

Personnalisation du contenu des bases de données multidimensionnelles

Housseem Jerbi, Franck Ravat,
Olivier Teste, Gilles Zurfluh

IRIT, Institut de Recherche en Informatique de Toulouse (IRIT – UMR 5505)
Université Paul Sabatier - 118 route de Narbonne
F-31062 Toulouse, France
{jerbi,ravat,teste,zurfluh}@irit.fr

Résumé. Les systèmes OLAP se basent généralement sur des Bases de Données Multidimensionnelles (BDM) qui représentent des extractions de l'entrepôt, dédiées à des groupes de décideurs. Les utilisateurs d'un même groupe ont souvent différentes perceptions du contenu d'une BDM. Nous proposons un cadre de personnalisation pour les systèmes de gestion des BDMs basé sur des profils utilisateurs. Ces profils sont constitués de préférences contextuelles qui permettent d'adapter le contenu de la BDM à la perception de chaque utilisateur, formant un contenu personnalisé. Durant l'exécution d'une requête, le système reformule la requête en tenant compte des préférences de l'utilisateur afin de simuler son exécution sur un contenu individuel.

1 Introduction

L'information représente un capital immatériel dont la bonne gestion est un facteur primordial pour la réussite et le progrès de toute organisation. La mise en place d'un système d'information décisionnel performant facilitant le stockage et l'exploration des informations est parmi les priorités impératives de ces organisations. Dans le monde de la recherche ainsi qu'au niveau des outils commerciaux, les bases de données multidimensionnelles (BDM) sont reconnues comme un espace de stockage adapté pour le stockage et la manipulation des données décisionnelles de l'entreprise. Les données sont tout d'abord chargées, à partir de plusieurs sources hétérogènes, dans un entrepôt de données. Puis, des extractions dédiées à des métiers particuliers de l'entreprise (marketing, risque, contrôle de gestion, ...) sont effectuées à partir de l'entrepôt et stockées dans les BDMs (appelés aussi magasins) selon des modèles facilitant l'analyse décisionnelle. Au sein des BDMs, les données sont structurées en sujets (faits) et axes d'analyse (dimensions).

Le processus de conception d'une BDM vise à définir le schéma multidimensionnel en réponse aux besoins d'information et d'analyse d'un groupe de décideurs. Un outil de type ETL (Extract-Transform-Load) permet d'extraire, transformer et charger les instances de ce schéma, représentant le contenu de la BDM (les instances du fait et valeurs des attributs). Généralement, un seul processus ETL est mis en place par magasin vu le coût d'alimentation et de rafraîchissement des données. Cependant, les utilisateurs partageant la même BDM ont souvent différentes perceptions de son contenu (Rizzi, 2007), (Golfarelli et Rizzi, 2009). Par

exemple, pour les valeurs de l'attribut *Pays*, un utilisateur a besoin d'analyser les ventes en France, alors qu'un autre préfère avoir une analyse plus globale pour les pays européens, tandis que le système stocke les valeurs pour tous les pays. Ceci oblige souvent les décideurs à définir des requêtes complexes afin de restituer le contenu qui leur est pertinent, ce qui demande un effort manuel et cognitif.

Par ailleurs, le même utilisateur peut avoir des préférences sur le contenu qui varient d'un contexte d'analyse à un autre (Garrigós et al., 2009), (Jerbi et al., 2009). Par exemple, le même utilisateur est intéressé par toutes les années dans le contexte d'analyse des achats, alors qu'il considère seulement l'année en cours pour l'analyse des ventes. Par conséquent, la personnalisation du contenu des BDMs représente une étape importante pour offrir un accès individualisé aux données en OLAP (Golfarelli et Rizzi, 2009), (Garrigós et al., 2009).

D'autre part, afin de supporter le processus de prise de décision des différents utilisateurs, les BDMs stockent souvent un volume de données très important. Bien qu'un système de gestion de BDM offre les moyens d'exécuter des requêtes complexes sur de grands volumes de données, le décideur peut ne pas savoir quelle partie de la BDM interroger pour obtenir des informations utiles. Ce problème gêne l'exploration des données et réduit les avantages d'usage d'un système de gestion de BDM. Ainsi, ces systèmes doivent évoluer pour permettre de nouvelles initiatives telles que fournir un accès à l'information plus en rapport avec les intérêts de l'utilisateur ou encore, aider les utilisateurs à trouver rapidement des données pertinentes. Selon (Rizzi, 2007) et (Golfarelli et Rizzi, 2009), la prise en compte des préférences de chaque utilisateur lors de l'interrogation d'une BDM permettrait de résoudre le problème du volume important des données OLAP.

Cet article propose un cadre de personnalisation pour les systèmes de gestion de BDM qui est basé sur des profils utilisateurs. Plus précisément, nous nous focalisons sur la personnalisation du contenu des BDMs (Olap Content Personalization, *OCP*). L'*OCP* est le processus d'adaptation des données décisionnelles stockées aux besoins spécifiques de chaque décideur. Nous montrons comment restituer un contenu de données personnalisé pour les décideurs.

Cet article est organisé comme suit : la section 2 présente les travaux existants, la section 3 présente une description générale de notre approche, la section 4 notre cadre de base pour la personnalisation du contenu et la section 5 propose une extension de ce cadre pour assurer une personnalisation avancée. La section 6 présente nos résultats expérimentaux. Enfin, les conclusions et les travaux futurs sont présentés dans la section 7.

2 Etat de l'art

La personnalisation de l'accès aux données a été largement étudiée dans le domaine des bases de données (Kiebling, 2002), (Koutrika et Ioannidis, 2005), (Abbar et al., 2008). Les approches existantes sont toutefois inadaptées au cadre OLAP où les requêtes utilisateur suivent un caractère navigationnel (une succession de contextes d'analyse) (Jerbi et al., 2009). Ainsi, l'utilisateur définit ses besoins en fonction de son contexte d'analyse courant. Par conséquent, les modèles des préférences ainsi que leur gestion (sélection et intégration dans la requête) doivent être dépendants du contexte d'analyse (Jerbi et al., 2008, 2009).

Dans son agenda de recherche, Rizzi (2007) affirme que la définition d'une approche ad-hoc de personnalisation pour l'OLAP est une problématique principale qui doit être abordée. La nécessité de cette personnalisation a été soulignée dans (Garrigós et al., 2009), où les

auteurs la qualifient d'une solution efficace à la diversité des caractéristiques, des comportements et des contextes des différents usagers d'une BDM.

Différentes approches ont été proposées pour la personnalisation du schéma d'une BDM. Nous relevons principalement deux catégories de travaux. La première catégorie traite l'évolution du schéma par la mise à jour des dimensions et de leurs hiérarchies afin de les adapter aux besoins des utilisateurs (Hurtado et al., 1999), (Favre et al., 2007). La deuxième catégorie de travaux concerne la définition de préférences utilisateur (Jerbi et al., 2008) ou de règles ECA (Ravat et Teste, 2009) sur le schéma d'une BDM afin d'individualiser la navigation au sein des structures multidimensionnelles et de la rendre plus facile.

Contrairement au niveau du schéma, peu de travaux ont étudié la personnalisation du contenu d'une BDM. Ces travaux ont traité principalement le tri du résultat d'une requête afin de présenter à l'utilisateur les données les plus pertinentes en premier ordre (Li et al., 2006), (Xin et al., 2006). Ce tri est généralement basé sur des techniques de clustering afin de répondre aux requêtes de type Top-k.

A notre connaissance, il existe seulement deux travaux qui traitent la personnalisation dans un cadre OLAP en se basant sur des profils utilisateur. Bellatreche et al. (2006) proposent de personnaliser la visualisation des résultats de requêtes en fonction d'une contrainte d'affichage. Cette proposition vise plutôt à remédier aux problèmes des limites de l'espace d'affichage dans certains dispositifs. Le mécanisme de personnalisation est appliqué après la sélection du contenu, alors que dans notre approche la requête subit une étape de personnalisation avant d'interroger la BDM. Plus récemment, Golfarelli et Rizzi (2009) ont proposé, en s'inspirant des travaux de Kießling (2002) en bases de données relationnelles, une algèbre pour l'expression des préférences OLAP. Les préférences sont intégrées directement au sein de la requête. Par contre, dans notre approche, des préférences plus durables sont stockées dans un profil utilisateur et prises en compte lors de l'évaluation des requêtes, sans être obligatoirement redéfinies au niveau de chaque requête.

Dans ces deux travaux, les préférences sont représentées par des ordres stricts partiels. Ceci ne permet pas de capturer les différents degrés d'intérêt, comme « Je suis très intéressé par les produits de santé », « Je suis peu intéressé par les produits importés ». Nous détectons de telles variations d'importance par l'association des préférences avec des degrés d'intérêt. De plus, les préférences utilisateur dans ces deux travaux sont indépendantes du contexte d'analyse, ce qui engendre leur prise en compte dans tout contexte. Dans notre modèle, chaque préférence est rattachée à un contexte d'analyse qui précise son cadre d'application.

Finalement, différents travaux ont été proposés récemment pour la recommandation de requêtes OLAP (Jerbi et al., 2009), (Giacometti et al., 2009). Ces travaux visent à suggérer, en plus du résultat de la requête Q encours, des requêtes supplémentaires qui ont été déjà appliquées sur la BDM (Giacometti et al., 2009), ou qui sont calculées d'une façon incrémentale à partir des préférences du décideur (Jerbi et al., 2009). Contrairement à la recommandation de requêtes, la démarche de personnalisation que nous proposons permet d'ajouter à la requête initiale Q des contraintes issues du profil utilisateur. Plus précisément, notre démarche renvoie une requête augmentée Q' , telle que $Q \subseteq Q'$, alors qu'une requête recommandée n'est pas obligatoirement incluse dans la requête initiale Q . Notons que le mécanisme de personnalisation que nous proposons pourrait être appliqué à posteriori d'un processus de recommandation afin de rendre la requête recommandée plus appropriée à l'utilisateur en l'affinant davantage.

3 Description générale de notre approche

3.1 Exemple illustratif

Dans cet article, nous utilisons la base de données des publications du laboratoire comme exemple illustratif pour nos différentes propositions. Une partie du schéma de la BDM associée est représentée dans FIG 1. Pour représenter le schéma d'une constellation, nous adoptons des notations graphiques proches de (Golfarelli et al., 1998). Cette BDM permet d'analyser les publications ainsi que le suivi des missions de recherche des membres du laboratoire. Le schéma comporte deux faits (*Publications* et *Missions*) et trois axes d'analyse (*Dates*, *Manifestations* et *Auteurs*).

Au niveau de la dimension *manifestations*, l'attribut *catégorie* représente la catégorie de la publication (IEEE, ACM, ...), alors que le *type* détermine s'il s'agit d'une revue, une conférence ou un atelier. Le *niveau* de la manifestation admet deux valeurs possibles : nationale ou internationale. La hiérarchie *HPOS* de la dimension *auteurs* permet l'analyse des publications ainsi que le suivi des missions selon les *statuts* (permanant, non permanent) ou les *postes* des auteurs (professeur, maître de conférences, doctorant, ...).

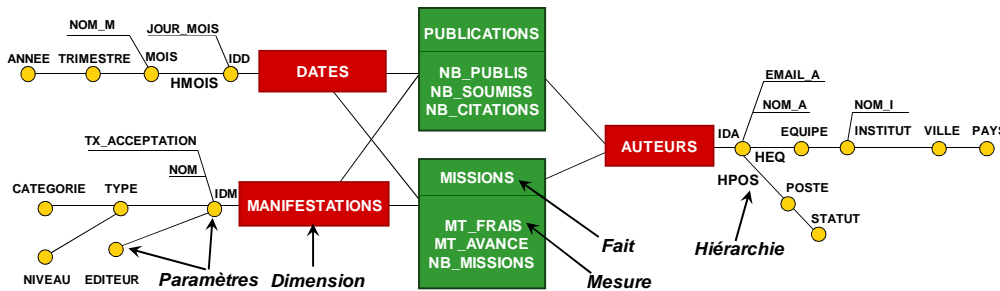


FIG 1 – Schéma d'une BDM de gestion des publications et des missions de recherche.

Exemple 1. Afin de bien spécifier le problème, nous allons prendre l'exemple de deux responsables d'équipes de recherche. Le premier s'intéresse essentiellement aux publications dans des conférences IEEE et ACM, alors que le deuxième considère toutes les publications.

Pour suivre les performances de leurs équipes, les deux responsables expriment la requête suivante : « nombre de publications de l'année en cours par trimestre, par catégorie de publication ». Le système leur restitue un même résultat comprenant toutes les catégories de publications. Le premier responsable est donc obligé de chercher au sein de l'espace multidimensionnel résultat les données correspondants seulement aux catégories IEEE et ACM. Notons que, pour le premier responsable, l'année de publication (*année* = 2010) représente un besoin instantané qui est exprimé explicitement dans la requête. Cependant, la catégorie de la publication est une information à long terme qui représente un centre d'intérêt du responsable qui le différencie des autres chercheurs du laboratoire. Il s'agit d'une préférence qui doit être prise en compte automatiquement par le SGBD.

La non-prise en compte du besoin particulier de ce décideur peut avoir des effets plus problématiques, surtout dans le cas où la catégorie de conférence ne représente pas un attribut d'agrégation de la requête. Dans ce cas, les données renvoyées par le système représente-

ront un calcul qui est effectué sur une autre base de valeurs. Par exemple, lorsque ce décideur désire analyser le nombre de publications des auteurs par année et statut de l'auteur, la valeur calculée pour chaque année et statut sera différente dans le cas où seules les conférences IEEE et ACM sont considérées.

Nous proposons de personnaliser le contenu d'une BDM, et d'individualiser par la suite son interrogation. Ceci sera permis grâce à la connaissance préalable de l'utilisateur par le système. Dans notre exemple, le système stocke les préférences du premier décideur sous la forme de contraintes. Lors de l'interrogation de la BDM, seules les valeurs répondant à ces contraintes seront prises en compte pour le calcul du résultat. Par conséquent, ces deux décideurs, qui expriment la même requête, auront des résultats différents adaptés à chacun.

3.2 Architecture du système

Nous définissons la personnalisation du contenu *OCP* par le processus d'adaptation des instances de la BDM aux besoins spécifiques et aux caractéristiques de l'utilisateur. L'ensemble de ces informations sont définies en tant que préférences d'analyse et sont stockées sous formes de profils utilisateurs. FIG 2 décrit un cadre *OCP*.

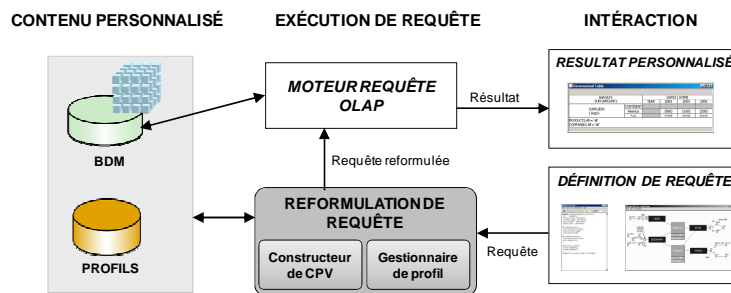


FIG 2 – Processus de personnalisation du contenu des BDMs (*OCP*).

Le système maintient pour chaque utilisateur un profil qui forme avec le contenu de la BDM un contenu personnalisé. Les profils sont définis au moment de la conception, comme ils peuvent être créés après la mise en place de la BDM. Ils sont mis à jour pour suivre l'évolution des besoins spécifiques des décideurs dans le temps. Dans cet article, nous ne traitons pas le processus d'acquisition et de mise à jour des profils. Nous nous focalisons plutôt sur leur exploitation pour la restitution du résultat. Plus précisément, l'approche que nous décrivons ici se situe dans un point t ponctuel du temps, où les profils utilisateurs sont dans un état $E(t)$. Au moment de l'exécution d'une requête, le processus *OCP* reformule la requête afin de prendre en compte les préférences de l'utilisateur en cours (les éléments du profil). La requête reformulée est alors exécutée sur la BDM générant un résultat personnalisé, simulant l'exécution de la requête initiale sur une BDM individuelle.

Nous distinguons entre les profils utilisateurs, stockés en tant qu'une couche descripteur enrichissant une BDM et les requêtes utilisateurs : un profil est un modèle utilisateur qui décrit les centres d'intérêt d'un utilisateur qui le différencie des autres, alors qu'une requête est un besoin utilisateur qui est exprimé par un ordre explicite, et dont l'évaluation doit tenir compte de son profil.

L'étape la plus importante du processus *OCP* est la reformulation de la requête qui est assurée par le gestionnaire de profils ainsi que le constructeur du contenu personnalisé virtuel :

- La gestion de profils est faite en deux étapes : une étape hors ligne qui permet d'acquérir et de mettre à jour les profils, et une étape en ligne qui permet de rechercher les éléments du profil qui sont relatifs à la requête en cours de l'utilisateur.
- La construction du contenu personnalisé consiste à construire, à partir des éléments du profil, un contenu personnalisé sur lequel sera appliquée la requête. Ceci implique la gestion d'éventuels conflits entre la requête et les profils.

4 Cadre *OCP*

Nous décrivons dans cette section notre cadre *OCP* de personnalisation du contenu OLAP.

4.1 Contenu OLAP Personnalisé

4.1.1 Modèle des données OLAP

Les données OLAP sont généralement stockées dans des tables de fait qui sont composées des attributs de mesures du fait et d'attributs de dimensions qui sont connectées à ce fait.

Définition. Une BDM \mathcal{BM} est définie par $\mathcal{BM} = (F^{BM}, D^{BM})$ où $F^{BM} = \{F_1, \dots, F_n\}$ est un ensemble de faits et $D^{BM} = \{D_1, \dots, D_m\}$ est un ensemble de dimensions.

- Pour $\forall i \in [1, m]$, D_i est une table de dimension de schéma $Sch(D_i) = \{a_i^1, a_i^2, \dots\}$, où a_i^1, a_i^2, \dots est l'ensemble d'attributs de D_i qui sont organisés du paramètre de plus faible granularité vers le paramètre de plus forte granularité. Nous supposons que a_i^1 est le niveau le plus bas au sein de D_i . a_i^1 représente la clé primaire de la table de D_i .
- $\forall j \in [1, n]$, F_j est une table de fait de schéma $Sch(F_j) = \{a_1^1, \dots, a_m^1, m_j^1, \dots, m_j^v\}$ où $\{m_j^1, \dots, m_j^v\}$ est un ensemble de *mesures* (ou indicateurs) de F_j qui peuvent être agrégées selon une fonction f^{AGREG} (AVG, SUM, COUNT, ...).

4.1.2 Modélisation du profil utilisateur

Le profil d'un utilisateur est constitué d'un ensemble de prédicats traduisant son intérêt particulier en une partie du contenu de la BDM. Nous distinguons :

- des préférences sur les valeurs des attributs des dimensions désignant les tranches du cube de données qui sont pertinentes (les publications des trois dernières années)
- des préférences sur les valeurs des mesures du fait précisant les cellules du cube qui sont appropriées pour l'utilisateur (nombre de publications supérieur à 3).

Définition. Etant donné une BDM \mathcal{BM} , pour un attribut A donné, une préférence est définie par $P^A = (pred^A; \theta)$, où

- $pred^A$ est une disjonction de prédicats sous la forme $A \text{ op } a_i$ qui spécifie une condition sur les valeurs a_i de A

- θ est un nombre réel entre 0 et 1 indiquant le degré d'intérêt de l'utilisateur en les données qui sont générées par $pred^A$.

A désigne un attribut de dimension ou une mesure, éventuellement associée avec une fonction d'agrégation $f^{AGREG}(m_i)$, où $f^{AGREG} = \text{SUM, AVG, COUNT, ...}$

Dans un cadre OLAP, les préférences d'un même utilisateur varient selon son contexte d'analyse. Par exemple, pour l'analyse des publications en recherche d'information (qui n'est pas la thématique de recherche de l'équipe), le décideur est intéressé seulement par les revues, alors qu'il aime analyser tous les types de publications dans la thématique des entrepôts de données. Afin de modéliser cette variation, nous proposons d'associer une préférence donnée à un contexte d'analyse (Jerbi et al., 2008, 2009). Ces préférences sont qualifiées de contextuelles. Notons qu'une préférence peut être indépendante de tout contexte. Il s'agit de préférences absolues qui sont associées au contexte par défaut *ALL*. La contextualisation des préférences utilisateur permet de préciser le cadre de l'application d'une préférence.

Contexte de préférence. Les préférences utilisateurs peuvent dépendre d'analyses selon une ou plusieurs dimensions (Jerbi et al., 2008, 2009). Par exemple, le décideur admet une préférence dans le contexte d'analyse des données selon les auteurs et les manifestations. De plus, ces préférences peuvent être exprimées à différents niveaux de détail. Par exemple, une préférence peut être associée au contexte d'analyse des données mensuelles, alors que d'autres sont associées à l'analyse des données par année.

Nous définissons le contexte d'analyse d'une préférence par : $cp = ([F_i (.m^{Fi} / \text{pre}/.../ \text{pre})^*]; [D_1 / (/ \text{pre}/.../ \text{pre})^*]; \dots; [D_p / (/ \text{pre}/.../ \text{pre})^*])$, où : F_i est le fait analysé à travers la mesure m^{Fi} , D_1, \dots, D_p sont les dimensions de l'analyse, pre est le niveau de détail de l'analyse selon la dimension D_q , et pre est une sélection sur l'attribut ou la mesure x .

Le signe "*" désigne l'absence d'un élément ou la présence de plusieurs. En effet, certains éléments de cp peuvent être vides. Ceci est traduit par l'affectation de la valeur *ALL* aux propriétés correspondantes. Par exemple, une préférence qui est associée à $cp_1 = (\text{PUBLICATIONS.NB_PUBLIS}, \text{ALL}, \text{ALL})$ est pertinente pour toute analyse des nombres de publications, indépendamment des axes courants. De même, une préférence associée à $cp_2 = (\text{ALL}, \text{AUTEURS/POSTE}=\text{'Doctorant'}, \text{DATES/ALL})$ est appropriée pour toute analyse des données des doctorants selon l'axe temporel, indépendamment du sujet analysé et du niveau de détail selon la dimension *DATES*.

Définition. Un profil utilisateur \mathcal{P} est défini par l'ensemble de couples $M_i = (P_i, cp_i)$ où M_i représente l'association (*mapping*) de la préférence P_i au contexte d'analyse cp_i .

Exemple 2. La préférence absolue présentée dans l'exemple 1 se traduit par : $M_1 = (P_1, \text{ALL})$ où $P_1 = (\text{CATEGORIE}=\text{IEEE OR CATEGORIE}=\text{ACM}; 1)$.

4.2 Notre approche

Dans cette section, nous présentons le mécanisme d'exécution d'une requête OLAP Q sur un contenu personnalisé. L'objectif de ce mécanisme est de générer une requête augmentée dont l'exécution simulera l'application de la requête Q sur une autre BDM qui est propre à l'utilisateur en cours. Ceci est assuré par l'enrichissement de Q par les prédicats de profils stockés dans la BDM.

La construction du contenu personnalisé consiste à chercher, pour une requête Q , les prédicats du profil pour enrichir le contenu de la BDM. Il s'agit de rechercher l'ensemble de préférences candidates $P^{Cand} = (P^C, P^A)$, où P^C (respectivement P^A) est un ensemble de préférences contextuelles (resp. absolues).

Préférences valides. Parmi les préférences stockées dans le profil \mathcal{P} de l'utilisateur, certaines sont relatives à la requête Q en cours. Ces préférences sont qualifiées de préférences *actives*. En effet, sans perte de généralité, nous considérons dans cet article le cas des requêtes OLAP qui interrogent les données d'un seul fait: une requête s'applique sur un schéma en étoile E de la BDM. Par conséquent, seules les préférences qui sont rattachées à E (fait F^Q et mesures M^Q_1, \dots, M^Q_w de Q , ainsi que toutes les dimensions connectées à F^Q) seront prises en compte pour l'exécution de Q .

Afin d'accélérer la recherche des préférences actives, un prétraitement hors ligne de préparation de données est effectué. Il s'agit de stocker dans une méta-table les correspondances (*mappings*) entre les couples (fait, mesure) de la BDM et les préférences associées. Pour une requête Q , les préférences actives sont celles mises en correspondance avec (F^Q, M^Q) .

Préférences candidates. Vu la contextualisation des préférences OLAP, afin de personnaliser le contenu par rapport à une requête Q , ne seront considérées parmi les préférences valides que celles associées au contexte d'analyse courant de Q . Il s'agit de l'ensemble des préférences candidates P^{Cand} .

Définition. Le Contexte d'Analyse Courant CAC est le contexte d'analyse induit par la requête utilisateur Q (Jerbi et al., 2008, 2009). Il s'agit d'une représentation arborescente intégrant le fait F^Q , les mesures M^Q_1, \dots, M^Q_w , les dimensions d^Q_1, \dots, d^Q_x , les niveaux de granularité p^Q_1, \dots, p^Q_y , ainsi que les éventuelles restrictions $pred^Q_1, \dots, pred^Q_z$ de la requête Q .

P^{Cand} est donc déterminé grâce à une résolution de contexte qui détermine les préférences qui sont rattachées au CAC . Cette étape est détaillée dans la section 4.2.2.

Toutefois, au cours du mécanisme d'OCP, des conflits peuvent survenir.

Gestion de conflits. Nous identifions deux types de conflits potentiels :

- Conflit entre les prédicats de la requête et ceux des préférences
- Conflit entre les prédicats de deux préférences

Définition. Deux prédicats $pred_1$ et $pred_2$ (prédicats normalisés) sont dits en conflit si et seulement si $pred_1 \wedge pred_2 = \text{Faux}$.

En cas de conflit entre un prédicat de préférence et un prédicat de requête, le prédicat de préférence est rejeté.

Dans le cas de conflits mutuels entre préférences, certains conflits sont gérés hors ligne d'une façon anticipée. Il s'agit du cas de la définition ou de l'apprentissage d'une nouvelle préférence P_1 qui porte sur la même propriété (mesure ou attribut de dimension) qu'une préférence P_2 : P_1 et P_2 sont absolues ou sont les deux contextuelles et rattachées au même contexte. Le système laisse le choix au décideur d'en choisir une ou de garder par défaut la plus récente.

Par contre, certains conflits ne sont détectés qu'au moment de la requête, suite au processus de résolution de contextes: une préférence contextuelle par rapport à une préférence ab-

solue ou deux préférences contextuelles qui sont relatives à des contextes différents. Dans ce cas, l'incompatibilité de deux prédicats de préférences induit un rejet des deux.

Le résultat de l'étape de gestion de conflits est un ensemble de préférences homogènes (compatibles mutuellement et avec la requête). Si cet ensemble est vide, le contenu de la BDM représente un contenu qui est parfaitement adapté aux préférences de l'utilisateur en cours pour le contexte d'analyse courant *CAC*. Il s'agit du cas où les préférences ont été définies explicitement dans la requête, ou du cas où l'utilisateur n'admet pas de préférences valides pour le *CAC*. Par conséquent, le système exécute normalement la requête. Autrement, les préférences homogènes seront intégrées dans *Q*.

Nous décrivons dans la sous-section suivante l'étape d'intégration de préférences.

4.2.1 Algorithme de personnalisation

Dans un premier temps, nous présentons le scénario de personnalisation le plus simple où toutes les préférences de l'utilisateur sont supposées absolues. Nous considérons le cas de systèmes OLAP stockant les données dans un contexte R-OLAP. Les opérations OLAP sont appliquées à la BDM sous forme de requêtes SELECT-GROUP-BY-HAVING, éventuellement étendues par les opérateurs rollup et cube (Gray et al., 1996) pour le calcul des agrégats à différents niveaux d'agrégation (voir FIG 3).

```
SELECT D1. a1, ..., Dn.an, [fAGREG ( ] Fi.mj [ )],...
FROM Fi, D1, ...,Dn
WHERE Fi. Clé1 = D1.ID AND ... AND Fi. Clén = Dn.ID
[AND Dj.aj Op Cstej] [AND Fi.mj Op Cstei]
GROUP BY Dk.ak, ..., Dm.am | ROLLUP(Dk.ak, ..., Dm.am) | CUBE(Dk.ak, ..., Dm.am)
[HAVING fAGREG (Fi.mj) ...]
[ORDER BY Dx.ax, ...]
```

FIG 3 – Syntaxe d'une requête OLAP.

La reformulation de la requête *Q* permet de générer une requête augmentée *Q'* tel que $Q \subseteq Q'$. Ceci consiste à 1) insérer dans la clause WHERE ou HAVING les prédicats sélectionnés en conjonction avec ceux de *Q*, 2) étendre *Q* par l'ajout des tables dimensions correspondant aux prédicats des dimensions non affichées par la requête (clause FROM), et 3) ajouter en conséquence les jointures correspondantes (clause WHERE).

Les prédicats sur les paramètres (année = 2010) et sur les mesures non agrégées (nb_publicis > 2) sont insérés dans la clause WHERE pour restreindre les tuples sur lesquelles seront appliqués les agrégats. Par contre, les prédicats sur les mesures agrégées (SUM(nb_publicis) > 4) sont insérées dans la clause HAVING pour éliminer les cellules du résultat qui ne sont pas conformes aux préférences de l'utilisateur.

Exemple 3. Reprenons l'exemple de décideur de l'exemple 1. La requête exprimée est :

```
SELECT TRIMESTRE, CATEGORIE, SUM (NB_PUBLIS)
FROM PUBLICATIONS AS P, DATES AS D, MANIFESTATIONS AS M
WHERE P.IDD = D.IDD AND P.IDM=M.IDM AND ANNEE = 2010
GROUP BY TRIMESTRE, CATEGORIE;
```

Personnalisation du contenu des bases de données multidimensionnelles

Sachant que la préférence P_1 apparie avec la requête. La requête formulée par le système est:

```
SELECT TRIMESTRE, CATEGORIE, SUM (NB_PUBLIS)
FROM PUBLICATIONS AS P, DATES AS D, MANIFESTATIONS AS M
WHERE P.IDD = D.IDD AND P.IDM=M.IDM AND ANNEE = 2010
AND (M. CATEGORIE = 'IEEE ' OR M. CATEGORIE = 'ACM')
```

Contenu personnalisé

4.2.2 Personnalisation contextuelle naïve

Dans cette section, nous montrons comment prendre en compte des scénarios plus complexes, notamment avec des préférences contextuelles qui sont rattachées à des contextes d'analyse de différents niveaux de détail.

Intuitivement, seules les préférences qui sont appariées complètement au CAC seront prises en compte ($cp_i = CAC$). Cependant, certaines préférences qui sont associées à une partie du CAC restent aussi valables ($cp_i \subset CAC$).

Algorithme 2. ContextMatching

Entrées :

T^{CAC} : arbre de la requête utilisateur en cours Q

$M = \{(P_i, cp_i), \dots, (P_m, cp_m)\}$: préférences contextuelles valides,

Sortie : $P^{CAND} = \{A^{cp_j}, \dots, A^{cp_k}\}$: préférences candidates pour Q

Début

1. $P^{CAND} \leftarrow \emptyset$
2. Pour chaque $M_i = (P_i, cp_i)$ faire
3. inclus \leftarrow Vrai
4. $T_i \leftarrow$ ConstruireArbre (cp_i)
5. Pour tout arc e_i de T_i faire
6. Si ($e_i \notin T^{CAC}$) Alors
7. inclus \leftarrow Faux
8. Fin Si
9. FinPour
10. Si (inclus = Vrai) Alors
11. $P^{CAND} \leftarrow P^{CAND} \cup P_i$
12. Fin Si
13. FinPour
14. Retourner (P^{CAND})

Fin.

FIG 4 – Algorithme de sélection des préférences contextuelles candidates.

Exemple 4. Considérons Q : nombre de publications des deux dernières années par type de manifestation, par équipe. Les contextes des trois préférences valides sont respectivement:

- cp_1 : analyse des publications selon l'axe manifestation,
- cp_2 : analyse du nombre de publications durant l'année en cours, et
- cp_3 : analyse du nombre de publications par statut d'auteur.

La première préférence est valable puisque le *CAC* concerne bien l'analyse des publications par manifestation. La deuxième l'est aussi parce que l'année en cours fait partie des années du *CAC*. Par contre, la troisième n'est pas pertinente par rapport au *CAC* dont le détail de l'analyse est différent de l'attribut *statut*.

Par conséquent, une préférence est candidate, si son contexte cp est plus général que *CAC*. Afin de simplifier la tâche de résolution de contexte, nous proposons de représenter le *CAC* ainsi que les contextes de préférences par des arbres respectant les relations hiérarchiques entre les différents éléments d'une analyse (Jerbi et al., 2009). La détermination des préférences contextuelles candidates revient à considérer les arbres des cp_i qui sont inclus dans l'arbre de *CAC*. FIG 4 présente l'algorithme correspondant.

Finalement, P^{CAND} est l'ensemble regroupant les préférences contextuelles candidates et toutes les préférences absolues valides.

Exemple 5. FIG 5 illustre la représentation arborescente du contexte cp_2 de l'exemple précédent, ainsi que son appariement avec l'arbre de *CAC*.

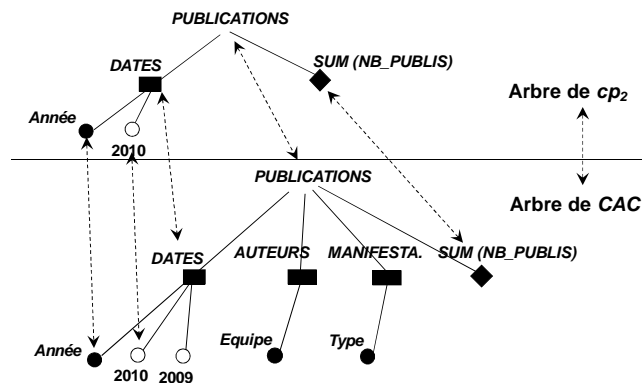


FIG 5 – Appariement de l'arbre d'un contexte de préférence avec l'arbre de *CAC*.

5 Personnalisation avancée du contenu (*OCP* avancé)

Dans cette section, nous décrivons comment peut-on prendre en compte des contraintes extérieures en intégrant un nombre limité de préférences, rendant le mécanisme de personnalisation plus flexible.

En effet, personnaliser vise à satisfaire au mieux les besoins de l'utilisateur. Théoriquement, il s'agit d'un problème d'optimisation : pour une requête Q et un utilisateur U donnés, l'objectif est de trouver les éléments du profil de U qui, une fois combinés avec Q , pourraient maximiser l'intérêt de U en Q . Certains considèrent que la personnalisation implique également l'optimisation des intérêts de l'utilisateur en fonction de certaines contraintes. Dans cette perspective, nous étendons notre cadre *OCP* afin de prendre en compte certaines contraintes de personnalisation.

Supposons que le paramètre K déterminant le nombre de préférences à prendre en compte est fixé par une contrainte de personnalisation C donnée (charge d'exécution, taille limite du

résultat, ...). Par la suite, seules les K meilleures préférences seront utilisées par le système dans son processus de personnalisation. Ceci implique :

- le tri des préférences candidates P^{CAND} pour un CAC donné, et
- la redéfinition de l'algorithme *ContextMatching* en ajoutant le paramètre K en entrée pour renvoyer seulement les K meilleures préférences candidates.

Rappelons que pour une préférence contextuelle $P_i = (pred_i, \theta_i)$, qui est associée à un contexte cp_i , θ_i représente le degré d'intérêt de $pred_i$ dans le contexte cp_i . Ceci traduit l'importance de son intégration dans le contexte d'analyse cp_i . Afin de trier les préférences candidates par rapport à CAC , il est nécessaire de calculer leurs degrés d'intérêt par rapport à CAC (relaxation de score au niveau de CAC). Ce degré est proportionnel au taux de couverture de cp_i par rapport à CAC .

Le taux de couverture est défini par le rapport entre le nombre de nœuds (ou d'arcs) de l'arbre associé à cp_i et le nombre total de nœuds (ou d'arcs) de celui de CAC :

$$Tx_Couverture = \frac{Card(c)}{Card(C)} \in [0, 1] \quad (1)$$

Remarque. Le taux de couverture est un réel entre 0 et 1 puisque $cp_i \subseteq CAC$.

Par conséquent, le score de $P_i = (pred_i, \theta_i)$ sous CAC est calculé comme suit :

$$Score (P_i)^{CAC} = \theta_i * Tx_Couverture \quad (2)$$

Les scores sont des réels entre 0 et θ_i . Plus cp_i couvre CAC , plus $Score (P_i)^{CAC}$ sera proche de θ_i .

Suite au calcul de scores, le système intègre les K meilleures préférences dans Q afin de générer Q' .

Exemple. Considérons les préférences de l'exemple 4. Supposons que les degrés d'intérêt des préférences P_1 et P_2 sont respectivement $\theta_1 = 1$ et $\theta_2 = 0.8$. Leurs taux de couverture de CAC calculés selon la formule (1) sont respectivement 0.2 et 0.5. Le score de $P_1 = 0.2$, alors que le score de P_2 s'élève à 0.4. Ainsi, dans le cas où $K=1$, c'est P_2 qui sera renvoyée (bien que initialement $\theta_2 < \theta_1$).

6 Expérimentations

Afin d'évaluer les performances de notre approche, nous avons implanté un prototype java au dessus du SGBD Oracle. Les données OLAP sont stockées dans un contexte R-OLAP. Les préférences ainsi que les contextes sont stockées sous forme de métatables au niveau de la BDM. Les données sont extraites de la BD de publication de notre laboratoire (voir FIG 1). La BDM utilisée comprend 2 faits, 3 dimensions, 5 mesures et 22 attributs. Elle contient 2 500 000 instances (tuples) de faits et 500 instances de dimensions. Nous menons des expérimentations sur des profils synthétiques générés automatiquement grâce à un générateur de profils, ainsi que des profils réels définis par des utilisateurs.

6.1 Coût de stockage des profils

Dans cette première expérience, nous avons mesuré le changement de la taille d'une BDM suite à son extension par des profils utilisateurs. Rappelons qu'une préférence peut être associée à plusieurs contextes et inversement. Nous avons utilisé 20 profils réels avec différents nombres de préférences et différents nombres de contextes par préférence. FIG 6 montre le pourcentage de la taille des profils par rapport à la taille de la BDM pour des intervalles de 200 préférences et différents nombres moyens de préférences par contextes (*nb. cps* = 0, 2, et 5). La taille des préférences absolues (avec *nb. cps* = 0) est faible, elle ne dépasse pas 0.4% de la taille de BDM initiale. Ceci est expliqué par leur faible coût de stockage puisqu'elles comprennent seulement des prédicats (chaines de caractères) et des scores entre 0 et 1 (nombres réels). Par ailleurs, pour le même intervalle de nombre de préférences, la taille des profils est plus importante lorsque les préférences sont associées à plus de contextes. La différence de taille résulte du stockage des contextes et de leurs mappings aux préférences.

FIG 6 montre une augmentation importante de la taille des profils pour un nombre de préférences entre 0 et 600. Par contre, nous observons une légère augmentation à partir de 800 préférences. En effet, le nombre total de contextes, dont la taille constitue la partie majeure de celle des profils, n'évolue pas trop à partir de ce point. En effet, les utilisateurs ne visitent pas tous les contextes d'analyse possibles, puisqu'ils naviguent à travers une partie de la constellation (Garrigós et al., 2009). Par conséquent, les nouvelles préférences sont associées avec des contextes existants (seuls les mappings sont stockés).

En conclusion, cette expérience a montré l'utilité de la personnalisation dynamique des données OLAP en terme de coût de stockage (au maximum 5% de la taille de la BDM). La génération de différentes versions de la BDM pour différents types d'utilisateurs entraîne un coût de stockage très important, alors que dans notre approche seuls les profils sont stockés.

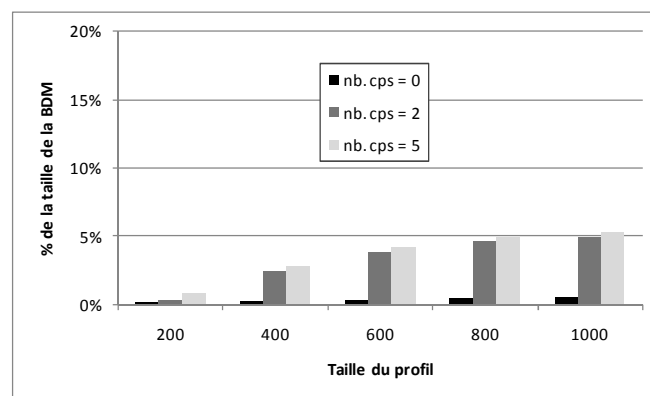


FIG 6 – Coût de stockage des profils utilisateurs.

6.2 Coût du processus de personnalisation

Cette expérience permet d'évaluer le coût de la personnalisation. Nous avons effectué plusieurs séries d'expériences pour 10 requêtes avec différentes valeurs de K et 20 profils synthétiques. Dans chaque série, nous avons calculé la moyenne du temps de calcul des re-

quêtes personnalisées, ainsi que celles des requêtes initiales. Les résultats sont présentés dans FIG 7. Toutes les durées sont exprimées en millisecondes.

Le temps global d'exécution de requêtes personnalisées est légèrement supérieur au temps de calcul de la requête initiale. En effet, comme les préférences sont intégrées sous forme de prédicats de restriction, la requête personnalisée renvoie moins de tuples que la requête initiale. La diminution de la taille du résultat de la requête permet de récompenser relativement le temps de réponse global qui aurait augmenté considérablement à cause du temps de personnalisation (sélection et intégration de préférence).

Les résultats de l'expérience sont conformes aux résultats attendus vu la nature du rôle de K qui permet de contrôler le coût du processus de personnalisation en fonction de contraintes spécifiques : la personnalisation est plus performante avec K . Notons que pour $K = 1$, le temps de la requête personnalisée est moins que celui de la requête initiale : la diminution du temps de sélection du contenu en raison de la restriction des données est plus importante que l'augmentation de temps de réponse entraîné par la sélection et l'intégration des préférences.

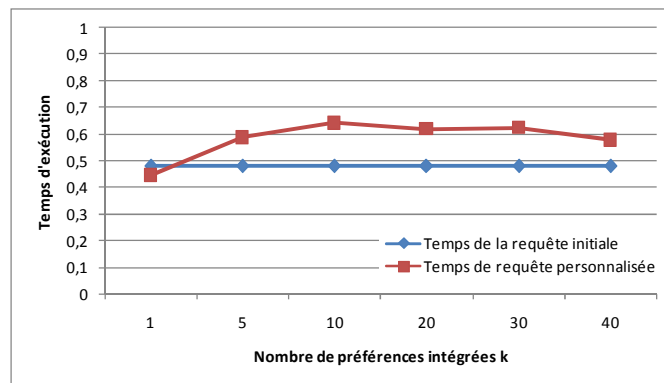


FIG 7 – Performance de la personnalisation en fonction de K .

7 Conclusion

Dans cet article, nous nous sommes focalisés sur la personnalisation du contenu OLAP afin de mieux servir les besoins d'analyse des décideurs qui partagent la même BDM.

Notre approche consiste à stocker une seule BDM et de l'enrichir de profils qui décrivent la perception individuelle de la BDM qu'a chaque utilisateur. Ceci permet d'éviter la redondance des magasins de données, et d'avoir un seul processus ETL partagé.

Les profils utilisateurs sont composés de préférences sur les instances de la BDM, qui sont associées à des contextes d'analyse plus au moins détaillés.

Une telle personnalisation du contenu permet d'offrir des services non disponibles dans les outils OLAP traditionnels, soit un accès personnalisé aux données en fonction des préférences et du contexte d'analyse de chaque décideur. Ceci permet de personnaliser, d'une part les interactions des différents utilisateurs selon leurs préférences (différents résultats pour la même requête), et, d'autre part les interactions d'un même utilisateur lorsque ce dernier change de contexte d'analyse.

Nous avons introduit, en plus du cadre *OCP* de base, une approche avancée permettant à l'administrateur et/ou à l'utilisateur de configurer le processus de personnalisation en définissant des contraintes précisant le nombre de préférences à prendre en compte. Cette option peut servir pour contrôler le coût du mécanisme de personnalisation.

Actuellement, nous mettons au point des expérimentations sur la BD des publications du laboratoire avec les différents membres (direction, administratifs, chercheurs).

Notre modèle de préférences actuel permet de définir les tranches du cube de données qui intéressent l'utilisateur afin de restituer seulement les données pertinentes. La prochaine étape serait d'envisager un mécanisme permettant à l'utilisateur de rajouter, à partir des données de l'entrepôt, des instances de mesure ou de dimension. Par exemple, un utilisateur pourrait ajouter une année, qui n'est pas chargée dans la BDM et les valeurs de mesures correspondantes. Cette année ne serait restituée que pour les requêtes de cet utilisateur.

Références

- Abbar, S., Bouzeghoub, M., Kostadinov, D., Lopes, S., Aghazaryan, A., Betge-Brezetz S. (2008). Personalized Access Model: Concepts and Services for Content Delivery Platform. Journées Bases de Données Avancées (BDA'08).
- Bellatreche, L., Giacometti, A., Marcel, P., Mouloudi, H., Laurent, D. (2005). A personalization framework for OLAP queries. DOLAP, ACM, New York, pp. 9–18.
- Favre, C., Bentayeb, F., Boussaid, O. (2007). Evolution of Data Warehouses' Optimization: a Workload Perspective. DaWaK, Springer, pp. 13–22.
- Garrigós, I., Pardillo, J., Mazón, J., Trujillo, J. (2009). A Conceptual Modeling Approach for OLAP Personalization. ER, pp. 401-414.
- Giacometti, A., Marcel, P., Negre, E. (2009) Recommending Multidimensional Queries. DaWaK, Springer-Verlag, pp. 453–466.
- Golfarelli, M., Maio, D. et Rizzi, S. (1998). Conceptual design of data warehouses from E/R schemes, 31st Hawaii International Conference on System Sciences.
- Golfarelli, M. et Rizzi, S. (2009). Expressing OLAP Preferences. SSDBM, pp 83–91
- Gray, J., Bosworth, A., Layman, A., Pirahesh, H. (1996). Data cube: A relational aggregation operator generalizing group-by, cross-tab, and sub-total. ICDE, pp. 152–159.
- Hurtado, C. A., Mendelzon, A. O., Vaisman, A. A. (1999). Updating OLAP Dimensions. DOLAP, ACM Press, pp. 60–66.
- Jerbi, H., Ravat, F., Teste, O., Zurfluh, G. (2008). Management of Context-aware Preferences in Multidimensional Databases. IEEE International Conference on Digital Information Management (ICDIM), IEEE, pp. 669-675.
- Jerbi, H., Ravat, F., Teste, O., Zurfluh, G. (2009). Preference-Based Recommendations for OLAP Analysis. DaWaK, Springer-Verlag, pp. 467–478.
- Kießling, W. (2002). Foundations of preferences in database systems. VLDB, pp. 311–322.
- Koutrika, G., Ioannidis, Y. E. (2005). Personalized Queries under a Generalized Preference Model. ICDE, pp. 841-852.
- Li, C., Chang, K. C.-C., Ilyas, I. F. (2006). Supporting ad-hoc ranking aggregates. SIGMOD, pp. 61–72.

Personnalisation du contenu des bases de données multidimensionnelles

- Ravat, F., Teste, O., (2009). Personalization and OLAP databases. In: Volume New Trends in Data Warehousing and Data Analysis of Annals of Information Systems, Springer, Heidelberg, pp. 71–92.
- Rizzi, S. (2007). OLAP preferences: a research agenda. DOLAP, pp. 99–100.
- Xin, D., Han, J., Cheng, H., Li, X. (2006). Answering top-k queries with multi-dimensional selections: The ranking cube approach. VLDB, pp. 463–475.

Summary

OLAP systems are based on multidimensional databases (MDB) that are extracted from the data warehouse. The users sharing the same MDB may have different specific analysis needs. We propose a personalization framework for MDB management systems based on user profiles. Such profiles consist of contextual preferences that are used to adapt the MDB content to the perception of each user. At query time, the system rewrites the query taking into account the related user preferences in order to simulate its performance upon an individual content.