partir du niveau de base, en utilisant des *contraintes d'agrégation*.

1.3. *Organisation de l'article*

La section 2 présente le modèle conceptuel multidimensionnel classique avant de présenter nos extensions pour les agrégations multiples différentiées. Ensuite nous présentons le formalisme graphique de ces extensions. La section 3 décrit le modèle logique en étoile R-OLAP avec ses relations d'optimisation. Nous présentons nos expérimentations dans la section 4.

2. Modèle conceptuel de données

2.1. *Concepts classiques*

Soient $\mathcal{N} = \{n_1, n_2,...\}$ un ensemble fini de noms non redondants, $F = \{F_1,..., F_n\}$ un ensemble fini de faits, $n \geq 1$, $D = \{D_1,..., D_m\}$ un ensemble fini de dimensions, $m \geq 2$.

DEFINITION 1. — Un *fait*, noté F_i, $\forall\, i \in [1..n]$, est défini par (n^{Fi}, M^{Fi}).

– $n^{Fi} \in \mathcal{N}$ est le nom identifiant le fait,

– $M^{Fi} = \{m_1,..., m_{pi}\}$ est un ensemble de *mesures*.

On pose $M = \bigcup_{i=1}^{n} M^{F_i}$

DEFINITION 2. — Une *dimension*, notée D_i, $\forall\, i \in [1..m]$, est définie par (n^{Di}, A^{Di}, H^{Di}).

– $n^{Di} \in \mathcal{N}$ est le nom identifiant la dimension,

– $A^{Di} = \{ a_1^{D_i},..., a_{r_i}^{D_i} \}$ est l'ensemble des *attributs de la dimension*,

– $H^{Di} = \{ H_1^{D_i},..., H_{s_i}^{D_i} \}$ est un ensemble de *hiérarchies*.

Les hiérarchies organisent les attributs d'une dimension, appelés paramètres, de la graduation la plus fine jusqu'à la graduation la plus générale. Ainsi une hiérarchie définit les chemins de navigation valides sur un axe d'analyse.

On pose $A = \bigcup_{i=1}^{m} A^{D_i}$ et $H = \bigcup_{i=1}^{m} H^{D_i}$

DEFINITION 3. — Une *hiérarchie*, notée H_j (notation abusive de $H_j^{D_i}$, $\forall i \in [1..m]$, $\forall j \in [1..s_i]$), est définie par $(n^{Hj}, P^{Hj}, \prec^{Hj}, Weak^{Hj})$.

– $n^{Hj} \in \mathcal{N}$ est le nom identifiant la hiérarchie,

– $P^{Hj} = \{ p_1^{H_j}, ..., p_{q_j}^{H_j} \}$ est un ensemble d'attributs de la dimension appelés *paramètres*, $P^{Hj} \subseteq A^{Di}$,

– $\prec^{Hj} = \{(p^{Hj}_x, p^{Hj}_y) \mid p^{Hj}_x \in P^{Hj} \wedge p^{Hj}_y \in P^{Hj} \}$ est une relation binaire antisymétrique et transitive. Rappelons que l'antisymétrie signifie que $(p^{Hj}_{k1} \prec^{Hj} p^{Hj}_{k2}) \wedge (p^{Hj}_{k2} \prec^{Hj} p^{Hj}_{k1}) \Rightarrow p^{Hj}_{k1} = p^{Hj}_{k2}$ tandis que la transitivité signifie que $(p^{Hj}_{k1} \prec^{Hj} p^{Hj}_{k2}) \wedge (p^{Hj}_{k2} \prec^{Hj} p^{Hj}_{k3}) \Rightarrow p^{Hj}_{k1} \prec^{Hj} p^{Hj}_{k3}$.

– $Weak^{Hj} : P^{Hj} \rightarrow 2^{A^{Di} \setminus P^{Hj}}$ est une application qui associe à chaque paramètre un ensemble d'attributs de dimension, appelés *attributs faibles*.

On pose $P^{D_i} = \bigcup_{j=1}^{Si} P^{H_j}$ et $P = \bigcup_{i=1}^{m} P^{D_i} = \bigcup_{i=1}^{m} \bigcup_{j=1}^{Si} P^{H_j}$

LEMME 1. — Pour chaque dimension D_i, un *paramètre racine*, noté $Id^{Di} \in P^{Di}$, existe. Il est défini comme suit. $\forall j \in [1..s_i], \forall p_k^{H_j} \in P^{Di}, Id^{Di} \neq p_k^{H_j} \mid Id^{Di} \prec^{Hj} p_k^{H_j}$.

LEMME 2. — Pour chaque dimension D_i, un *paramètre extrémité*, noté $All^{Di} \in P^{Di}$, existes. Il est défini comme suit. $\forall j \in [1..s_i], \forall p_k^{H_j} \in P^{Di}, All^{Di} \neq p_k^{H_j} \mid p_k^{H_j} \prec^{Hj} All^{Di}$.

On pose $W^{D_i} = \bigcup_{\forall j \in [1..s_i], \forall k \in [1..q_j]} Weak^{H_j}(p_k^{H_j})$

et $W = \bigcup_{i=1}^{m} W^{D_i} = \bigcup_{i=1}^{m} \bigcup_{\forall j \in [1..s_i], \forall k \in [1..q_j]} Weak^{H_j}(p_k^{H_j})$

LEMME 3. — Pour chaque dimension D_i, ses attributs de dimension sont de manière exclusive soit des paramètres, soit des attributs faibles, $P^{Di} \cap W^{Di} = \varnothing$ et $P^{Di} \cup W^{Di} = A^{Di}$.

2.2. *Extensions pour les agrégations multiples différentiées*

Afin que le modèle multidimensionnel réponde à notre problématique, nous l'enrichissons par les mécanismes suivants :

– **Agrégation différentiée ($\mathcal{Ag}^{D}$)** : c'est la fonction que l'on utilise pour agréger les valeurs d'une mesure entre deux paramètres (niveaux d'agrégation) d'une hiérarchie. Elle est associée à une mesure et à un paramètre. Cette fonction est celle qui donne l'autorisation des agrégations spécifiques à chaque niveau de granularité.

– **Agrégation multiple ($\mathcal{Ag}^{M}$)** : c'est la fonction que l'on utilise pour agréger les valeurs d'une mesure entre les paramètres des hiérarchies d'une même dimension. On l'utilise pour simplifier la représentation graphique au lieu d'un usage répété d'une fonction différentiée pour plusieurs niveaux de granularité. Cette fonction est

associée à une mesure et à une dimension. Donc, il est possible d'associer à une même mesure, plusieurs fonctions d'agrégation, une pour chaque dimension.

– **Agrégation générale ($\mathcal{Ag}^G$) :** c'est la fonction que l'on utilise pour agréger les valeurs d'une mesure entre n'importe quel paramètre. Cette fonction n'est associée qu'à la mesure sans prendre en compte ni le paramètre, ni les dimensions. On l'utilise pour simplifier la représentation graphique au lieu d'un usage répété d'une fonction multiple pour plusieurs dimensions. Cette fonction représente la fonction d'agrégation dans le modèle classique.

Soit $\mathcal{F} = \{f_1, f_2, ...\}$ un ensemble fini de fonctions d'agrégation.

DEFINITION 4. — Un *schéma multidimensionnel*, noté S, est défini par (F, D, Star, $\mathcal{Ag}^G$, $\mathcal{Ag}^M$, $\mathcal{Ag}^D$).

– $F = \{F_1, ..., F_n\}$ est l'ensemble des faits, si $|F|=1$ alors le schéma multidimensionnel est appelé schéma en étoile alors que si $|F|>1$ alors le schéma est appelé schéma en constellation,

– $D = \{D_1, ..., D_m\}$ est l'ensemble des dimensions,

– Star : $F \rightarrow 2^{D \times \mathbb{N}^*}$ est une fonction qui associe chaque fait à un ensemble de dimensions en fonction desquelles il peut être analysé. Chaque dimension est étiquetée par un ordre de priorité d'exécution $\mathbb{N}^*$.

– $\mathcal{Ag}^G : M \rightarrow \mathcal{F} \times \mathbb{N}^-$ associe chaque mesure à une fonction d'agrégation et à un niveau précis d'agrégation.

– $\mathcal{Ag}^M : M \times D \rightarrow \mathcal{F} \times \mathbb{N}^-$ associe chaque mesure et dimension à une fonction d'agrégation et à un niveau précis d'agrégation.

– $\mathcal{Ag}^D : M \times H \times P \rightarrow \mathcal{F} \times \mathbb{N}^-$ associe chaque mesure et paramètre d'une hiérarchie à une fonction d'agrégation et à un niveau précis d'agrégation.

$\mathbb{N}^*$ sert à déterminer l'ordre des dimensions utilisées pour les calculs d'agrégation. En effet, il est possible d'avoir deux fonctions d'agrégation différentes pour chaque dimension considérée. Ces fonctions sont généralement non commutatives. Pour contrôler la validité des résultats obtenus, il faut prévoir un **ordre d'exécution**. Nous étiquetons chaque dimension par un numéro d'ordre qui représente la priorité dans l'exécution : la fonction d'agrégation de la dimension ayant l'ordre le plus petit est considérée comme prioritaire. Si les fonctions d'agrégation des deux dimensions sont commutatives, les deux dimensions sont étiquetées avec le même numéro d'ordre.

$\mathbb{N}^-$ sert à contraindre une agrégation en indiquant un niveau d'agrégation spécifique à partir duquel l'agrégation considérée doit se calculer. Une agrégation non contrainte sera associée à 0 tandis qu'une agrégation contrainte sera associée à une valeur négative pour forcer le calcul à partir d'un niveau inférieur choisi par rapport au niveau considéré.

LEMME 4. — Les fonctions d'agrégation assurent la *couverture* du schéma multidimensionnel, c'est-à-dire qu'il ne doit pas exister de paramètre (niveaux

d'agrégation) pour lequel nous ne connaissons pas la fonction d'agrégation à appliquer.

$$\forall i \in [1..n], \forall m_k \in M^{F_i}, \begin{cases} \exists f \in \mathrm{Ag}^{\mathrm{G}}(m_k) \\ \forall D_j \in Star(F_i), \exists f \in \mathrm{Ag}^{\mathrm{M}}(m_k) \\ \forall H_p \in H^{D_j}, \forall p_q \in P^{D_j} \setminus \left\{Id^{D_j}\right\}, \exists f \in \mathrm{Ag}^{\mathrm{D}}(m_k) \end{cases}$$

De manière moins formelle, la couverture du schéma est réalisée de plusieurs façons :

– par l'utilisation d'une fonction d'agrégation générale,

– par l'utilisation d'une fonction d'agrégation multiple pour chaque dimension,

– par l'utilisation d'une fonction d'agrégation différentiée pour chaque niveau d'agrégation,

– par combinaison de fonctions d'agrégation multiple et différentiée où la dimension, qui n'a pas de fonction multiple doit avoir une fonction différentiée pour chaque niveau d'agrégation.

2.3. *Formalismes graphiques*

L'exemple de la figure 4 se définit formellement par (F, D, Star, $\mathcal{Ag}^{\mathrm{G}}$, $\mathcal{Ag}^{\mathrm{M}}$, $\mathcal{Ag}^{\mathrm{D}}$) où

– $F = \{F_{Notation}\}$

– $D = \{D_{Matières}, D_{Etudiants}\}$

– $Star : F \rightarrow 2^{D \times \mathbb{N}^*}$ |

$Star(F_{Notation}) = \{(D_{Matières}, 1), (D_{Etudiants}, 2)\}$

– $\mathcal{Ag}^{\mathrm{G}} : M \rightarrow \mathcal{F} \times \mathbb{N}^-$ est indéfinie sur cet exemple

– $\mathcal{Ag}^{\mathrm{M}} : M \times D \rightarrow \mathcal{F} \times \mathbb{N}^-$ |

$\mathcal{Ag}^{\mathrm{M}}(m_{NoteM}, D_{Etudiants}) = (AVG\ (m_{NoteM}), 0)$[1]

– $\mathcal{Ag}^{\mathrm{D}} : M \times H \times P \rightarrow \mathcal{F} \times \mathbb{N}^-$ |

$\mathcal{Ag}^{\mathrm{D}}(m_{NoteM}, H^{Matières}_{Habilit}, p^{H_j}_{N°UE}) = (\text{Moyenne}\ (m_{NoteM}, \text{Coeff}), -1)^2$,

$\mathcal{Ag}^{\mathrm{D}}(m_{NoteM}, H^{Matières}_{Habilit}, p^{H_j}_{Semectre}) = (\text{Moyenne}\ (m_{NoteM}, \text{ECTS}), -1)^2$,

$\mathcal{Ag}^{\mathrm{D}}(m_{NoteM}, H^{Matières}_{Habilit}, p^{H_j}_{ALL}) = (AVG\ (m_{NoteM}), -1)^2$.

---

[1] Il n'y a pas de contrainte sur l'agrégation

[2] Les valeurs sont agrégées à partir des valeurs agrégées au niveau directement inférieur de celui considéré.

Le fait est défini par $F_{Notation}$ = (*'Notation'*, {m_{NoteM}}).

Les dimensions sont définies par

– $D_{Matières}$ = (*'Matières'*, {$a_{N°M}$, a_{Coeff}, $a_{Intitulé}$, $a_{N°UE}$, a_{ECTS}, $a_{Libellé}$, $a_{Semestre}$, $ALL^{DMatières}$}, {$H_{Habilit}$}) avec $H_{Habilit}$ = (*'Habilit'*, {$a_{N°M}$, $a_{N°UE}$, $a_{Semestre}$, $ALL^{DMatières}$}, {($a_{N°M}$, $a_{N°UE}$), ($a_{N°UE}$, $a_{Semestre}$), ($a_{Semestre}$, $ALL^{DMatières}$)}, {($a_{N°M}$, {a_{Coeff}, $a_{Intitulé}$}), ($a_{N°UE}$, {a_{ECTS}, $a_{Libellé}$})}), et

– $D_{Etudiants}$ = (*'Etudiants'*, {$a_{N°E}$, a_{Nom}, $a_{N°G}$, a_{NomG}, $ALL^{DEtudiants}$}, {$H_{HGroupe}$}) avec $H_{HGroupe}$ = (*'HGroupe'*, {$a_{N°E}$, $a_{N°G}$, $ALL^{DEtudiants}$}, {($a_{N°E}$, $a_{N°G}$), ($a_{N°G}$, $ALL^{DEtudiants}$)}, {($a_{N°E}$, {a_{Nom}}), ($a_{N°G}$, {a_{NomG}})}).

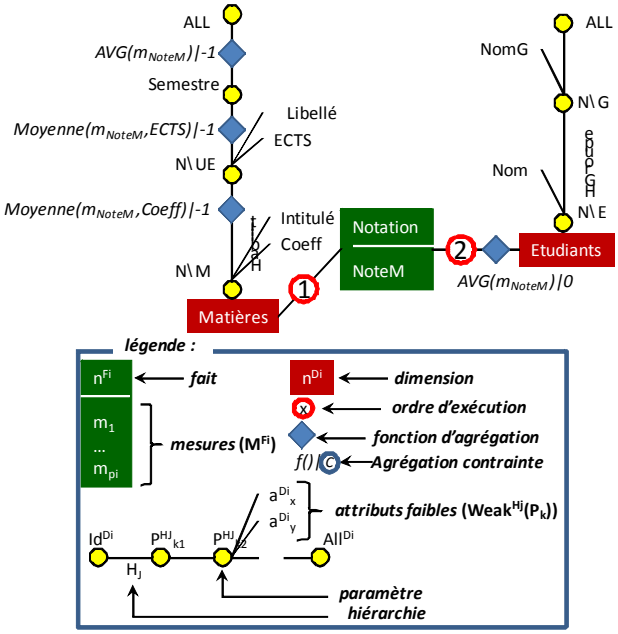


Figure 4. *Extensions des notations graphiques.*

Le formalisme graphique que nous utilisons repose sur les propositions de (Golfarelli *et al.,* 1998) et (Ravat *et al.*, 2008). Comme le montre la figure précédente, l'ordre d'exécution est symbolisé par des chiffres cerclés sur les arcs reliant le fait aux dimensions tandis que les fonctions d'agrégation sont modélisées par des losanges. Nous utilisons le même symbole (losange) pour toutes les fonctions pour ne pas surcharger le schéma. Les positions de ces losanges dépendent du type de fonction :

– la fonction générale est représentée par un losange sur le bord du fait (ce cas n'est pas utilisé dans notre exemple figure 4),

– la fonction d'agrégation multiple est localisée sur l'arc reliant le fait à la dimension,

– la fonction d'agrégation différentiée étiquette l'arc reliant deux paramètres.

3. Modèle logique R-OLAP

L'implantation courante consiste à utiliser l'approche relationnelle pour implanter les schémas multidimensionnels (Kimball, 1996). Cette approche procure de nombreux avantages : réutilisation des mécanismes de gestion des données éprouvés et capacité à gérer des volumes de données très importants.

3.1. *Etoile R-OLAP*

Dans le contexte relationnel, la BDM est traduite par des relations (Kimball, 1996). Appliqué à notre exemple, le schéma R-OLAP en étoile est le suivant :

MATIERES (**N°M**, Coeff, Intitulé, N°UE, ECTS, Libellé, Semestre)

ETUDIANTS (**N°E**, Nom, N°G, NomG)

NOTATION (**N°M#, N°E#**, NoteM)

Les fonctions d'agrégation sont stockées dans la BDM. Nous utilisons un méta-schéma que nous ne présentons pas dans cet article en raison du manque de place. Ce méta-schéma décrit le schéma multidimensionnel (faits, dimensions, hiérarchies) correspondant aux relations ROLAP stockant les données à analyser. Il décrit également les différentes fonctions d'agrégation et les éventuelles contraintes de calcul.

3.2. *Etoile optimisée*

La modélisation conceptuelle permet de structurer hiérarchiquement les graduations (paramètres) des axes d'analyses (dimensions). Ces hiérarchies sont exploitées de manière à pré-calculer les agrégations nécessaires aux décideurs lors de leurs navigations et analyses OLAP au sein de l'espace multidimensionnel.

Classiquement, les pré-agrégations sont modélisées par un treillis de pré-agrégats (Gray, *et al*., 1996) (Chaudhuri *et al.,* 1997) où chaque nœud représente un pré-agrégat et chaque arc représente le chemin des calculs d'agrégation. Lorsque la fonction d'agrégation utilisée est distributive ou algébrique, un agrégat est calculable directement à partir de l'agrégat inférieur direct, tandis que dans le cas d'une agrégation holistique, l'agrégat se calcule en cheminant jusqu'aux relations de base.

La flexibilité que nous avons introduite dans le modèle conceptuel impacte ce treillis. Comme l'illustre la figure 5, les fonctions d'agrégation multiples et différentiées impliquent l'utilisation d'agrégations différentes sur chaque arc du treillis, contrairement à l'approche classique qui considère une fonction d'agrégation unique, le plus souvent distributive.

Lorsque plusieurs chemins sont possibles, le chemin le moins coûteux est préféré. La fonction de coût, que nous ne détaillons pas, privilégie les temps de calcul les plus efficaces (Kotidis *et al.,* 1999). Nous pouvons néanmoins remarquer que l'utilisation de fonctions d'agrégation différentes sur chaque arc rend l'estimation du coût plus complexe que dans les treillis habituels.

En outre, certains chemins ou arcs sont invalides et peuvent donc être éliminés pour réduire le treillis. Cet élagage est rendu possible par l'utilisation de l'ordre d'exécution. Dans notre exemple, on ne peut pas appliquer la fonction (Moyenne (m_{NoteM}, Coeff)) après la fonction (AVG (m_{NoteM})) sur la dimension Etudiants. Ainsi, pour obtenir la moyenne de notes des UE par groupes (N°UE_N°G), on ne peut pas la calculer à partir de la moyenne de notes de matière par groupes (N°M_N°G). Donc, l'arc entre N°M_N°G et N°UE_N°G peut être supprimé.

De plus, les contraintes (niveau d'agrégation spécifique à partir duquel l'agrégation considérée doit se calculer) associées aux fonctions d'agrégation ont des répercutions sur le treillis. Les arcs apparaissant en rouge sont obtenus à partir de ces contraintes qui imposent de calculer un nœud à partir d'un nœud précis. Il est alors interdit de calculer un nœud supérieur par transitivité des nœuds inférieurs comme cela est classiquement possible. Ainsi les chemins de calcul sont bloqués dès qu'un arc rouge intervient. Par exemple, le nœud Semestre_ALL^{Etudiants} est calculable à partir du nœud inférieur direct Semestre_N°G, par transitivité, il est également calculable à partir du nœud inférieur Semestre_N°E. Par contre, l'arc rouge issu de la contrainte de la fonction Moyenne_Semestre qui opère sur l'arc (Semestre_N°E, N°UE_N°E) bloque la transitivité des calculs.

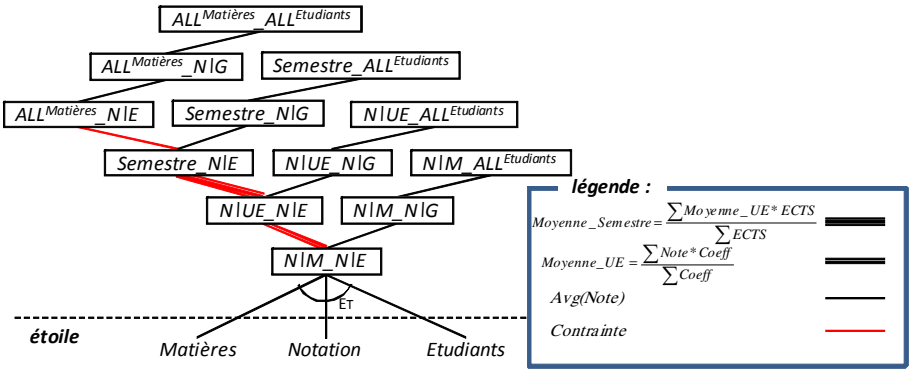


Figure 5. *Treillis d'optimisation.*

4. Expérimentations

Afin de montrer la faisabilité de notre approche, nous avons réalisé un prototype avec le SGBD Oracle 12g. Nous avons développé les fonctions d'agrégation (1) et (2) décrites dans cet article. Pour cela, nous avons implanté une fonction d'agrégation générique. Nous avons créé un type objet (classe) qui implémente les 4 méthodes de l'interface `ODCIAggregate`. Ces méthodes correspondent aux opérations internes que chaque fonction d'agrégation accomplit (*Initialize*, *Iterate*, *Merge*, *Terminate*).

Nous avons ensuite créé notre fonction d'agrégation AVG_W calculant une moyenne pondérée. Cette fonction reçoit un paramètre (`TYPE ty_donnee_ponderee AS OBJECT (valeur NUMBER, poids NUMBER)`) composé de la donnée à agréger et de sa pondération.

Le code ci-dessous présente l'utilisation de notre propre fonction d'agrégation au sein d'une série de trois requêtes SQL simulant une navigation dans l'espace multidimensionnel.

– le décideur visualise la moyenne semestrielle des étudiants.

Requête SQL sur le schéma en étoile de base (le schéma R-OLAP) :

```
SELECT Semestre, N°E, AVG_W(ty_donnee_ponderee (Note_EU, ECTS)) as NoteM
FROM (SELECT Semestre, ECTS, N°UE, D1.N°E,
                    AVG_W(ty_donnee_ponderee (NoteM, Coeff)) AS Note_EU
        FROM Notation F, Etudiants D1, Matières D2
        WHERE F.N°E = D1.N°E
            AND F.N°M = D2.N°M
        GROUP BY Semestre, ECTS, N°UE, D1.N°E)
GROUP BY Semestre, ID_Etudiant;
```

– afin de visualiser le détail des notes des étudiants éliminés, le décideur effectue un forage sur les données de sorte à obtenir la moyenne de l'étudiant par UE.

```
SELECT N°UE, N°E, NoteM
FROM (SELECT N°UE, N°E, AVG_W(ty_donnee_ponderee(NoteM, Coeff)) as NoteM
         FROM Notation F, Etudiants D1, Matières D2
         WHERE F.N°E = D1.N°E
             AND F.N°M = D2.N°M
         GROUP BY N°UE, N°E)
WHERE NoteM<10;
```

– enfin, pour connaître les matières que les étudiants devront présenter en session de rattrapage, un nouveau forage est opéré pour visualiser les moyennes par matières.

```
SELECT N°M, N°E, NoteM
FROM Notation F, Etudiants D1, Matières D2
WHERE NoteM <10
    AND F.N°E = D1.N°E
    AND F.N°M = D2.N°M;
```

Le graphique suivant montre les temps d'exécution (*ms*) de chaque requête. Le temps des deux premières requêtes augmente régulièrement en fonction du nombre de tuples. La troisième requête est quasi-stable car le nombre de tuples était faible et relativement constant dans nos expérimentations. Ces premiers résultats ont pour objet de montrer la faisabilité de notre approche. Ils sont encourageants puisque nous n'observons pas de changement notable dans le comportement des requêtes dans un entrepôt de données construit classiquement ou sur la base de nos propositions d'agrégations multiples différentiées.

Il nous reste cependant à développer nos expérimentations afin de définir les contours de notre approche. Egalement, nous devons adapter les algorithmes de calcul des treillis ; ceci est prometteur puisque notre modèle permet d'envisager des élagages comme nous l'avons évoqué à la section précédente.

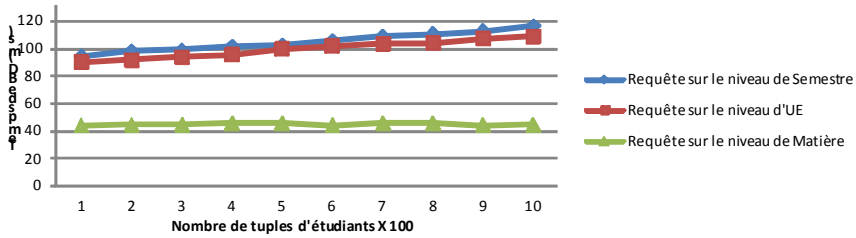


Figure 6. *Résultats d'exécution des requêtes.*

5. Conclusion

Cet article définit un modèle conceptuel de données multidimensionnelles suffisamment expressif pour autoriser le concepteur à spécifier des agrégations multiples et différentiées. Ce modèle permet ainsi la combinaison d'une mesure avec différentes fonctions d'agrégation suivant les paramètres utilisés. En outre, le modèle est suffisamment expressif pour contrôler la validité des calculs des fonctions. Au niveau relationnel le schéma R-OLAP peut être optimisé par un treillis de pré-agrégat contrôlé, c'est-à-dire dans lequel les arcs invalides peuvent être élagués.

Nous devons poursuivre nos expérimentations en utilisant des jeux de tests plus complexes afin de dresser les contours de notre approche en termes de performances, de volumes de stockage, de complexités du modèle, etc. Nous envisageons également de poursuivre ces travaux par l'étude des opérateurs de manipulation OLAP appliqués à notre modèle de données.

6. Bibliographie

Abelló A., Samos J., Saltor F., « YAM2: A multidimensional conceptual model extending UML. », *Information Systems*, vol. 31, n°6, 2006, p. 541–567.

Chaudhuri S., Dayal U., « An Overview of Data Warehousing and OLAP Technology. », *SIGMOD Record*, vol. 26, n°1, 1997, p. 65-74.

Golfarelli M., Maio D., Rizzi S., « Conceptual Design of Data Warehouses from E/R Schemes. », *Intl. Conf. HICSS*, Vol. 7, 1998, p. 334-343.

Gray J., Bosworth A., Layman A., Pirahesh H., « Data Cube: A Relational Aggregation Operator Generalizing Group-By, Cross-Tab, and Sub-Total. », *Intl. Conf. ICDE*, 1996, p. 152–159.

Gyssens M., Lakshmanan L. V. S., « A Foundation for Multi-Dimensional Databases. », *Intl. Conf. VLDB*, 1997, p. 106-115.

Harinath S., Zare R., Meenakshisundaram S., Carroll M., Guang-Yeu Lee D., *Professional Microsoft SQL Server Analysis Services 2008 with MDX*, 2009.

Kimball R., *The Data Warehouse Toolkit: Practical Techniques for Building Dimensional Data Warehouses*, John Wiley & Sons, USA, 1996.

Kotidis Y., Roussopoulos N., « DynaMat: A Dynamic View Management System for Data Warehouses. », *Intl. Conf. SIGMOD,* 1999, p. 371-382.

Jaecksch B., Lehner W., « The Planning OLAP Model - A Multidimensional Model with Planning Support. », *Intl. Conf DaWaK*, 2011, p. 14-25

Lujàn-Mora S., Trujillo J., Song I. Y., « A UML profile for multidimensional modeling in data warehouses. », *Data & knowledge Engineering 59,* 2006, p. 725-769.

Mazón J. N., Lechtenbörger J., Trujillo J., « A survey on summarizability issues in multidimensional modelling. », *Data & Knowledge Engineering 68,* 2009, p. 1452-1469.

Oliveira R., Rodrigues F., Martins P., Moura J. P., « Extending the Dimensional Templates Approach to Integrate Complex Multidimensional Design Concepts. », *Intl. Conf DaWaK*, LNCS 6862, 2011, p. 26–38.

Pedersen T., Jensen C., Dyreson C., « A foundation for capturing and querying complex multidimensional data », *Information Systems,* vol. 26*,* n°5, 2001, p. 383–423

Ravat F., Teste O., Tournier R., Zurfluh G., « Graphical Querying of Multidimensional Databases », *11th East-European Conference on Advances in Databases and Information Systems (ADBIS'07)*, 2007, p.298-313. doi: 10.1007/978-3-540-75185-4_22

Ravat F., Teste O., Tournier, R., Zurfluh G., « Algebraic and graphic languages for OLAP manipulations », *International Journal of Data Warehousing and Mining, IGI Publishing, D. Taniar,* vol. 4, N°1, 2008, p.17-46. doi: 10.4018/jdwm.2008010102

Torlone R., *Conceptual Multidimensional Models*, Chapitre 3 dans *Multidimensional Databases: Problems and Solutions,* 2003, p. 69–90, IGI Publishing Group.

Vassiliadis P., Sellis T. K., « A Survey of Logical Models for OLAP Databases. », *SIGMOD Record*, vol. 28, n°4, 1999, p. 64-69.

Vassiliadis P., Simitsis A., Skiadopoulos S., « Modeling ETL activities as graphs. », *Intl. Conf. DMDW*, 2002, p. 52-61.