Généralisation contextuelle de mesures dans les entrepôts de données

Application aux entrepôts de données médicales

Yoann Pitarch* — Cécile Favre — Anne Laurent***
Pascal Poncelet*

** LIRMM, UMR 5506*
161, Rue Ada, F-34392 Montpellier cedex 5
{pitarch ,laurent, poncelet}@lirmm.fr

*** ERIC, Université Lyon 2, Lyon*
5 av. Pierre Mendès-France, F-69676 Bron cedex
cecile.favre@univ-lyon2.fr

RÉSUMÉ. *Les hiérarchies occupent une place importante pour l'analyse dans les entrepôts de données. Mais les modèles existants ne considèrent ce concept de hiérarchie qu'au niveau des axes d'analyse que constituent les dimensions, restreignant les possibilités d'interrogation par rapport à une éventuelle hiérarchisation sur les mesures. Par exemple, dans le cadre de données médicales, une tension sera qualifiée de basse, normale ou élevée, non seulement par rapport à la tension effective mesurée, mais également vis-à-vis des caractéristiques du patient telles que l'âge. La contribution de cet article se décline selon quatre axes principaux. (1) Un modèle conceptuel graphique d'entrepôt de données permet de représenter les généralisations de mesures contextualisées. (2) Une implémentation relationnelle permet de stocker les connaissances liées aux contextes. (3) Pour gérer ces connaissances, une application web est proposée. (4) Un algorithme est présenté pour la création du cube de données qui permettra l'analyse flexible des données.*

ABSTRACT. *The hierarchies are crucial for analysis in data warehouses. Unfortunately, existing models consider these hierarchies just for dimensions. This limits the interrogation in case of a hierarchy for measures. For instance, in case of medical data, a given blood pressure will be either low, normal or high regarding not only the collected measure but also characteristics of the patient such as the age. The contribution of this paper is fourfold. (1) A conceptual graphical model of a data warehouse allows to represent contextualized measure generalizations. (2) A relational implementation stores the knowledge concerning the contexts. (3) In order to manage this knowledge, a Rich Internet Application is proposed. (4) In order to provide a flexible analysis, an algorithm is presented to build the data cube.*

MOTS-CLÉS : *entrepôt de données, généralisation de mesure, contexte, connaissance.*

KEYWORDS: *data warehouse, measure generalization, context, knowledge.*

DOI:10.3166/ISI.16.6.67-90

RSTI - ISI – 16/2011. Qualité des entrepôts de données, pages 67 à 90

1. Introduction

Les entrepôts de données (Inmon, 1996) permettent de consolider, stocker et organiser ces données à des fins d'analyse. Des faits peuvent alors être analysés à travers des indicateurs (les mesures) selon différents axes d'analyse (les dimensions). En s'appuyant sur des mécanismes d'agrégation, les outils OLAP *(On Line Analytical Process)* (Agrawal *et al.*, 1997 ; Chen *et al.*, 1996) (Han, 1997) permettent de naviguer aisément le long des hiérarchies des dimensions (Malinowski *et al.*, 2004). La puissance de ces outils place les entrepôts au centre des systèmes d'information décisionnels (Mallach, 2000). Ces considérations justifient l'émergence d'entrepôts dans des domaines aussi variés que l'analyse de ventes, la surveillance de matériel, le suivi de données médicales (Einbinder *et al.*, 2001)… Dans cet article, nous considérons cette dernière application des données médicales[1] afin d'exhiber un manque d'expressivité dans les solutions actuelles et ainsi illustrer l'intérêt de notre proposition.

Considérons le cas réel d'un entrepôt de données médicales rassemblant les paramètres vitaux (e.g., la tension artérielle...) des patients d'un service de réanimation. Afin de réaliser un suivi efficace des patients, un médecin souhaiterait par exemple connaître ceux qui ont eu une tension artérielle basse au cours de la nuit. Pouvoir formuler ce type de requête suppose l'existence d'une hiérarchie sur la tension artérielle dont le premier niveau d'agrégation serait une catégorisation de la tension artérielle (e.g., basse, normale, élevée). Toutefois, cette catégorisation est délicate car elle dépend à la fois de la tension artérielle mesurée mais aussi de certaines caractéristiques physiologiques (âge du patient, fumeur ou non...). Dès lors, une même tension peut être généralisée différemment selon le contexte d'analyse considéré. Par exemple, 13 est une tension élevée chez un nourrisson alors qu'il s'agit d'une tension normale chez un adulte. Introduites formellement dans (Pitarch *et al.*, 2009) et explicitées davantage dans (Pitarch *et al.*, 2010), ces hiérarchies dites contextuelles, hiérarchies dans lesquelles au moins une généralisation est contextualisée, ne sont implantées dans aucun modèle d'entrepôt de données actuel.

En effet, la plupart des solutions logicielles existantes pour construire un entrepôt de données souffrent de deux faiblesses majeures quant à leur gestion des hiérarchies :

– une hiérarchie ne peut être définie que sur un attribut associé à une dimension d'analyse. Si l'on considère l'exemple précédent, il n'est donc pas possible de définir une hiérarchie sur la tension artérielle car cet attribut est une mesure ;

– les hiérarchies sont considérées indépendantes (orthogonales) (Eder *et al.*, 2001). Ainsi, dans les modèles classiques, l'agrégation d'une valeur n'est fonction que d'elle-même. Plus généralement, il n'est donc pas possible de modéliser le fait qu'une caractéristique externe puisse influencer le lien d'agrégation d'une valeur.

1. Ce travail a été réalisé dans le cadre du projet ANR MIDAS (ANR-07-MDCO-008).

Ici se pose donc le problème de qualité de la modélisation, qui doit refléter la réalité du domaine métier pour permettre des analyses fiables. L'enjeu est ainsi l'intégration de la connaissance pour améliorer le processus d'analyse. Ainsi, dans cet article, nous avons pour objectif de présenter une solution complète pour obtenir un modèle, et donc a *fortiori* des analyses, de qualité, au sens du reflet de la réalité. Des travaux ont d'ores et déjà tenté de relier l'OLAP avec d'autres bases d'informations (Perdersen *et al.*, 2009). Mais il s'agit ici de connaissances d'experts eux-mêmes. Nous faisons ainsi état de l'avancement de nos travaux sur la proposition de hiérarchies contextuelles dans un entrepôt de données en étendant nos précédents travaux (Pitarch *et al.* 2010) avec une formalisation plus adaptée et un modèle graphique au niveau conceptuel permettant la discussion au moment de la conception, la mise en avant de l'implémentation prouvant expérimentalement la faisabilité de notre approche. Ce travail est mené en nous posant les quatre questions suivantes : (1) comment représenter cette connaissance au travers d'un modèle compréhensible par les utilisateurs, (2) comment stocker efficacement cette connaissance, (3) comment l'exploiter en vue d'accroître les possibilités d'analyse offertes au décideur et (4) comment la gérer (mise à jour, suppression ou ajout) facilement.

La notion de contexte est largement exploitée dans différents domaines de l'informatique. La littérature ayant trait à ce sujet est abondante et le travail de P. Brazillon sur le sujet est lui-même diversifié et considérable (Brazillon, 2011). Dans le domaine du décisionnel, on parle généralement de contexte d'analyse pour désigner le cadre multidimensionnel qui constitue le cadre d'analyse des faits. Autrement dit, par ce terme de contexte, il est sous-entendu que les faits (à travers les valeurs que prennent les mesures) sont fonction des valeurs prises par les dimensions. Or, dans la réalité des données, il s'avère que finalement les mesures elles-mêmes peuvent être hiérarchisées et que les valeurs des attributs caractérisant cette hiérarchisation de mesure peuvent elles-mêmes dépendre d'un certain contexte.

Notons que, dans cet article, nous nous focalisons plus précisément sur la généralisation de la mesure pour le premier niveau de la hiérarchie étant donné qu'elle est déterminée par le contexte, ce qui est *a priori* moins le cas pour les autres niveaux de la hiérarchie. En effet, compte tenu des valeurs, nous pouvons imaginer un autre niveau dans la hiérarchie sur la tension, regroupant par exemple les tensions basses et élevées dans une instance « anormale ». Or, cette deuxième généralisation ne fait pas intervenir de contexte.

Ainsi, pour répondre à notre objectif de généralisation contextuelle de mesure, nous proposons de modéliser la connaissance des experts du domaine d'application. Pour représenter ces connaissances, nous avons choisi d'utiliser une base de données relationnelle permettant la définition des différents liens d'agrégation en fonction des différents contextes. Par raccourci, nous l'appelons dans la suite de cet article « base de connaissances ». Précisons néanmoins que cette expression de base de connaissances est à considérer au sens des termes qui la composent ; ainsi, elle ne contient pas de moteur d'inférence. Cette base de connaissances est composée de deux tables. La première correspond à une méta-table des connaissances qui permet

de modéliser la structure des différents contextes existants dans l'entrepôt. Par exemple, nous y représentons le fait que la catégorisation de la tension artérielle d'un patient est fonction de la tension mesurée, de la catégorie d'âge du patient et du fait que le patient soit ou ne soit pas fumeur. Ensuite, la deuxième constitue la table des connaissances qui permet de modéliser les différentes instances des contextes. Par exemple, c'est dans cette table que sera exprimé le fait qu'une tension à 17 chez un adulte fumeur est élevée. Ce mode de stockage arrive après une phase de modélisation grâce à un modèle graphique. Nous décrivons ensuite une méthode pour exploiter cette base externe en permettant la création d'un cube pour l'analyse des mesures généralisées.

La suite de cet article est organisée de la façon suivante. La section 2 expose notre cas d'étude sur des données médicales qui permettra d'illustrer le problème posé et le modèle proposé. Nous discutons dans la section 3 de l'inadéquation des différentes solutions existantes par rapport à la problématique de cet article à travers un état de l'art. La section 4 s'intéresse à la représentation des connaissances introduites dans l'entrepôt pour la construction de ces hiérarchies contextuelles au travers d'une modélisation conceptuelle. Dans la section 5, nous décrivons comment mettre en œuvre notre approche en stockant, exploitant et gérant cette connaissance, et ce, en nous appuyant sur le cas d'étude précédemment présenté et en évoquant l'implémentation. La faisabilité de cette approche est ainsi prouvée de manière expérimentale. Enfin, nous concluons et indiquons les perspectives de ce travail dans la section 6.

2. Cas d'étude

Pour illustrer la problématique liée à notre approche, nous considérons le cas d'un entrepôt de données rassemblant, pour chaque patient d'un service de réanimation, sa tension ainsi que les médicaments prescrits au fil du temps. Ces valeurs sont mesurées par des capteurs et alimentent directement l'entrepôt. Dès lors, le volume des données stockées au sein de l'entrepôt est potentiellement immense.

Le schéma décrivant cet entrepôt est présenté dans la figure 1, selon le formalisme graphique présenté dans (Golfarelli et Rizzi, 2009). Ainsi, l'entrepôt permet d'observer deux faits : la tension et la posologie. Les dimensions considérées sont les suivantes. La dimension temps (partagée par les deux faits), la dimension médicament rattachée au fait posologie et la dimension patient (partagée également par les deux faits). Nous nous attardons plus spécifiquement sur la dernière dimension. Chaque patient est décrit par son nom, son âge, son sexe, la ville où il habite et par un attribut fumeur qui indique si le patient fume ou non. L'âge du patient peut être considéré selon trois niveaux de détail différents : Age, SubCatAge et CatAge.

En pratique, un médecin qui consulte un tel entrepôt peut trouver que les informations qui y sont stockées sont insuffisantes pour assurer un suivi efficace des patients du service. En effet, offrir la possibilité de formuler des requêtes telles que « Quels sont les patients dont la tension artérielle a été élevée pendant la nuit ? » ou

« Quels sont les patients qui se sont vu prescrire une quantité trop importante de médicament X ? » faciliterait le travail des médecins en leur évitant une analyse manuelle des tables de faits.

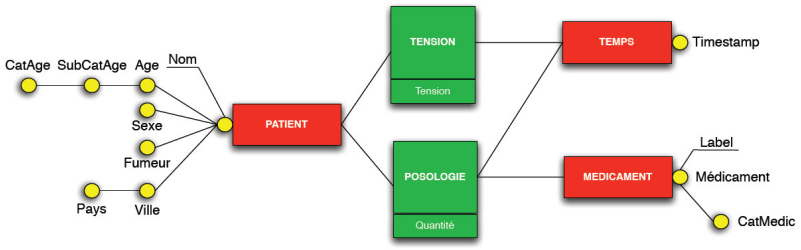


Figure 1. *Schéma simplifié de l'entrepôt de données pour l'analyse de la tension et de la posologie*

Malheureusement, le modèle présenté sur la figure 1 ne permet pas de répondre à de telles requêtes pour deux raisons. Premièrement, la notion de tension (resp. posologie) élevée peut être considérée comme une généralisation de la tension mesurée (resp. de la quantité prescrite). Dans la mesure où les modèles classiques ne permettent pas d'établir une hiérarchie sur les mesures, ces requêtes ne peuvent être formulées. De plus, même si l'on suppose que de telles requêtes sont formulables, les notions de tension élevée ou de posologie élevée sont directement liées à certaines caractéristiques des patients et/ou des médicaments prescrits. Par exemple, un bébé ne doit pas recevoir la même quantité d'un médicament qu'un adulte. Ainsi, une même posologie pourra être considérée comme faible, normale ou élevée selon l'âge du patient considéré.

Le tableau 1 présente quelques exemples de connaissances expertes sur la catégorisation d'une tension en fonction des attributs SubCatAge et Fumeur d'un patient. Par exemple, une tension à 13 est normale chez un adulte fumeur mais est élevée chez un nourrisson. Notons que dans le cas des nourrissons, l'attribut Fumeur correspond au fait de subir une tabagie passive.

Une connaissance experte est alors nécessaire pour (1) définir quels sont les attributs qui impactent sur la généralisation d'une mesure et (2) décrire cette généralisation en fonction des valeurs prises par ces attributs.

Nous remarquons donc ici le besoin en matière d'expression des connaissances pour permettre une analyse en ligne adéquate pour les experts. Dans la suite de cet article, nous nous focalisons sur la catégorisation d'une tension pour illustrer l'approche proposée.

Par la suite, nous décrivons le modèle utilisé pour représenter cette base experte ainsi que l'algorithme mis au point pour répondre efficacement à ce type de requêtes en construisant le cube adéquat. Dans la section suivante, nous présentons

préalablement les travaux qui pourraient apporter des éléments dans la construction de notre solution.

SubCatAge	Fumeur	Tension	CatTension
Nourrisson	Oui ou Non	>12	Elevée
Adulte	Oui	>14	Elevée
$3^{ème}$ âge	Oui ou Non	> 16	Elevée
Nourrisson	Oui ou Non	Entre 10 (inclus) et 12 (inclus)	Normale
Adulte	Oui	Entre 12 (inclus) et 14 (inclus)	Normale
...	...	...	...

Tableau 1. *Exemple de règles expertes décrivant la catégorie d'une tension (CatTension) en fonction de la tension mesurée, de la classe d'âge d'un patient (SubCatAge) et de l'attribut Fumeur*

3. État de l'art

Une des grandes puissances des outils d'analyse en ligne est la navigation au travers de données plus ou moins détaillées. Comme évoqué en introduction, ceci s'appuie sur des opérateurs d'agrégation qui fonctionnent à partir de la hiérarchisation des dimensions. Ainsi un certain nombre de travaux se sont d'ores et déjà intéressés aux hiérarchies de dimension, et aux problèmes d'additivité des mesures selon les dimensions qui peuvent en découler. Par ailleurs, on peut trouver la notion de mesure dérivée (ou calculée). Mais, à notre connaissance, notre proposition est la seule à apporter une alternative à la hiérarchisation de mesure.

Compte tenu de notre problématique de hiérarchisation de mesure contextuelle (non figée vis-à-vis du chemin d'agrégation), nous nous sommes intéressés aux différents travaux introduisant une certaine forme de flexibilité ou de capacité d'expression au niveau des dimensions, des hiérarchies.

Notons tout d'abord qu'un travail important de formalisation conceptuelle des différentes hiérarchies a été proposé par (Malinowski *et al.*, 2004), tentant d'établir une liste des hiérarchies possibles, plus ou moins complexes, posant plus ou moins de problèmes quant à leur exploitation lors du processus d'agrégation. Mais ce travail se situe dans le contexte de dimensions indépendantes. Et finalement, dans le problème que nous avons posé, l'enjeu est de pouvoir prendre en compte le fait que les dimensions ne sont pas forcément indépendantes, et surtout, que les mesures peuvent elles-mêmes être hiérarchisées en prenant en compte le contexte.

Afin de pouvoir rendre l'analyse plus flexible, un langage à base de règles a été développé dans (Espil *et al.*, 2001) pour la gestion des exceptions dans le processus d'agrégation. Le langage IRAH *(Intensional Redefnition of Aggregation*

Hierarchies) permet de redéfinir des chemins d'agrégation pour exprimer des exceptions dans les hiérarchies de dimensions prévues par le modèle. Ce travail permet aux utilisateurs d'exprimer eux-mêmes les exceptions dans le processus d'agrégation. En effet, afin de prendre en compte ces exceptions, les utilisateurs définissent et exécutent un programme IRAH, produisant ainsi une révision des chemins d'agrégation. L'exemple considéré est l'étude des prêts d'une compagnie de crédit en fonction de la dimension emprunteur qui est hiérarchisée. La catégorie de l'emprunteur est définie en fonction de son revenu. Mais les auteurs expliquent qu'il est possible que l'analyste veuille réaffecter un emprunteur dans une autre catégorie en voulant tenir compte d'autres paramètres que le revenu. Dans ce cas, le processus d'agrégation doit tenir compte de cette « exception ». Dans ces travaux les auteurs proposent alors un langage à base de règles qui permet de définir des analyses révisées qui tiennent compte de ce type d'exception.

Si ce langage constitue une alternative à la rigidité du schéma multidimensionnel dans le processus d'agrégation pour les utilisateurs, il ne fait qu'en modifier les chemins en fonction d'exceptions dans les hiérarchies de dimension. Or dans notre cas, il ne s'agit pas de prendre en compte des exceptions, mais bel et bien de prendre en compte des chemins d'agrégation qui dépendent d'un contexte, au niveau des mesures, en se basant sur des connaissances d'experts.

Dans (Favre *et al.*, 2007), un précédent travail s'est penché sur la proposition d'un modèle d'entrepôt à base de règles qui visait à représenter la création de nouveaux niveaux d'analyse pour répondre à un besoin de personnalisation des analyses en fonction de la connaissance individuelle d'utilisateurs. Les nouveaux niveaux étaient créés le long des hiérarchies de dimension (insertion d'un niveau ou ajout en fin de hiérarchie), forcément au-dessus du premier niveau (dimension). Le lien d'agrégation exprimé ne peut être directement lié à la table des faits, empêchant l'expression d'une hiérarchisation de mesure comme nous avons besoin de le faire.

Une alternative pouvant présenter un certain intérêt vis-à-vis du problème posé par la nécessité de prendre en compte le fait que les chemins d'agrégation le long d'une hiérarchie peuvent changer est le versionnement. Le lien avec notre travail n'est pas direct mais l'intérêt se situe dans le fait d'envisager des liens d'agrégation qui dépendent d'une autre information, en l'occurrence temporelle pour le versionnement. Il s'agit de pouvoir exprimer un changement (plutôt temporel) au niveau des instances de dimension comme proposé par (Bliujute *et al.*, 1998), au niveau des liens d'agrégation comme envisagé par (Mendelzon *et al.*, 2000) ou de la structure comme proposé par (Morzy *et al.*, 2003). Ceci répond au problème évoqué dans l'introduction sur l'indépendance des dimensions, par rapport à la dimension temporelle entre autres. Le versionnement est mis en avant pour la possibilité de stocker le fait que les dimensions puissent évoluer. Le versionnement permet donc non seulement d'historiser les données de la table des faits à travers une dimension temporelle, mais également les modifications au sein même des hiérarchies de dimension. Le problème majeur de ces approches est que ces versions sont développées par rapport à une évolution dans le temps et ne sont pas faites pour

prendre en compte une modification non pas dans le temps mais par rapport à ce que nous avons appelé « contexte ». Par ailleurs, cela ne résout pas le problème du point de vue de la hiérarchisation de mesure que nous avons besoin de modéliser.

Si l'ensemble des travaux présentés ici apportent une flexibilité dans l'analyse des données en se focalisant sur ce qui définit en l'occurrence le contexte d'analyse, à savoir les dimensions, ils ne prennent pas en compte le besoin de flexibilité au niveau de la mesure. Et il ne s'agit pas de « simples » hiérarchies dépendant seulement de la valeur de la mesure. A notre connaissance, il n'y a pas de travaux proposant ou permettant de solutionner ce problème. Ainsi, par la suite, nous proposons une modélisation, un stockage, une exploitation permettant de prendre en compte cette hiérarchisation contextuelle de mesure.

4. Représenter les connaissances des experts

Dans cette section, nous nous focalisons sur la manière de représenter la connaissance des experts. Notons que par rapport à la consistance du modèle, la généralisation des attributs mesures ne peut pas être stockée, ni dans les tables de dimension, ni dans les tables de faits. En outre, comme la connaissance des experts doit être prise en compte lors du processus d'agrégation, ceci ne peut être géré au niveau de la phase classique de chargement des données mais doit être considéré durant la phase d'analyse. Mentionnons que cette connaissance peut également s'enrichir ou évoluer dans le temps par les experts.

4.1. *Représentation des connaissances : modèle formel avec hiérarchies contextuelles de mesures*

Pour supporter le processus qui vise à la prise en compte de contextes par rapport à la détermination de la valeur de certains attributs généralisant les mesures, il est alors crucial de disposer d'un modèle d'entrepôt qui retrace cette contextualisation, par conséquent un modèle plus flexible. Cette formalisation pourrait s'adapter aussi bien au schéma en étoile qu'en flocon de neige, sous réserve de gérer le concept de niveau dans une hiérarchie.

Définition 1 (Dimensions). Soit $D = \{D_s;\ s=1..t\}$ l'ensemble des t tables de dimension de l'entrepôt. Soit $A_s = \{a_{sg} / s=1..t\ ;\ 1 \leq g \leq h_s\}$ l'ensemble des h attributs de la dimension D_s. Notons id_{Ds} l'attribut de A_s identifiant la dimension D_s.

Exemple 1. Dans notre étude, $t = 3$: $D_1 \equiv$ PATIENT, $D_2 \equiv$ TEMPS, $D_3 \equiv$ MEDICAMENTS.

A_1 = {Nom;Age;SubCatAge;CatAge;Sexe;Fumeur ;Ville ;Pays},

A_2 = {Timestamp},

A_3 = {Medicament;Label;CatMedic}.

$id_{D1} \equiv$ IdPatient, $id_{D2} \equiv$ IdTemps, $id_{D3} \equiv$ IdMedic

Définition 2 (Faits). Une table des faits est déterminée structurellement par un ensemble de dimensions et de mesures.

$F = \{F_\gamma, \gamma \geq 1\}$ constitue l'ensemble des tables de faits de l'entrepôt, certaines d'entre elles pouvant partager des dimensions communes.

$F_\gamma = (ID_\gamma, M_\gamma)$ avec $ID_\gamma = \{ id_{Ds}, s=1..\ w_\gamma\}$ les identifiants des w_γ dimensions décrivant F_γ et $M_\gamma = \{M_u, u = 1..\ v_\gamma\}$ dénote l'ensemble des v_γ mesures de F_γ.

Exemple 2. Pour l'étude de la tension, on a :

F_1 = ({IdPatient, IdTemps}, {Tension})

Pour l'étude des posologies, on a :

F_2 = ({IdPatient, IdTemps, IdMedic}, {Quantite})

Définition 3 (Attributs contextualisés et contextualisant). Un attribut est dit contextualisé lors d'un contexte si sa valeur dépend des valeurs prises par un ensemble d'autres attributs de l'entrepôt (qualifiés alors de contextualisant par rapport à ce contexte).

Notons qu'un attribut donné peut avoir tour à tour le rôle de contextualisé ou de contextualisant selon les cas.

Exemple 3. Dans l'étude de la tension, CatTension aura un rôle d'attribut contextualisé puisque sa valeur dépendra des attributs SubCatAge, Fumeur et Tension qui auront un rôle de contextualisant.

Définition 4 (Contexte : structure). Soit $\{c_i\}$ / $1 \leq i \leq n$ l'ensemble des n structures de contexte. La structure du contexte c_i est définie par un ensemble d'attributs contextualisants et un ensemble d'attributs contextualisés :

$$c_i = (\{K_{\Omega i}\}, \{L_{\Psi i}\})$$

avec $\{K_{\Omega i}\}$ un sous-ensemble des attributs contextualisants et $\{L_{\Psi i}\}$ un sous-ensemble des attributs contextualisés tels que $\Omega_i \geq 1$ et $\Psi_i \geq 1$.

Notons ainsi que $\{L_{\Psi i}\}$ peut se ramener à un singleton (un seul attribut contextualisé défini par un contexte) et, qu'en général, $\{K_{\Omega i}\}$ contient au minimum deux attributs (dans le cas contraire, la valeur de l'attribut contextualisé ne dépendrait plus que d'une seule valeur, celle de la mesure en l'occurrence, et on se ramènerait à une conception classique de hiérarchie).

Exemple 4. c_1 = ({SubCatAge, Fumeur, Tension}, {CatTension})

Définition 5 (Contexte : instances). Chaque structure de contexte est ensuite instanciée. On a alors C = $\{c^j_i, i = 1..n, j = 1..m\}$ constitue l'ensemble de toutes les instances de contextes.

Notons c^j_i l'instance j de la structure du contexte i. Elle est définie par l'instanciation de chacun des attributs.

L'instanciation des attributs correspond à l'affectation d'une expression. En termes d'implémentation en base de données qui sera évoquée par la suite, cette expression devra respecter une syntaxe SQL valide.

L'instanciation I^j des attributs contextualisés de $\{L_{\Psi i}\}$ correspond à une expression caractérisant l'affectation d'une et une seule valeur : $I^j(L_{\Psi i})$ = " $=val^j_{\Psi i}$ "

Exemple 5. Instanciation d'attribut contextualisé : $I^1(L_1)$ = "='Elevée'"

Par contre, l'instanciation des autres attributs du contexte ne correspond pas forcément à l'expression d'une affectation de valeur. Elle correspond à une expression pouvant représenter plusieurs valeurs. Ainsi ce terme peut être noté comme suit :

Soit $T^{\omega j}_i$ le terme correspondant à l'instance j de l'attribut ω du contexte i. $T^{\omega j}_i$ est de la forme "op {ens | val}" où op est un opérateur (=, <, >, ≤, ≥, ≠, ∈ ...) ; ens est un ensemble de valeurs et val une valeur.

Exemple 6. Instanciation d'attributs contextualisants :

T^{11}_1 = "='Nourrisson'"

T^{21}_1 = "IN('Oui', 'Non')"

T^{31}_1 1 = "> 12"

Exemple 7. Contexte :

c^1_1 = ({"='Nourrisson'", "IN('Oui', 'Non')", "> 12"}, {"='Elevée'"})

Ceci représente le fait qu'une tension supérieure à 12, chez un nourrisson, victime ou non de tabagie passive, constitue une tension élevée.

Notons que le contexte est construit par conjonction de prédicats dans cette étude. L'expressivité pourrait être enrichie au moyen d'autres opérateurs pour combiner les prédicats (le OU par exemple).

Dans la suite, l'appellation contexte représente en fait une instance de contexte (à la fois sa structure et ses valeurs) lorsque nous ne précisons pas spécifiquement qu'il s'agit de la structure.

4.2. *Modèle conceptuel graphique*

Notre objectif est ici de fournir une représentation graphique d'un modèle conceptuel qui permettrait de représenter les hiérarchies de mesures contextuelles, modèle utile lors de la phase de conception pour permettre des échanges fructueux entre concepteur(s) et décideurs/utilisateurs.

En effet, la formalisation, que nous venons de proposer, permet certes de représenter ces hiérarchies contextuelles mais demeure assez lourde en terme de notation et n'est donc pas forcément facilement interprétable.

Pour introduire la contextualisation dans les généralisations de mesures, nous introduisons le concept de satellite.

G = A → B un chemin de généralisation entre deux attributs. La représentation graphique associée à G sera différente en fonction de sa nature :

– si G est un chemin classique entre mesures, le formalisme graphique associé est celui présenté dans la figure 2 (schéma de gauche). Les mesures sont représentées par des petits cercles gris reliés entre eux par un trait plein ;

– si G est un chemin contextuel entre mesures, le formalisme graphique associé est celui présenté dans la figure 2 (schéma de droite). La mesure A est représentée par un cercle gris et la mesure contextualisée B par un satellite entourant un cercle gris. La liste des éléments contextualisant est adossée au satellite et le lien entre A et B est représenté par un trait plein.

NOTE. — Rappelons ici la correspondance avec les notations employées dans la formalisation précédente (les « membres » de C n'étant autres que les attributs contextualisant des contextes).

Figure 2. *Représentation graphique d'un chemin de généralisation de mesure*

Si nous appliquons cette représentation graphique à notre cas d'étude, nous obtenons le modèle de la figure 3.

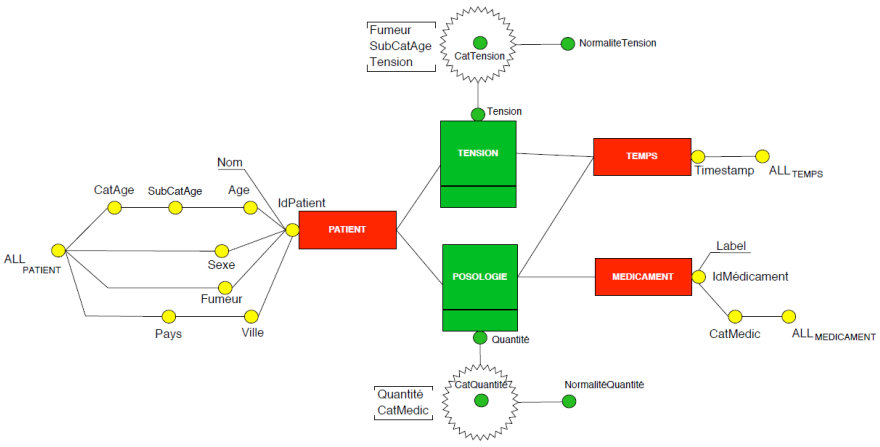


Figure 3. *Représentation graphique de l'entrepôt de données médicales*

Nous pouvons relever la présence de deux hiérarchies contextuelles de mesures : (Tension ; CatTension ; NormaliteTension) et (Quantite ; Cat-Quantite ; NormaliteQuantite). Un tel formalisme graphique rend immédiate la compréhension de l'entrepôt modélisé.

4.3. *Discussion*

Nous avons proposé ici la modélisation au niveau conceptuel de la prise en compte des connaissances utilisateurs. Cette phase de modélisation nécessite une prise en compte des utilisateurs, en partant de l'hypothèse d'un besoin d'échange avec les experts sur le modèle. Cette phase de modélisation est cruciale. Nous distinguons trois types d'approches de modélisation : les approches orientées données (ascendantes), les approches orientées besoins (descendantes), et les approches hybrides combinant les deux types précédents. Les approches hybrides permettent de combiner une réalité des données avec la satisfaction des besoins utilisateurs, satisfaction permettant d'assurer l'utilisation du système. Pour cette prise en compte des besoins utilisateurs, un échange avec les futurs utilisateurs, ou du moins des personnes connaissant bien le domaine métier, apparaît nécessaire. Pour ce faire, le recours à un modèle graphique facilement lisible est alors crucial. Rappelons qu'historiquement, les premiers modèles d'entrepôt de données (étoile, constellation proposés dans (Kimball, 1996)) ont émergé et connu un vif succès au sein des entreprises.

Il est assez naturel de faire l'hypothèse que la simplicité de lecture/d'interprétation de ces modèles graphiques a sans doute participé à leur succès. D'un point de vue support d'échange entre concepteur et utilisateur final

(décideurs/experts du domaine), il paraît assez naturel de faire un parallèle entre le modèle entité/association dans le contexte des bases de données et les modèles en étoile/flocon/constellation dans le contexte des entrepôts de données. Le modèle entité-association est qualifié consensuellement de modèle conceptuel ; cependant, dans le domaine des entrepôts, aucun consensus sur la modélisation n'a encore réellement émergé et c'est un sujet qui fait toujours débat aujourd'hui.

Dans ce travail, au-delà d'étendre le principe de généralisation contextuelle aux attributs de dimension, il s'agit donc surtout de donner toute son importance à la représentation graphique en apportant comme contribution, une visualisation qui permettra de discuter des données et connaissances à mettre en œuvre dans l'entrepôt de données grâce à ce modèle graphique volontairement simple, à l'image des premiers modèles d'entrepôt, tout en permettant une puissance d'expressivité accrue sur un aspect plutôt complexe dans les hiérarchies de mesure, à savoir les hiérarchies contextuelles. Bien évidemment, ce modèle graphique se situe au niveau de la représentation structurelle. Cela permet de structurer les connaissances à exprimer et de les confronter avec tous les acteurs de la modélisation. Pour l'explicitation des connaissances, il s'agit de s'intéresser ensuite aux instances. La réalisation de cet aspect est abordée dans la section suivante.

5. Mise en œuvre d'une solution

La figure 4 présente l'architecture considérée pour notre approche. Elle montre l'utilisateur qui saisit ses connaissances sur les contextes (celles-ci-étant stockées dans des tables, elles sont partageables). Cette partie stockage est détaillée dans la sous-section 5.1. La construction du cube de données en prenant en compte les satellites permettra l'analyse OLAP en tenant compte des contextes exprimés. La partie création du cube est détaillée dans la section 5.2.

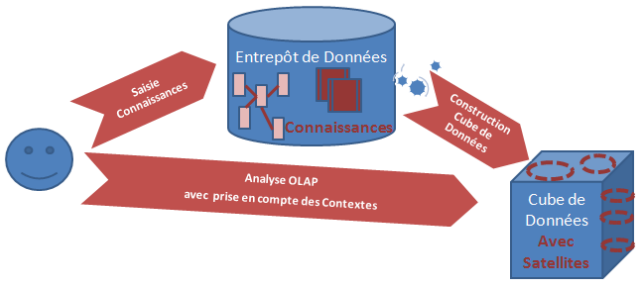


Figure 4. *Architecture proposée pour l'exploitation de généralisations de mesures contextualisées*

5.1. *Stockage relationnel des connaissances*

Cette section aborde la problématique du stockage des connaissances expertes dans l'hypothèse d'une mise en œuvre dans un système relationnel (ROLAP) pour le stockage de l'entrepôt de données, étant donné que c'est le mode de stockage dans le cadre de ces données médicales. Pour cela, nous proposons donc la création d'une base de données dont nous détaillons ici le contenu.

Dans la mesure où plusieurs contextes peuvent cohabiter au sein d'un même entrepôt (e.g., la catégorisation d'une tension artérielle et celle d'une posologie), il est nécessaire de stocker quels sont les attributs qui interviennent dans chaque contexte (*i.e.*, quels sont les attributs contextualisants) car ceux-ci peuvent différer pour chaque contexte. Nous proposons alors la création d'une table MTC (méta-table des connaissances) afin de stocker la structure associée à chaque contexte présent dans l'entrepôt.

Définition 6 (MTC). Soit MTC = (Contexte, Attribut, Table, Type) une table relationnelle où :

– *Contexte* désigne l'identifiant du contexte (c_i) ;

– *Attribut* désigne un attribut intervenant dans le contexte c_i (*i.e.* contextualisé ou contextualisant) ;

– *Table* correspond à la table relationnelle de l'entrepôt où Attribut est instancié ;

– *Type* définit le rôle joué par Attribut dans le contexte c_i. Cet attribut vaut "Contexte" si Attribut est un attribut contextualisant de c_i et vaut "Résultat" si Attribut est l'attribut contextualisé de c_i.

MTC			
Contexte	**Attribut**	**Table**	**Type**
Tension	SubCatAge	PATIENT	Contexte
Tension	Fumeur	PATIENT	Contexte
Tension	Tension	TENSION	Contexte
Tension	CatTension	TC	Resultat
Quantite	…	…	…

Tableau 2. *Extraits de la table MTC décrivant le contexte associé à la généralisation d'une tension artérielle*

Exemple 8. Considérons le cas d'étude présenté dans la section 2 et analysons comment est stockée la structure du contexte associé à la généralisation d'une tension artérielle. Nous rappelons que la catégorisation d'une tension est fonction de plusieurs paramètres : la tension mesurée, la catégorie d'âge du patient et du fait qu'il soit fumeur ou non. Par convention, le nom donné à chaque contexte est celui

de la mesure à généraliser. Ici, l'attribut Contexte vaut donc "Tension". Ensuite, les attributs dont le type est Contexte (*i.e*, les attributs contextualisants) sont SubCatAge, Fumeur et Tension. Ceux-ci sont définis dans les tables PATIENT et TENSION. Enfin, l'attribut contextualisé est CatTension. Le tableau 2 présente un extrait de la table MTC associé au contexte Tension.

Une fois la structure de chaque contexte définie dans MTC, nous nous intéressons à la matérialisation des instances de chaque contexte. Pour cela, nous définissons au sein de la base de données externe une nouvelle table relationnelle TC (table des connaissances). Notons que sur l'exemple précédent, la table associée à l'attribut contextualisé (CatTension) est cette nouvelle table TC. Dans la mesure où les valeurs prises par cet attribut sont dépendantes des valeurs prises par les attributs contextualisants, ce choix de modélisation semble le plus adapté.

Définition 7 (TC). Soit TC = (Contexte; Instance_contexte; Attribut; Valeur) une table relationnelle de la base de données externe telle que :

– *Contexte* désigne l'identifiant du contexte (c_i) ;

– *Instance_contexte* identifie l'instance du contexte concernée (correspond à l'indice j pour c^j_i) ;

– *Attribut* désigne un attribut intervenant dans le contexte ci (*i.e*. contextualisé ou contextualisant) ;

– *Valeur* correspond à la valeur affectée à Attribut dans c^j_i .

TC			
Contexte	**Instance_contexte**	**Attribut**	**Valeur**
Tension	1	SubCatAge	= 'Nourisson'
Tension	1	Fumeur	IN ('Oui', 'Non')
Tension	1	Tension	>12
Tension	1	CatTension	= 'Elevée'
Tension	2	SubCatAge	= 'Adulte'
Tension	2	Fumeur	= 'Oui'
Tension	2	Tension	>14
Tension	2	CatTension	= 'Elevée'
Tension	3	SubCatAge	= '3ème âge'
Tension	3	Fumeur	IN ('Oui', 'Non')
Tension	3	Tension	>16
Tension	3	CatTension	= 'Elevée'
Tension	4	…	…

Tableau 3. *Extraits de la table TC présentant quelques instances de contexte associées au contexte Tension*

Pour diminuer la taille de la table TC, le nombre d'instances de contexte stockées peut être réduit en autorisant l'attribut Valeur à contenir une expression SQL syntaxiquement correcte. Par exemple, "IN ('Oui', 'Non')" évite la création de deux instances de contexte (une pour chaque valeur de l'opérateur IN).

Considérons le contexte Tension présenté lors de l'exemple précédent. Comme illustré dans le tableau 1, de nombreuses règles expertes existent pour déterminer la généralisation d'une tension artérielle. Alors que la structure de ces règles (*i.e.*, quels sont les attributs qui impactent sur la généralisation d'une mesure) est définie au niveau de la méta-table des connaissances MTC, les instances de ces contextes (*i.e.*, les règles expertes) sont définies au niveau de la table des connaissances TC. Le tableau 3 présente un extrait du contenu stocké dans TC. Chaque ligne du tableau 1 représente une instance du contexte Tension.

Exemple 9. Illustrons le stockage des instances de contexte en considérant la connaissance « chez un nourrisson, une tension supérieure à 12 est élevée ». Cette instance est représentée dans TC par les 4 n-uplets dont la modalité de l'attribut Contexte est Tension et celle de l'attribut Instance_contexte est 1. Parmi ces 4 n-uplets, les 3 premiers permettent de décrire les conditions à réunir pour que la généralisation (stockée dans le quatrième n-uplet) soit valide.

5.2. *Exploiter les connaissances des experts*

Maintenant que nous avons décrit comment représenter et stocker les hiérarchies contextuelles, nous nous focalisons sur leur exploitation et leur interrogation afin de fournir à l'utilisateur une plus grande flexibilité dans l'analyse des données de l'entrepôt. La solution adoptée doit satisfaire deux conditions : ne pas diminuer la puissance d'analyse des outils OLAP et autoriser l'interrogation des hiérarchies contextuelles. Pour cela, nous proposons la prise en compte de ces hiérarchies lors de la création du cube de données pour l'analyse.

Ainsi, en assimilant la construction du cube à la création d'une vue, notre problème d'interrogation est résolu par la création de la vue adéquate. Nous ne discutons pas ici le choix entre vue et vue matérialisée puisque la performance n'est pas notre préoccupation dans cette étude.

Pour illustrer cette solution, nous partons sur l'hypothèse de la création d'une vue FG dont les attributs sont (1) ceux de la table de faits et (2) un attribut contenant la généralisation de la mesure considérée. Les analyses OLAP classiques sont alors possibles en considérant au choix la table des faits ou la nouvelle vue FG. L'interrogation des hiérarchies contextuelles peut quant à elle être réalisée en interrogeant FG. Notons en outre que, par définition, la création de cette vue n'entraîne pas d'augmentation du volume de données stockées (si cette vue n'est pas matérialisée).

La création de la requête générant cette vue est présentée dans l'algorithme 1. Dans un premier temps, nous recherchons quels sont les attributs ayant un impact sur la mesure que nous désirons généraliser. Pour cela, il est nécessaire de rechercher dans MTC quels sont les attributs contextualisants appartenant au contexte recherché. Afin de faciliter les jointures futures, les tables dans lesquelles sont

stockés ces attributs sont également récupérées. L'algorithme se poursuit par la recherche de l'attribut contextualisé. L'étape suivante (lignes 4 à 12) permet d'écrire la requête générant la vue souhaitée. Pour cela, une sous-requête calculant l'instance du contexte associé à chaque n-uplet de la table de faits est mise en place et se base sur le principe suivant. Pour chaque attribut contextualisant, il faut d'abord récupérer la valeur associée puis trouver dans TC quelles sont les instances de contexte qui concordent avec cette valeur. Enfin, l'intersection de ces ensembles de contextes permet de récupérer le contexte qui valide toutes les conditions associées à chaque n-uplet de la table de faits. En vue de la mise en œuvre de ce modèle, nous envisageons l'utilisation du modèle relationnel-objet. Finalement, la requête permettant de créer la vue est générée.

Illustrons cet algorithme en utilisant le cas d'étude présenté au cours de la section 2. Ici, nous cherchons à généraliser les tensions des patients. Dès lors, les attributs contextualisants sont SubCatAge, Fumeur et Tension. L'attribut contextualisé est CatTension. Etudions maintenant la requête générée grâce à la boucle qui permet de déterminer l'instance du contexte associé à chaque n-uplet de la table des faits. Pour cela, nous considérons le premier n-uplet de la table des faits. Si l'on considère l'extrait de la table TC présenté dans le tableau 3, l'instance du contexte Tension associée à un nourrisson dont la tension est 13 est l'instance 1. En effet, l'instance de contexte 1 est la seule dont les modalités de l'attribut Valeur associées aux attributs SubCatAge et Tension coïncident avec celles du n-uplet de la table de faits étudiée ici. Nous remarquons par contre que les contextes 1 et 3 sont tous les deux valides pour l'attribut Fumeur. L'intersection de ces ensembles d'instances permet de déduire que l'instance 1 du contexte Tension doit être associée au premier n-uplet de la table de faits. Par conséquent, lors de la création de la vue FG, la modalité de l'attribut contenant la généralisation sera "='Elevée'" (*i.e.*, celle stockée dans l'attribut Valeur dont le contexte est Tension, l'instance du contexte est 1 et la modalité de l'attribut Attribut est CatTension).

A travers cet exemple, nous avons illustré comment est stockée et exploitée la connaissance experte du domaine. Dès lors, la puissance d'analyse d'un entrepôt utilisant des hiérarchies de mesure contextuelles est accrue. De plus, cet apport de flexibilité n'entraîne pas une complexification du processus d'analyse dans la mesure où la création de la vue est transparente pour l'utilisateur. En outre, notons que l'ajout ou la modification de connaissances au sein des méta-tables et table de connaissances peut être rendue très aisée grâce à une interface de saisie.

L'approche que nous proposons permet un gain d'expressivité important dans les entrepôts de données médicales en permettant la modélisation, le stockage et l'exploitation de connaissances expertes dans le processus de généralisation de mesures. Néanmoins, un autre aspect essentiel doit être considéré ici : gérer, maintenir et mettre à jour cette connaissance. En effet, en fonction des patients accueillis dans le service ou des progrès de la médecine, certains contextes et instances de contexte peuvent être ajoutés, modifiés ou bien supprimés. En outre, la

simple consultation de cette connaissance *via* une interface simple représente un besoin pour des utilisateurs non spécialistes en bases de données.

C'est pour répondre à ces objectifs que nous avons développé une application web basée sur le SGBD PostgreSql. Nous détaillons maintenant quelques-unes des principales fonctionnalités en nous appuyant sur le cas d'étude présenté dans la section 2.

Algorithme 1 : Construction de la vue étendant une table de faits

Data : $F = (k_1, \ldots, k_n, m_1, \ldots, m_i, \ldots, m_l)$ une table de faits où m_i est la mesure à généraliser, $\mathcal{T} = \{T_1, \ldots, T_n\}$ (avec $T_i = (k_i, a_i^1, ...)$) les dimensions d'analyse associées avec $k_1, \ldots, k_n$les clés primaires de ces tables, MTC la Méta Table des connaissances et TC la table des connaissances

Result : ReqVue,une chaîne de caractères contenant la requête permettant la création de la vue généralisée

```
/* 1) Recherche des couples <Table,Attribut> associés au
   contexte étudié                                          */
```
1 PairAttrib = SELECT Table, Attribut FROM MTC WHERE Contexte='m_i' and Type='Contexte' ;

```
/* 2) Recherche de l'attribut contextuaisé                  */
```
2 *attrGen* = SELECT Table, Attribut FROM MTC WHERE Contexte='m_i' AND Type='Resultat' ;

```
/* 3) Construction du nom de la vue qui sera créée          */
```
3 *nameView* = m_i+'_View' ;
4 ReqVue = "CREATE OR REPLACE VIEW *nameView* AS
5 SELECT' F.k_1,. . .,F.k_n,F.$m_1, \ldots, F.m_i, \ldots, F.m_l$, (SELECT Valeur FROM TC WHERE Attribut = '*Resultat*' AND Contexte = 'm_i' ;
6 **pour chaque** $\langle T_i, A_i \rangle \in$ *PairAttrib* **faire**
7 ReqVue = ReqVue + " AND instanceCtxt IN (select instance_match FROM ";
8 **si** $T_i = F$ **alors**
9 ReqVue = ReqVue + " instance_match('m_i','m_i','m_i','F) ";
10 **sinon**
11 ReqVue = ReqVue + " instance_match('m_i','F','k_i','k_i','A_i','T_i) ";
12 ReqVue = ReqVue + ") as *AttrGen* FROM F ";
13 **retourner** *ReqVue* ;

Puisque la création de la base de données experte ne peut être réalisée sans l'aide des médecins, il est essentiel que l'interface de saisie des contextes et de leurs instances soit la plus simple et intuitive possible. C'est pourquoi l'application web développée propose les fonctionnalités suivantes :

– la création, suppression et modification de contextes,

– la création, suppression et modification d'instances de contexte.

Nous nous attardons plus spécifiquement sur une des fonctionnalités pour montrer la facilité d'utilisation de notre application. La figure 5 présente et développe la partie de l'application gérant les instances de contexte. La partie supérieure affiche les différentes instances pour un contexte donné. Un simple clic sur [Update] transforme cette ligne d'affichage en un formulaire permettant la modification de l'instance. Sur le même principe, cliquer sur [X] permet de supprimer une instance. La partie inférieure de cette page permet la saisie de nouvelles instances de contexte. Lors de la soumission de ces formulaires, une vérification est effectuée pour s'assurer que deux instances de contexte identiques n'aient pas une généralisation différente.

5.3. *Discussion*

Notre approche permet ainsi de répondre à la problématique de représentation et de stockage de contextes. Elle offre ainsi un moyen de prendre en compte une certaine forme de hiérarchisation de mesure, définie par un contexte.

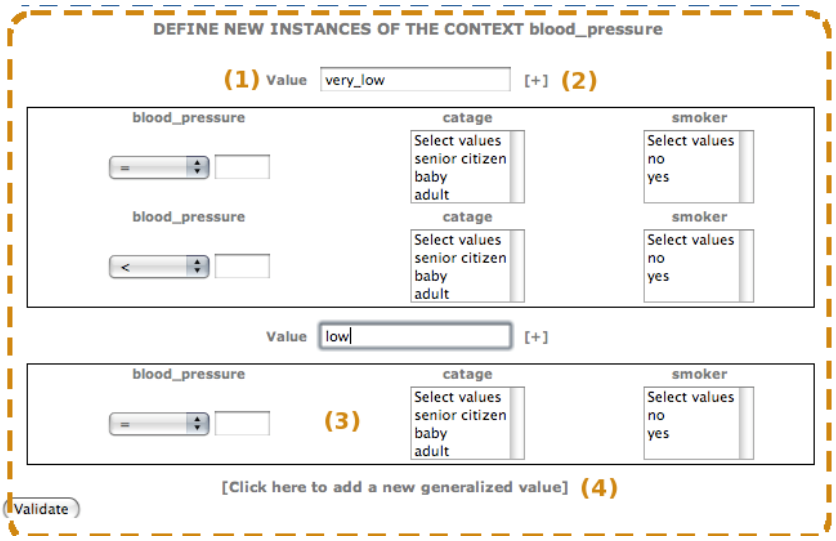
Cette approche présente différents avantages. Ce mode de représentation permet de représenter différents contextes composés de structures différentes. Elle constitue ainsi une approche générique et permet l'ajout facile de contextes, autrement dit de nouvelles connaissances. Les contextes sont stockés dans une même table évitant que chaque contexte soit représenté par différentes tables, facilitant à terme le processus de réécriture de requêtes.

Un des intérêts de cette proposition est également de pouvoir regrouper des instances ensemble pour définir la fonction d'agrégat plutôt que de stocker le chemin d'agrégation de chaque instance, ce qui constitue un avantage en termes de complexité. Par exemple, chaque valeur de tension ne va pas faire l'objet de l'expression de son propre chemin d'agrégation mais un chemin pourra être défini pour un ensemble de valeurs (ainsi > 12 pour l'attribut Tension, reprend un ensemble de tensions).

Le choix d'une représentation relationnelle permet alors d'exploiter la puissance d'interrogation relationnelle. Notons par ailleurs que compte tenu de la connaissance à représenter, le recours à une ontologie n'était pas envisageable, les types de liens utilisés au niveau des ontologies ne permettant pas de représenter nos contextes (structures et instanciation). Par ailleurs, partant sur une mise en œuvre en relationnel, dans un contexte d'implémentation tel que celui-ci, le recours à la construction d'une vue peut alors constituer une approche pertinente. L'algorithme de construction de vue constitue une première proposition pour l'exploitation de ce type de connaissances. Il méritera d'être affiné au niveau de la construction des cubes, prenant donc en entrée d'autres paramètres. Mais dans un premier temps, cela permet une exploitation réelle de ces connaissances introduites.

high	> (11)	IN (child,baby,teen)	IN (yes,no)	[Update]	[X]
normal (1)	Betwwen 10 14	Select values senior citizen baby adult	Select values no yes	[Update] (2)	[X] (3)
normal	between (8,16)	IN (adult)	IN (no)	[Update]	[X]
normal	between (11,14)	IN (senior_citizen)	IN (no,yes)	[Update]	[Cancel delete]
normal	between (9,11)	IN (teen,baby)	IN (no,yes)	[Update]	[X]

Mise à jour ou suppression d'instances :
(1) Chaque ligne correspond à une instance
(2) Cliquer sur [Update] permet de modifier une instance
(3) Cliquer sur [X] permet de supprimer une instance



Nouvelles instances :
(1) Saisir une nouvelle valeur pour la généralisation
(2) Cliquer sur [+] permet de créer une nouvelle instance possédant la même généralisation
(3) Renseigner le formulaire
(4) Cliquer ici permet de créer un ensemble d'instances possédant la même généralisation

Figure 5. *Capture d'écran commentée de la page permettant la gestion des contextes*

IDPATIENT	IDT	BLOOD_PRESSURE	PULSE	CAT_BLOOD_PRESSURE
1	1	13	70	high
2	1	12	90	high
3	1	10	100	normal
4	1	14	65	normal
5	1	8	80	low
6	1	14	70	normal
1	2	10	80	normal
3	2	16	70	high
6	2	10	100	low

Nous remarquons que :

-13 est une tension élevée pour le le patient 1 (un nourrisson)

- 14 est une tension normale pour le patient 6 (une personne du 3ème âge qui ne fume pas)

Figure 6. *Capture d'écran de la page de visualisation de la vue étendant la table de faits TENSION avec la généralisation de l'attribut Tension artérielle (Blood Pressure)*

En début de section, nous avons proposé un algorithme de création de vue pour exploiter efficacement les connaissances stockées dans la base externe. L'avantage de ce mécanisme est de conserver la puissance d'analyse des outils OLAP tout en permettant d'interroger des mesures généralisées. L'application web proposée permet de visualiser ces vues. Une fois le contexte sélectionné, l'algorithme pour calculer la vue est appelé et le résultat est affiché. La figure 9 présente la vue générée à partir de la table des faits et de la connaissance présentée au cours de la section 2. Nous remarquons que la généralisation de la mesure tension s'effectue correctement car 13 est bien considéré comme une tension élevée pour un bébé alors que 14 est considéré comme une tension normale chez une personne âgée.

En exploitant efficacement la connaissance stockée et en permettant une gestion facile de la base de données externe, nous prouvons que l'approche proposée présente un réel intérêt dans le contexte des entrepôts de données médicales.

6. Conclusion et perspectives

Dans cet article, nous nous sommes attachés à montrer l'état actuel de nos travaux sur la prise en compte de hiérarchies dites contextuelles au sein d'un entrepôt de données et ce, du point de vue des mesures. En s'intéressant au domaine

médical, nous avons montré l'importance du problème d'une limitation du point de vue de la flexibilité des analyses pour la prise en compte de contextes dans la généralisation des mesures. Ainsi, afin de représenter les différents contextes de généralisation, nous proposons la construction de deux tables externes pour stocker la connaissance experte du domaine. La méta-table des connaissances stocke la structure des différents contextes de l'entrepôt (e.g., la normalité d'une tension artérielle dépend de la tension mesurée, de la catégorie d'âge du patient et du fait qu'il fume ou non). La table des connaissances permet quant à elle d'exprimer les différentes instances d'un contexte donné (e.g., une tension supérieure à 14 chez un adulte fumeur est une tension élevée). En nous appuyant sur ces tables, nous proposons la création d'une vue étendant la table de faits et garantissant à l'utilisateur une analyse flexible, efficace et adéquate de l'entrepôt de données. Cette création peut s'apparenter à la construction d'un cube. Dans cet article, nous avons en particulier pu proposer une formalisation étendant la précédente (en ne se limitant pas au schéma en étoile), ainsi qu'une implémentation validant la faisabilité de notre approche.

De nombreuses perspectives s'ouvrent à la suite de ce travail. Parmi elles, il s'agit en premier lieu de pouvoir réaliser une étude auprès des utilisateurs du système pour valider la satisfaction d'utilisabilité du système compte tenu des besoins initiaux auxquels nous souhaitions apporter une solution.

De façon très pratique, l'implémentation en relationnel a été réalisée compte tenu du mode de stockage initial des données médicales. Etant donné que nous avons proposé un modèle conceptuel, il pourrait être intéressant d'envisager d'autres alternatives, que ce soit en termes de stockage de données, de connaissances, et de structures de données pour l'analyse.

Vis-à-vis des utilisateurs, il s'agit par la suite de mettre en œuvre des processus de validation des connaissances recueillies pour assurer l'intégrité et la cohérence des contextes définis. Notre approche permet de satisfaire un problème d'expressivité des modèles actuels mais dans un second temps, il serait intéressant de se pencher sur la validité des connaissances exprimées. L'expressivité du modèle pourrait également aller au-delà, par exemple au niveau de l'utilisation d'autres opérateurs tels que le OU, etc.

L'intégration de la logique floue est une perspective connexe qui nous paraît présenter un intérêt particulier de part l'augmentation d'expressivité qu'elle permettrait. Ceci mérite une étude d'intérêt auprès des experts, mais également de faisabilité.

Dans un cadre d'aide à l'expression des connaissances, il nous paraît alors également intéressant d'explorer la voie d'un système de découverte semi-automatique de contextes grâce à des approches de fouille de données. Il s'agit par exemple de pouvoir utiliser des approches d'apprentissage supervisées telles que les arbres de décision, qui permettraient, à partir d'un ensemble de données étiquetées par les experts, d'apprendre les contextes (structure des contextes à travers les

attributs utilisés pour déterminer les nœuds de l'arbre et instances des contextes grâce aux règles de décision proposées par le système).

Par ailleurs, d'un point de vue modélisation, dans la mesure où notre modèle permet la mise en place de hiérarchies sur des mesures, il serait intéressant d'étudier l'intégration des hiérarchies contextuelles à un modèle dit « en galaxie » proposé par (Ravat *et al.*, 2007) ne comportant pas de faits initialement définis.

En outre, l'exploitation de cette proposition dans d'autres domaines d'activités nous permettra de montrer la généricité de notre approche, et peut-être d'aller au-delà dans notre proposition, soulevant d'autres problèmes d'expressivité.

Enfin, bien que notre souci fût d'apporter une solution à un manque d'expressivité de modèles, il s'agit également de pouvoir porter une attention particulière à l'aspect performances en menant une étude sur les requêtes portant sur la contextualisation que nous avons proposée.

7. Bibliographie

Agrawal R., Gupta A., Sarawagi S., "Modeling multidimensional databases", *Proceedings of 13th International Conference on Data Engineering (ICDE '97)*, 1997, p. 232-243.

Bebel B., Eder J., Koncilia C., Morzy T., Wrembel R., "Creation and management of versions in multiversion data warehouse", *Proceedings of the ACM Symposium on Applied Computing (SAC'04)*, Nicosia, Cyprus, 2004, p. 717-723.

Bliujute R., Saltenis S., Slivinskas G., Jensen C., "Systematic Change Management in Dimensional Data Warehousing", *Proceedings of the IIIrd International Baltic Workshop on Databases and Information Systems*, Riga, Latvia, 1998, p. 27-41.

Brézillon P., "From expert systems to context-based intelligent assistant systems: a testimony", *Knowledge Eng. Review*, 2011, vol. 26, n° 1, p. 19-24.

Chen M.-S., Han J., Yu P.S., "Data mining: An overview from a database perspective", *IEEE Trans. on Knowl. and Data Eng.*, 1996, vol. 8, n° 6, p. 866-883.

Eder J., Koncilia C., "Changes of Dimension Data in Temporal Data Warehouses", *Proceedings of the IIIrd International Conference on Data Warehousing and Knowledge Discovery (DaWaK'01)*, Munich, Germany, 2001, volume 2114 of LNCS, Springer, p. 284-293.

Einbinder J.S., Scully K.W., Pates R.D., Schubart J.R., Reynolds R.E., "Case study: a data warehouse for an academic medical center", *Journal of Healthcare Information Management (JHIM)*, 2001, vol. 15, n° 2, p. 165-175.

Espil M.M., Vaisman A.A., "Efficient Intensional Redefinition of Aggregation Hierarchies in Multidimensional Databases", *Proceedings of the IVth ACM International Workshop on Data Warehousing and OLAP (DOLAP '01)*, Atlanta, Georgia, USA, 2001, p. 1-8.

Favre C., Bentayeb F., Boussaid O., « Evolution et personnalisation des analyses dans les entrepôts de données : une approche orientée utilisateur », *Actes du XXVe congrès*

INFormatique des ORganisations et Systèmes d'Information et de Décision (INFORSID '07), Perros-Guirec, France, 2007, p. 308-323.

Golfarelli M., Rizzi S., *Data Warehouse Design: Modern Principles and Methodologies*, McGraw-Hill, 2009

Han J., "OLAP mining: An integration of OLAP with data mining", *Proceedings of the 7th IFIP 2.6 Working Conference on Database Semantics (DS-7)*, 1997, p. 1-9.

Inmon W.H., *Building the Data Warehouse*, 2nd Edition, Wiley, 1996.

Kimball R., *The Data Warehouse Toolkit*, John Wiley & Sons, 1996.

Mallach E.G., *Decision Support and Data Warehouse Systems*, McGraw-Hill Higher Education, 2000.

Mendelzon A.O., Vaisman A.A., "Temporal Queries in OLAP", *Proceedings of the XXVIth International Conference on Very Large Data Bases (VLDB '00)*, Cairo, Egypt, 2000, p. 242-253.

Morzy T., Wrembel R., "Modeling a Multiversion Data Warehouse: A Formal Approach", *Proceedings of the Vth International Conference on Enterprise Information Systems (ICEIS '03)*, Angers, France, 2003, p. 120-127.

Malinowski E., Zimanyi E., "OLAP Hierarchies: A Conceptual Perspective", *Proceedings of the XVIth International Conference on Advanced Information Systems Engineering (CAiSE '04)*, Riga, Latvia, 2004, volume 3084 of LNCS, Springer, p. 477-491.

Malinowski E., Zimanyi E., "A conceptual model for temporal data warehouses and its transformation to the ER and the object-relational models", *Data Knowl. Eng.*, 2008, vol. 64, n° 1, p. 101-133.

Pedersen T.B, Gu J., Shoshani A., Jensen C.S., "Object-extended OLAP querying", *Data Knowl. Eng.*, 2009, vol. 68, n° 5, p. 453-480.

Pitarch Y., Favre C., Laurent A., Poncelet P., « Analyse flexible dans les entrepôts de données : quand les contextes s'en mêlent », *Actes de 6e conférence sur les Entrepôts de Données et l'Analyse en ligne (EDA '10)*, Djerba, Tunisia, 2010.

Pitarch Y., Laurent A., Poncelet P., "A conceptual model for handling personalized hierarchies in multidimensional databases", *Proceedings of the 1st International Conference on Management of Emergent Digital EcoSystems (MEDES '09)*, Lyon, France, 2009, ACM, p. 107-111.

Ravat F., Teste O., Tournier R., Zurfluh G., "A Conceptual Model for Multidimensional Analysis of Documents", *Proceedings of the International Conference on Conceptual Modeling (ER '07)*, Auckland, New Zealand, 2007, volume 4801 of LNCS, Springer, p. 550-565.