UNIVERSITE TOULOUSE III – PAUL SABATIER
U.F.R. MATHEMATIQUES INFORMATIQUE GESTION (MIG)
ECOLE DOCTORALE MATHEMATIQUE, INFORMATIQUE ET
TELECOMUNICATION DE TOULOUSE (MITT)

THÈSE

en vue de l'obtention du

DOCTORAT DE L'UNIVERSITE DE TOULOUSE
délivré par l'Université Toulouse III – Paul Sabatier

Discipline : Informatique

présentée et soutenue
par

Ronan Tournier

le 13 décembre 2007

Titre :

Analyse en ligne (OLAP) de documents

Jury :

Omar Boussaid	Maître de conférences habilité, université Lyon 2	Rapporteur
Patrick Marcel	Maître de conférences, université de Tours	Examinateur
Franck Ravat	Maître de conférences, université Toulouse 1	Examinateur
Michel Schneider	Professeur, université Clermont 2	Rapporteur
Chantal Soulé-Dupuy	Professeur, université Toulouse 1	Examinatrice
Olivier Teste	Maître de conférences, université Toulouse 3	Examinateur
Gilles Zurfluh	Professeur, université Toulouse 1	Directeur

IRIT, INSTITUT DE RECHERCHE EN INFORMATIQUE DE TOULOUSE (UMR 5505),
Université Paul Sabatier Toulouse 3, 118 route de Narbonne, F-31062 Toulouse Cedex 9, France.

Résumé (court)

Les entrepôts de données et les systèmes d'analyse en ligne OLAP (On-Line Analytical Processing) fournissent des méthodes et des outils permettant l'analyse de données issues des systèmes d'information des entreprises. Mais, seules 20% des données d'un système d'information est constitué de données analysables par les systèmes OLAP actuels. Les 80% restant, constitués de documents, restent hors de portée de ces systèmes faute d'outils ou de méthodes adaptés. Pour répondre à cette problématique nous proposons un modèle conceptuel multidimensionnel pour représenter les concepts d'analyse. Ce modèle repose sur un unique concept, modélisant à la fois les sujets et les axes d'une analyse. Nous y associons une fonction pour agréger des données textuelles afin d'obtenir une vision synthétique des informations issues de documents. Cette fonction résume un ensemble de mots-clefs par un ensemble plus petit et plus général. Nous introduisons un noyau d'opérations élémentaires permettant la spécification d'analyses multidimensionnelles à partir des concepts du modèle ainsi que leur manipulation pour affiner une analyse. Nous proposons également une démarche pour l'intégration des données issues de documents, qui décrit les phases pour concevoir le schéma conceptuel multidimensionnel, l'analyse des sources de données ainsi que le processus d'alimentation. Enfin, pour valider notre proposition, nous présentons un prototype.

Mots-clefs

OLAP, Document XML, Galaxie, Modélisation multidimensionnelle, Démarche de modélisation, Base de données multidimensionnelle, Entrepôts de données, Entrepôts de documents.

Summary

Data warehouses and OLAP systems (On-Line Analytical Processing) provide methods and tools for enterprise information system data analysis. But only 20% of the data of a corporate information system may be processed with actual OLAP systems. The rest, namely 80%, i.e. documents, remains out of reach of OLAP systems due to the lack of adapted tools and processes. To solve this issue we propose a multidimensional conceptual model for representing analysis concepts. The model rests on a unique concept that models both analysis subjects as well as analysis axes. We define an aggregation function to aggregate textual data in order to obtain a summarised vision of the information extracted from documents. This function summarises a set of keywords into a smaller and more general set. We introduce a core of manipulation operators that allow the specification of analyses and their manipulation with the use of the concepts of the model. We associate a design process for the integration of data extracted from documents within an OLAP system that describes the phases for designing the conceptual schema, for analysing the document sources and for the loading process. In order to validate these propositions we have implemented a prototype.

Keywords

OLAP, XML document, Galaxy, Multidimensional modelling, modelling process, Multidimensional database, Data warehouse, Document warehouse.

Résumé

Les entrepôts de données et les systèmes d'analyse en ligne OLAP (On-Line Analytical Processing) fournissent des méthodes et des outils puissants permettant l'analyse de données issues des systèmes d'information des entreprises. Mais, seules 20% des données d'un système d'information d'entreprise est constitué de données analysables par les systèmes OLAP actuels. Les 80% restant sont constitués de documents (rapports, notes, articles…). Ces documents restent hors de portée des systèmes OLAP faute d'outils ou de méthodes adaptés. Dans le cadre des systèmes d'aide à la décision, l'omission des données contenues dans ces documents peut mener à des analyses imprécises voire erronées engendrant un risque d'erreur pour la prise de décision. Les documents représentent une capitalisation de connaissances, au même titre que les données analysables du système d'information représentent une capitalisation d'évènements (ventes, achats…). Il est donc naturel de prévoir l'ajout dans l'analyse en ligne des documents. De nos jours, un décideur maîtrise très bien les processus OLAP. Ainsi se pose la question : comment lui fournir un environnement permettant l'analyse en ligne de 100% des données disponibles avec des méthodes et des moyens qu'il maîtrise ?

Pour répondre à cette problématique nous proposons un modèle conceptuel multidimensionnel. Contrairement aux modèles multidimensionnels classiques qui s'appuient sur la dualité de concepts fait / dimension, notre modèle ne repose que sur un unique concept permettant de modéliser à la fois les sujets et les axes d'une analyse. Le modèle fournit au décideur une vision des éléments multidimensionnels disponibles pour exprimer les analyses.

Les analyses multidimensionnelles reposent sur une capacité à résumer les informations en les agrégeant avec des fonctions d'agrégation. Toutefois, il n'existe pas de moyen dans un environnement OLAP pour agréger des données textuelles qui représentent le cœur des documents analysés. Ainsi nous proposons une fonction capable d'agréger des données textuelles pour permettre d'obtenir une vision synthétique des informations. Cette fonction d'agrégation cherche à résumer un ensemble de mots-clefs par un ensemble plus petit et plus général.

Afin de pouvoir spécifier des analyses sur les données issues de documents, nous introduisons des opérations permettant la manipulation des concepts du modèle. Dans un premier temps, ces opérations permettent la spécification d'une analyse multidimensionnelle à partir des éléments représentés par le modèle. Dans un second temps, nous définissons un noyau d'opérations élémentaires permettant la modification d'une analyse afin que le décideur puisse affiner ses observations et prendre la meilleure décision possible.

Nous proposons une démarche pour l'intégration des données issues de documents dans un système OLAP, car elle diffère des processus classiques. Cette méthode décrit les phases nécessaires pour concevoir le schéma conceptuel multidimensionnel à partir des besoins d'analyse, l'analyse des sources de données qui serviront à alimenter le système ainsi que le processus final d'alimentation.

Enfin, pour valider notre proposition, nous présentons un prototype écrit en Java. Des structures multidimensionnelles, implantées au sein d'un SGBD, représentent les concepts manipulés par le décideur. Les analyses sont spécifiées par l'intermédiaire d'une interface et les données de ces analyses sont synthétisées et restituées à l'utilisateur.

Remerciements

Je tiens à remercier très sincèrement Claude Chrisment et Gilles Zurfluh, responsables de l'équipe Systèmes d'Informations Généralisées (SIG) pour m'avoir si bien accueilli au sein de leur équipe afin que je puisse mener à bien cette thèse.

Je remercie sincèrement Omar Boussaïd, maître de conférence habilité de l'Université Lyon 2 pour avoir accepté d'être rapporteur de ce mémoire, pour ses remarques pertinentes et pour sa participation en tant que membre de mon jury.

Je remercie également Michel Schneider, professeur de l'ISIMA de l'Université Clermont 2, pour avoir accepté d'être rapporteur de ce mémoire. Je le remercie également pour ses remarques pertinentes ainsi que pour sa participation au jury.

Ma reconnaissance va envers Gilles Zurfluh, professeur de l'Université Toulouse 1, pour son soutien, ses conseils et ses critiques et pour m'avoir laissé une grande liberté dans mes travaux. Je le remercie pour m'avoir permis de débuter dans le monde de la recherche.

Je remercie Franck Ravat et Olivier Teste, maîtres de conférences pour leur encadrement, leur collaboration et surtout pour leur soutient sans faille. Je les remercie d'avoir eu confiance en ces travaux ainsi que pour leur très grande disponibilité.

Je remercie Patrick Marcel, maître de conférence de l'Université de Tours, pour avoir accepté de participer à mon jury.

Chantal Soulé-Dupuy, professeur de l'Université Toulouse 1, pour avoir accepté d'être membre de mon jury. Je la remercie également pour ses remarques encourageantes et son soutien.

Je remercie les enseignants d'informatique de l'Université Toulouse 1 pour m'avoir permis d'enseigner. Je les remercie de la confiance qu'ils m'ont faite.

Je tiens à remercier du fond du cœur les membres de l'équipe SIG pour leur soutien. Je remercie Mohand Boughanem pour sa jovialité, sa constante bonne humeur et ses conseils pertinents ; Bernard Dousset pour ses cafés et ses suggestions intéressantes ; Josiane Mothe pour ses remarques et critiques pertinentes ; et enfin Florence Sèdes pour son soutien et ses remarques qui permettent de continuer même quand tout semble s'effondrer. Je remercie Nathalie et Karen pour leur aide précieuse. Je remercie également du fond du cœur les « trois mousquetaires » Max, Gilles et Olivier pour leur amitié et leur soutien, mais aussi pour leurs remarques, leurs nombreuses critiques et leur bonne humeur. Je remercie les autres membres de l'équipe SIG, Guillaume, Sabine (pour ses piqûres de rappel) ainsi que tous les autres…

Je tiens aussi à remercier l'ensemble du personnel de l'IRIT, pour leur gentillesse, leur disponibilité et surtout leurs sourires !

Une mention spéciale pour les autres rédacteurs de l'été et de la rentrée : Franck Ravat, Jean-Denis Durou et Cécile Favre qui ont aussi du résumer plusieurs années de travail en quelques pages…

Je voudrais aussi remercier ceux qui m'ont fait apprécier les bases de données : Claude Chrisment, Michel Tuffery et Olivier Teste. Je tiens également à remercier ceux qui ont su me faire apprécier l'informatique : Patrick Bergougnoux, Alain Crouzil, Patrick Debord et Daniel Marquié.

Je remercie également tous ceux qui ont du me supporter tout au long de ma thèse, en général par mon absence. Notamment, ma famille avec mon père Bernard et mon frère Yannick ainsi qu'Emma pour sa petite dédicace (http://luciolland.canalblog.com). Je remercie enfin mes amis pour leur soutien surtout ceux que je n'ai pu aider cet été de 2007 à cause de ma brillante indisponibilité…

Sommaire général

À André Segorb,
Architecte des Écoles de Paris.

CHAPITRE I
Contexte des travaux

Résumé du chapitre

Ce chapitre présente le contexte de ce mémoire de thèse. Premièrement, il expose l'environnement des systèmes d'aide à la prise de décision en détaillant leur architecture. Puis le chapitre présente nos motivations pour l'intégration de données issues de documents au sein des systèmes d'analyse multidimensionnels OLAP. Cette intégration est envisagée grâce aux apports du format XML qui permet de structurer les données issues de documents. Partant de ces motivations et des possibilités offertes par XML, la problématique du mémoire de thèse est présentée. La section se termine par le plan du mémoire.

Sommaire

CHAPITRE I : Contexte des travaux

« Where is the wisdom? Lost in the knowledge. »
« Where is the knowledge? Lost in the information… »

— T.S. Elliot.

« Where is the information? Lost in the data. »
« Where is the data? Lost in the #@$$%?!& database! »

— Joe Celko (concepteur et consultant en bases de données).

1 Introduction

L'essor des technologies de l'information, avec l'avènement d'Internet et des réseaux, a accru le volume des informations disponibles de manière considérable. Désormais, les entreprises font face à un volume croissant de données qui transitent au sein de leur système d'information et cette masse d'informations rend difficile leur exploitation.

Pour faire face au problème de volume, les systèmes d'aide au processus de prise de décision (ou systèmes d'aide à la décision) ont été mis en place au sein des entreprises. Ces systèmes permettent un traitement synthétique de l'information pour faciliter les prises de décisions. Les résultats de ces systèmes sont exploités par les décideurs en vue d'effectuer des analyses pour piloter au mieux les entités économiques dont ils sont responsables. Un décideur peut ainsi confirmer sa ligne de conduite ou bien changer radicalement de stratégie pour s'adapter aux évolutions de l'environnement économique.

Plan du chapitre. Ce chapitre présente le contexte du mémoire de thèse, à savoir, les systèmes d'aide à la prise de décision et les outils qui les composent. La section 2 présente le contexte de ces systèmes et leur architecture. Toutefois, les systèmes actuels sont limités car ils ne traitent qu'une partie des données des systèmes d'information des entreprises. Les données non traitées sont principalement des données textuelles contenues dans des documents. La section 3 expose l'apport du format de stockage XML et l'intégration des documents dans les systèmes d'aide à la prise de décision. La section 4 poursuit sur notre problématique détaillée de ce mémoire de thèse. Enfin, la section 5, présente le plan de la thèse.

2 Les systèmes d'aide à la décision

De nos jours, les décideurs ont besoin d'une vision synthétique et globale des informations circulant dans leur organisation afin de guider et adapter leur prise de décision. Pour faciliter ce processus, ils emploient des systèmes d'aide à la décision. Ces outils permettent aux décideurs d'avoir une vision globale sur les activités d'une entreprise par un accès rapide et interactif à un ensemble de vues des données organisées pour refléter l'aspect multidimensionnel des données de l'entreprise [Colliat, 1996].

Définition. Un *système d'aide à la décision* est l'ensemble des outils informatiques (matériels et logiciels) qui permettent l'analyse des données opérationnelles issues du système d'information des entreprises. Ces données sont transformées en une vision orientée décideur puis analysées au moyen de manipulations et restitutions adaptées.

Comme l'illustre la Figure 1, trois catégories d'outils sont employés :

- Les outils d'*extraction*, de *transformation* et de *chargement* ou ETL (acronyme de *Extract*, *Transform*, *Load*) des données opérationnelles pour alimenter et rafraîchir les données contenues dans le système d'aide à la décision.
- Les outils de *stockage* et de *traitement* des données décisionnelles, extraites des données opérationnelles.
- Les outils de *restitution* et d'*analyse* des données décisionnelles sous une forme adaptée aux décideurs.

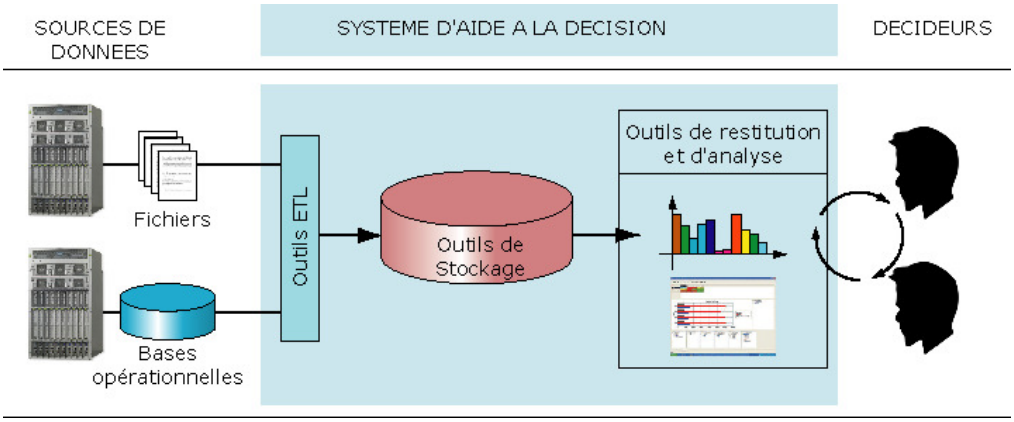


Figure 1 -Le système d'aide à la décision.

Au sein de cet environnement, nous considérons une architecture à plusieurs niveaux, décrite dans la sous-section suivante où la dualité des éléments de stockage sera présentée.

2.1 Architecture de systèmes d'aide à la décision

En 1993, E.F. Codd, suggère l'emploi de systèmes qui permettent d'améliorer les processus de prise de décision par la consultation et l'analyse de grandes masses de données : les systèmes d'analyse en ligne, « On-Line Analytical Processing » (OLAP) [Codd, 1993]. Ces systèmes OLAP ont pour but de fournir une réponse rapide à des requêtes analytiques de nature multidimensionnelle. Les requêtes analytiques représentent des analyses qui s'appuient sur un outil de centralisation des données appelé : entrepôt de données [Kimball, 1996], [Inmon, 1996]. À partir de ces éléments, une architecture décisionnelle à quatre niveaux est considérée (cf. Figure 2) :

- Les modules d'*extraction* : ils permettent l'alimentation du système décisionnel à partir des sources de données opérationnelles. Les sources étant souvent hétérogènes et réparties, des outils emploient des processus de transformation pour permettre l'intégration et l'alimentation des données dans l'espace de stockage ;
- l'*entrepôt de données* (premier niveau en terme de stockage et pouvant être optionnel) est un espace de stockage centralisé et uniforme. Les données y sont regroupées et restructurées afin de présenter une vue unifiée facilitant leur l'accès. Ces données représentent une vision horizontale des différents corps de métier d'une entreprise ;
- les *magasins de données*, second niveau en terme de stockage, ils sont destinés à permettre l'analyse des données. Ces environnements représentent tout ou partie des données de l'entrepôt (ou celles des sources si l'on se passe de l'entrepôt) selon une structuration adaptée aux besoins d'analyses. Les données des magasins sont

structurées via une modélisation multidimensionnelle et gérées par des bases de données multidimensionnelles (BDM) [Kimball, 1996]. Par opposition à l'entrepôt, un magasin est une vision verticale des données d'une entreprise ;

- l'*analyse multidimensionnelle* et la *restitution* : les données issues des magasins de données sont restituées au décideur via des outils d'analyse ou de « reporting ».

Cette architecture sépare clairement l'espace de stockage général (l'entrepôt), où les données sont présentées de manière uniformisée, et l'espace spécifique (le magasin), où les données sont orientées en fonction des analyses souhaitées.

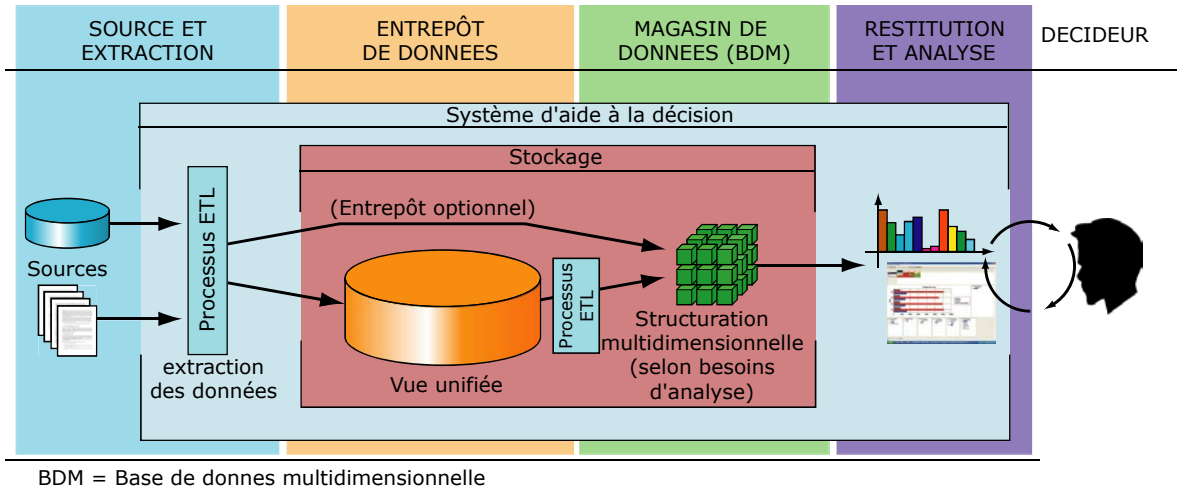


Figure 2 - Architecture d'un système décisionnel.

La section suivante, présente plus en détail les espaces de stockage, à savoir, l'entrepôt de données et les magasins.

2.2 Stockage : Entrepôt de données

Dans les années 90, Bill Inmon, est l'un des premiers à avoir employé le terme d'entrepôt de données. Ce dernier le définit comme : « une collection de données orientées sujet, intégrées, variant selon le temps et non volatiles, qui sert de support au processus de prise de décision des acteurs du management [(les décideurs)] ». Plus précisément la collection de données est [Inmon, 1996] :

- *Orientée sujet* : les données constituent des granules d'information concernant des sujets d'analyse plutôt que les opérations de gestion se déroulant au sein de l'entreprise.
- *Intégrée* : les données sont centralisées dans un entrepôt à partir d'un ensemble de sources de données variées. Les données sont fusionnées et agencées au sein d'une vision cohérente.
- *Variant selon le temps* : Toutes les données d'un entrepôt sont identifiées par des périodes temporelles spécifiques. On parle d'historisation des données [Kimball, 1996], [Inmon, 1996], [Teste, 2000]
- *Non volatile* : Les données d'un entrepôt sont stables. Il est possible d'ajouter des données, mais on ne modifie pas les données déjà intégrées. Il est toutefois possible de les archiver.

De son côté, Ralph Kimball a fourni une définition plus simple d'un entrepôt de données, mais qui n'en est pas moins précise : « un entrepôt de données est une copie des données transactionnelles d'une entreprise structurée de manière spécifique pour l'interrogation et l'analyse. » [Kimball, 1996].

Définition. Un *entrepôt de données* est l'espace de stockage centralisé d'un extrait des sources de données pertinentes pour les décideurs. Son organisation doit faciliter la gestion des données sous la forme d'une vision unifiée et doit permettre la conservation des évolutions nécessaires pour les prises de décisions.

Un entrepôt de données offre une vision uniforme des données qui seront extraites en fonction de besoins d'analyse pour alimenter les magasins de données.

2.3 Stockage : Magasin de données

La structuration des données de l'entrepôt en fonction des besoins d'analyse s'effectue au sein d'un magasin de données que nous définissons comme suit :

Définition. Un *magasin de données* est un extrait de l'entrepôt conforme à des besoins d'analyse particuliers et organisé selon un modèle adapté aux outils d'analyse et d'interrogation décisionnelle (un modèle en cube ou multidimensionnel). Le magasin est généralement stocké au sein d'une base de données multidimensionnelle (BDM).

Les magasins de données emploient généralement une modélisation dite en *étoile* ou en *flocon* pour représenter les sujets et les axes d'analyse des données [Kimball, 1996]. Au niveau logique, la base de données multidimensionnelle hébergeant le magasin de données est souvent structurée avec une technologie relationnelle dite « relational-OLAP » (*ROLAP*) ou multidimensionnelle : « multidimensional-OLAP » (*MOLAP*) [Chaudhuri & Dayal, 1997].

Par la suite, le terme *magasin de données classique* désignera un magasin de données modélisé selon un schéma en étoile ou flocons et bâti sur une architecture ROLAP, MOLAP ou similaire tel que ceux définit dans [Kimball, 1996] ou [Ravat et al., 2007e].

2.4 Restitution : Analyse OLAP multidimensionnelle

Les magasins de données reposent sur une modélisation multidimensionnelle des données extraites de l'entrepôt. Ceci permet de représenter les données d'un magasin sous la forme de points dans un espace à plusieurs dimensions avec la métaphore de *cube* ou d'*hypercube* de données. Cette modélisation permet l'expression d'analyses en ligne (OLAP) multidimensionnelles.

Exemple. La Figure 3 présente un exemple de cube, qui permet l'analyse des activités d'une chaîne de revendeurs informatique. Il modélise les activités de ventes de cette chaîne. Les *Quantités* et les *Montants* des ventes représentent les indicateurs d'analyse disponibles du sujet *Ventes*. Ces indicateurs peuvent être analysées en fonction de trois dimensions : les *Magasins* où ont été effectuées les ventes, les *Dates* correspondantes aux dates de ventes et les *Produits* vendus. Chacune de ces dimensions dispose de plusieurs niveaux de détail (ville, pays, continent…) et permettent d'obtenir une vision plus ou moins détaillée lors des analyses.

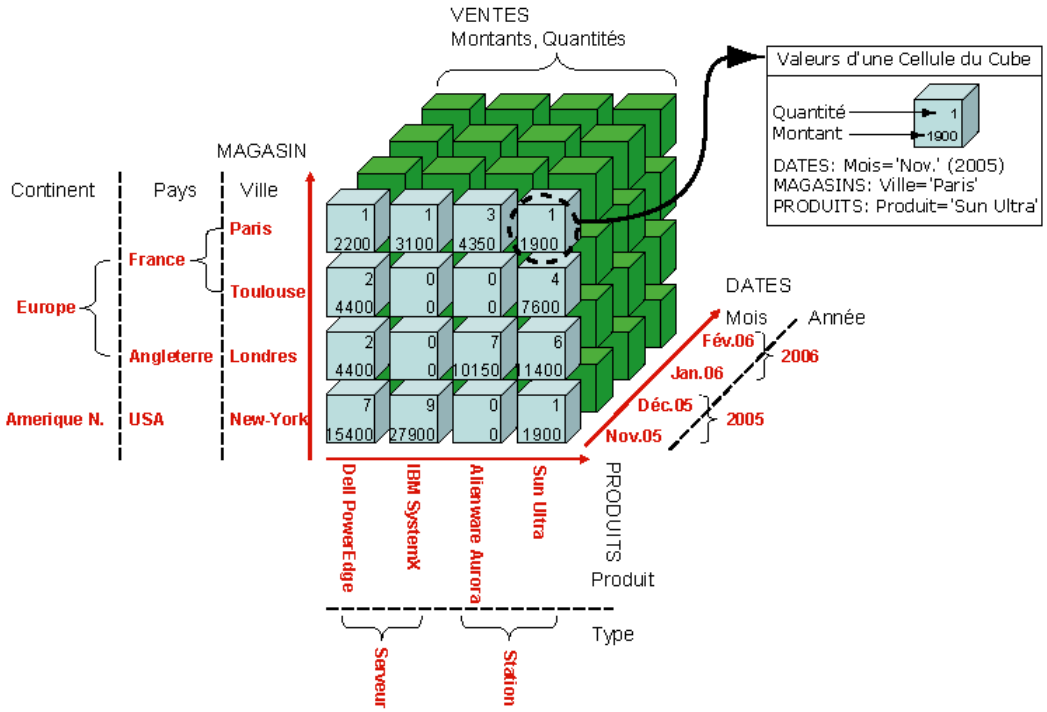


Figure 3 - Exemple d'analyse des ventes d'une chaîne de magasins informatiques (vision en cube).

La modélisation en cube est très limitée en terme de représentation des dimensions [Torlone, 2003]. Pour concevoir des magasins de données plus élaborés, des structures avancées ont été définies. Ces structures permettent la modélisation de sujets d'analyse appelés *faits*, et d'axes d'analyse appelés *dimensions* [Kimball, 1996], [Abelló et al., 2001a], [Abelló et al., 2001b]. Les faits sont des regroupements d'indicateurs d'analyse appelés *mesures*. Les dimensions sont composées d'attributs, appelés *paramètres*, agencés de manière hiérarchique, qui modélisent les différents niveaux de détails des axes d'analyse. Un fait et ses dimensions associées composent un *schéma en étoile* [Kimball, 1996]. Les données des mesures sont appelées *données factuelles* car elles représentent un évènement. Elles correspondent aux données des cellules du cube (cf. Figure 3) qui seront analysées en fonction des données des axes d'analyse, appelées *données dimensionnelles*.

Exemple. Le schéma multidimensionnel associé à l'exemple précédent est présenté en Figure 4 en employant le formalisme défini dans [Ravat et al., 2007e]. Un modèle multidimensionnel permet une représentation plus précise des dimensions avec les différents niveaux de détails pour les analyses : les paramètres.

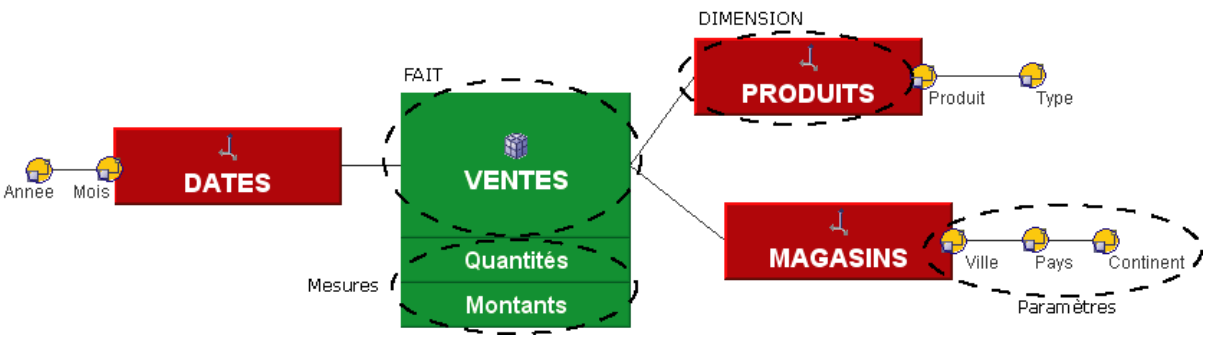
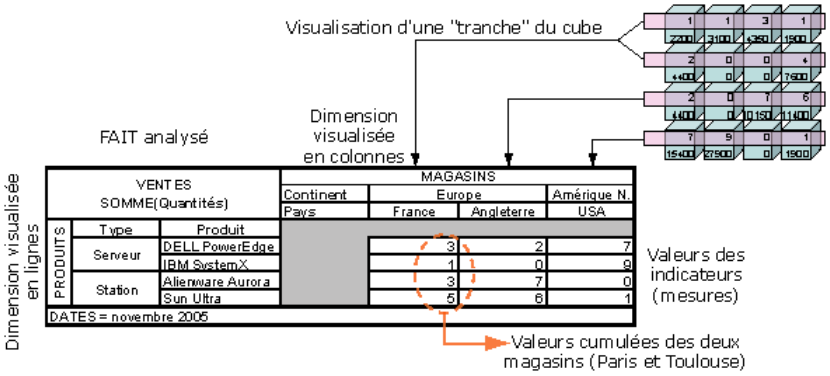


Figure 4 – Exemple de visualisation multidimensionnelle de la précédente analyse.

Une *analyse multidimensionnelle* est une requête analytique sur les données d'un magasin de données. Généralement, une table bidimensionnelle est employée pour afficher les données résultantes de la requête.

Exemple. Dans l'exemple qui suit (cf. Tableau 1) un décideur analyse les quantités de produits vendus en fonction du pays des magasins où les ventes ont été effectuées. Cette analyse est restreinte au mois de novembre 2005. Il s'agit des données extraites de la « tranche » du cube de la Figure 3, représentée par des cellules bleues (gris clair).

Tableau 1 - Exemple d'analyse multidimensionnelle de quantités de produits vendus.



L'environnement d'analyse en ligne OLAP repose sur une analyse de données numériques. La section suivante expose les limites de cette approche classique au regard des données disponibles au sein d'un système d'information d'une entreprise.

3 Analyse de documents : au-delà des nombres

Les entrepôts de données et les systèmes OLAP (On-Line Analytical Processing) fournissent des méthodes et des outils puissants permettant l'analyse de données transactionnelles [Sullivan, 2001]. L'analyse reposant sur des bases de données multidimensionnelles (BDM), avec des sujets d'analyse numériques, est une tâche relativement bien maîtrisée de nos jours [Sullivan, 2001]. Ces bases de données multidimensionnelles sont construites à partir de données transactionnelles extraites des systèmes d'information des entreprises. Cependant, seules 20% des données d'un système d'information sont des données transactionnelles et peuvent être traitées par un système OLAP [Tseng & Chou, 2006].

Les systèmes d'aide à la décision n'ont fait qu'entrevoir la partie émergée de l'iceberg informationnel que représente l'aide à la décision [Tseng & Chou, 2006]. Les 80% restant des données d'un système d'information représentent les documents électroniques d'une entreprise, tels que des rapports, des notes, des articles (…) [Sullivan, 2001]. Principalement constitués de texte, ces documents restent hors de portée des systèmes d'aide à la décision faute d'outils de traitement et d'analyse ou encore de méthode d'intégration adaptées [Sullivan, 2001].

Les documents représentent une capitalisation de connaissances, au même titre que les transactions dans les bases de données. Un décideur de nos jours maîtrise les processus d'analyse en ligne, il est donc naturel de prévoir l'ajout des documents dans ce processus. Le contenu de documents étant peu structuré, ceci explique la difficulté de l'intégration de ces derniers dans les systèmes d'aide à la décision.

Ainsi se pose la question : comment fournir au décideur un environnement permettant l'analyse en ligne de 100% des données d'un système d'information avec des méthodes et des moyens qu'il maîtrise (les processus OLAP) ?

3.1 XML : une solution d'intégration de documents

« *XML permet […] de représenter à la fois du texte non structuré, du texte ayant une structure partielle et jusqu'à des données fortement structurées telles que des tuples d'une source relationnelle.* » [Abiteboul, 2003]. Les données issues de documents sont peu structurées à première vue, mais en réalité elles ont hérité d'une structure hiérarchisée issue des documents papiers [Sullivan, 2001]. De plus, par exemple, le langage naturel structure ces données en phrases regroupées en paragraphes. Ainsi l'emploi du langage XML permet de représenter cette structure.

Le langage XML (eXtensible Markup Language) est un format de données très flexible dérivé du SGML (ISO 8879) [W3C-XML, 2006]. Conçu originellement pour faire face aux problèmes posés par la publication de données électroniques à grande échelle, XML joue désormais un rôle de plus en plus important dans l'échange de données sur le Web et au sein des entreprises [W3C-XML, 2006]. Plus précisément, il s'agit d'un format de stockage générique spécifié par le W3C[1]. Ce format, auto-descriptif, permet la structuration de son contenu au moyen dune grammaire composée de balises. Le contenu d'un document au format XML se retrouve réparti en éléments délimités par des balises, chacune éventuellement associées à des attributs. Les balises sont imbriquées les unes dans les autres et composent une structure arborescente. La flexibilité du format permet la représentation de données peu structurées, semi-structurées ou encore fortement structurées.

Exemple. Un élément est composé par des données contenues entre deux *balises* : une balise ouvrante pour le début de l'élément et une fermante spécifiant la fin de l'élément :

<*BALISE_1*> Contenu de l'élément 1 </*BALISE_1*>

Les balises sont spécifiées entre les symboles « < » et « > ». Le « / » au sein d'une balise spécifie la fermeture de cette balise. Un élément est composé soit d'un contenu, soit de sous-éléments. Les éléments composant un document XML sont ainsi agencés de manière arborescente :

<*BALISE_2*>
  <*BALISE_3*> Contenu de l'élément 3 </*BALISE_3*>
  <*BALISE_4*> Contenu de l'élément 4 </*BALISE_4*>
</*BALISE_2*>

Les balises peuvent contenir des éléments complémentaires : des *attributs*.

<*BALISE ATTRIBUT_1*="contenu de l'attribut" *ATTRIBUT_2*=… > … </*BALISE*>

Le langage XML dispose d'un formalisme dédié à la déclaration de structure : DTD (Document Type Définition), [W3C-XML, 2006]. Ce langage permet de définir l'agencement hiérarchique des éléments et des attributs au sein d'un document XML.

Exemple. La Figure 5 présente un exemple de document XML. Ce document commence par un en-tête auto-descriptif qui permet la spécification de la structure du document. La partie inférieure du document présente le contenu du document structuré selon l'en-tête. L'en-tête du document XML est une DTD, elle est déclarée, au sein de l'élément :

[1] W3C : World Wide Web Consortium : http://www.w3.org/

<!DOCTYPE[…]>

Au sein de cette DTD il est possible de déclarer des éléments parent d'autres élément :

!ELEMENT *nom_élément* (*élément_fils_1*, *élément_fils_2*,…)

Ou encore des éléments terminaux (contenant des données représentées par le type de données générique #PCDATA) :

!ELEMENT *nom_élément* (#PCDATA)

Il est aussi possible de déclarer des attributs associés à des éléments :

!ATTLIST *nom_élément nom_attribut_1* CDATA #REQUIRED

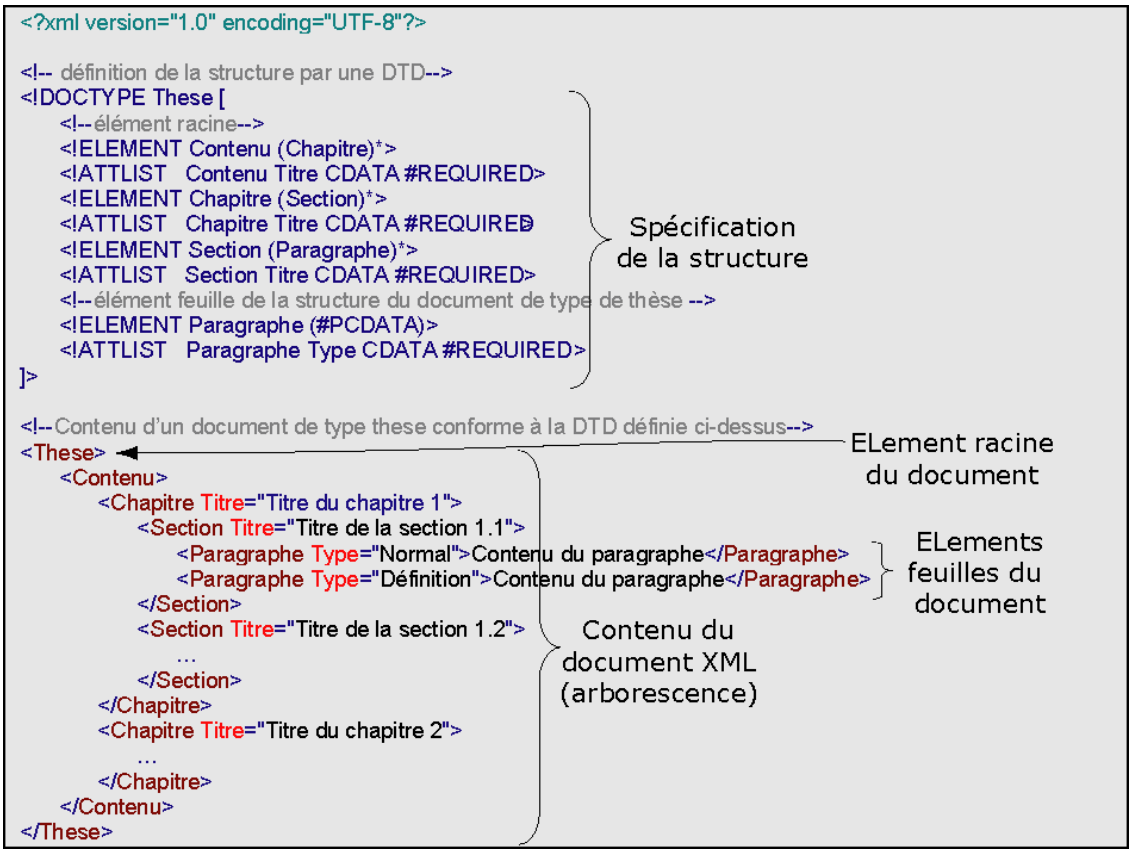


Figure 5 - Exemple de document XML intégrant sa propre définition de structure.

Exemple. Dans l'exemple de la Figure 5, une thèse est constituée d'éléments imbriqués : des chapitres, contenant des sections, contenant des paragraphes, constitués de fragments de texte. Les chapitres et les sections sont associés à un titre et les paragraphes sont associés à un type via l'emploi d'attributs.

La structure d'un document XML peut aussi être représentée sous forme arborescente (cf. Figure 6). Des cardinalités entre les éléments peuvent être ajoutées pour plus de précisions.

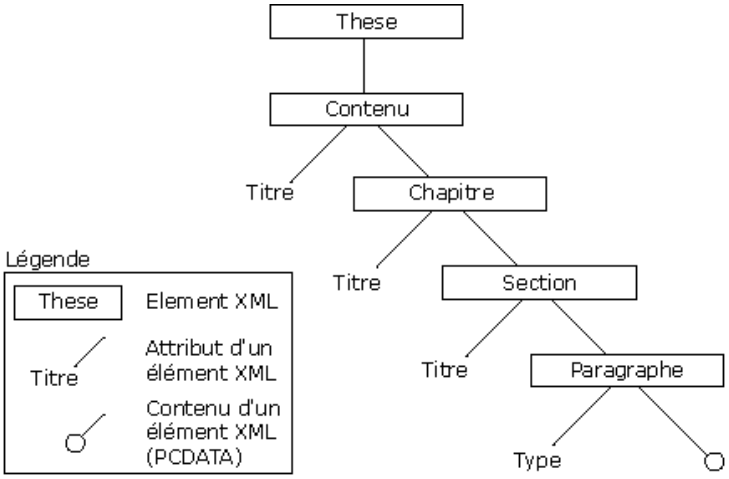


Figure 6 – Représentation arborescente de la structure du document XML précédent.

Le langage XML est associé à plusieurs langages tels que XSLT, XPath et XQuery qui sont présentées brièvement ci-après.

Pour permettre la modification et l'altération de la structure et du contenu d'un document XML, un langage de transformation a été proposé : XSLT (Extensible Stylesheet Language Transformation), [W3C-XSL, 1999]. Ce langage peut être employé pour permettre le remaniement des structures et des données au format XML.

Les données XML étant structurées de manière arborescente un langage d'expression de chemins permet la spécification simple de requêtes au sein de la structure arborescente du document : XPath [W3C-XQuery, 2007].

Exemple. Le langage XPath permet d'accéder à des éléments particuliers dans un document XML. Par exemple, l'instruction suivante permet de désigner tous les éléments section de la thèse :

```
/These/Contenu/Chapitre/Section   (en chemin absolu)
//Section (en chemin relatif)
```

Exemple. Il est possible d'accéder à une section particulière, en spécifiant le contenu d'un attribut associé à la balise section. Dans l'exemple suivant, il s'agit d'obtenir la section dont le titre est « Introduction » :

```
/These/Contenu/Chapitre/Section[@titre="Introduction"]
```

Ce langage d'expression de chemins est complété par un langage de requête plus complet : XQuery (XML Query) [W3C-XQuery, 2007] qui emploie les expressions de chemins XPath pour permettre la spécification de requêtes complexes sur la structure ou le contenu de documents XML.

Exemple. Dans l'exemple de requête XQuery suivant, pour le chapitre du document *thèse.xml* (cf. Figure 5) dont le titre est « *introduction* », le système retourne le texte de chaque section du chapitre entre des balises <Resultat>.

```
xquery version "1.0";
for $chap in doc("these.xml")/These/Contenu/Chapitre[@titre="introduction"]
return <Resultat>{$chap/Section/text()}</Resultat>
```

Le format XML, et ses langages associés, appliqués aux documents textuels ont fourni un environnement permettant la structuration et l'exploitation de ces documents. Le langage

XML possède des avantages intéressants : une absence (possible) de schéma prédéfini et des mécanismes pour représenter la structure et le contenu au sein d'un même document.

Le format XML est une solution permettant de représenter la structure des documents qui n'était pas représentée jusqu'à présent. Les sources de données XML ne cessent de croître en nombre et en disponibilités : de plus en plus de données sont disponibles sous ce format sur le Web et dans les entreprises. Ainsi intégrer des documents au sein des systèmes décisionnels est envisageable. Mais dans ce vaste environnement, « *il y a document et document*… ». En d'autres termes, il existe des documents XML plus délicats à intégrer que d'autres…

3.2 Différents types de documents

Peu à peu les documents structurés et semi-structurés furent intégrés au sein des entrepôts de données (entrepôt de document ou « document warehouse ») [Sullivan, 2001], avec par exemple Xyleme[2].

Définition. Un *document XML* représente une unité d'information au format XML, à savoir une chaîne de caractères XML délimité par deux balises. Cette unité peut être elle-même composée de plusieurs sous-éléments (la chaîne de caractères pouvant elle-même comporter des balises).

Au sein des systèmes d'information des entreprises, il est possible de rencontrer plusieurs types de documents XML. La grande flexibilité du format XML permet le stockage de diverses données, par exemple des factures, mais aussi des rapports ou encore des ouvrages complets. Nous désirons différencier clairement deux types de documents :

- Les documents XML assimilables au contenu d'une base de données : par exemple des données issues d'un tableur, d'un Web service, l'export d'une base de données relationnelle au format XML… Ce sont des documents XML dont le contenu est fortement structuré avec des champs clairement séparés et bien identifiés.
- Les documents XML assimilable à des documents textuels : par exemple, ce mémoire de thèse au format XML, des articles scientifiques, des ouvrages électroniques au format XML (e-books)… Ce sont des documents différenciés des autre de par le fait que leur contenu est principalement composé de texte et non de champs

Donc, à l'instar de [Fuhr & Großjohann, 2001], nous distinguons deux types de documents XML :

- Les *documents XML orientés données* représentent des données très structurées telles que le contenu d'une base relationnelle. Dans ce cadre, la structure arborescente est employée pour décrire le schéma des données (par exemple la description des tables et des attributs). Le contenu des tables est inséré entre les balises. Dans ce type de document, on trouve le contenu brut d'une base de données, des feuilles de calcul, des sorties d'application de type e-commerce…
- Les *documents XML orientés documents* et principalement composés de texte, tels que les versions électroniques des documents papiers qui nous entourent (par exemple, ce mémoire de thèse au format XML). Ces documents ont une structure plus hétérogène et contiennent différents types de données tels que des images, des tableaux… Le balisage sert à représenter la structuration logique des données, à savoir, leur agencement (sections, paragraphes…). Dans ce type de document, on retrouve les

[2] Xyleme XML Server: http://www.xyleme.com/xml_server

articles scientifiques, les articles d'information, les pages Web (essentiellement composée de texte), le contenu textuel d'un mail, les livres numériques (e-books)…

Le traitement associé à ces deux types de documents XML est différent. En effet, un document orienté données étant beaucoup plus structuré qu'un document orienté document, les traitements à appliquer sur un document orienté données sont similaires aux traitements applicables sur des données très structurées telles que des bases de données.

Les travaux qui suivent s'intéressent principalement aux documents XML orientés documents (tout n'excluant pas l'autre type). En effet, comme cela sera précisé dans l'état de l'art, ce type de documents, principalement composé de texte, n'est quasiment pas exploité par les environnements OLAP. De plus, les documents XML orientés documents représentent une grande partie des 80% du volume de données qui n'est pas géré par l'analyse en ligne.

Les documents XML représentant un fort volume au sein d'un système d'information, leur stockage dans un espace centralisé fut envisagé : un entrepôt de documents.

3.3 Les entrepôts de documents

Avec la structuration de documents via le langage XML, les documents ont commencé à être intégrés dans des entrepôts dit de documents et le terme « document warehousing » [Sullivan, 2001] émergea. Les *entrepôts de documents* sont des entrepôts de contenus [Abiteboul, 2006], définis par :

Définition. « Un *entrepôt de contenu* archive des informations qualitatives alors que les entrepôts de données sont plus orientés vers des données quantitatives » [Abiteboul, 2003]. Dans [Dudouet et al., 2005], la définition précise : « Un *entrepôt de contenu* est un entrepôt de données qualitatives sans processus mathématique trivial et inadapté pour un traitement de style OLAP. »

Un entrepôt de contenu doit permettre de trouver et d'extraire l'information, de la stocker et de l'interroger, de l'analyser et de l'enrichir, et d'offrir des outils de visualisation et de « reporting ». Actuellement, il faut noter que l'analyse, la visualisation et le « reporting » ne font pas partie des priorités des entrepôts de contenu qui sont plus orientés vers des problématiques de structuration et de recherche d'informations.

4 Problématique : analyse OLAP de documents

Dans [Sullivan, 2001], l'auteur argumente en faveur de l'utilisation de la fouille de texte sur des documents regroupés au sein d'un entrepôt de documents. Mais bien que la technologie XML soit suffisamment mature pour permettre la conception d'outils d'analyse de texte assez ambitieux [Fankhauser & Klement, 2003], il serait dommage de se passer de la puissance de l'environnement OLAP bien maîtrisée par les décideurs.

En allant au-delà des propos de Peter Fankhauser [Fankhauser & Klement, 2003], nous pensons que la technologie XML permet d'envisager l'intégration de documents dans un environnement OLAP. Ainsi, **notre problématique s'articule autour de l'analyse multidimensionnelle de données documentaires.** Nous nous intéressons plus particulièrement aux données issues de documents XML principalement constitués de données textuelles (des documents XML orientés documents).

Pour permettre l'analyse en ligne de près de 100% des données issues d'un système d'information, les systèmes d'aide à la décision doivent non seulement pouvoir intégrer des données issues de documents orientés documents mais aussi de les analyser. Nous souhaitons adapter un environnement d'analyse, connu des décideurs, aux spécificités des documents.

Enfin, pour permettre l'analyse de données issues de documents XML, il est nécessaire d'intégrer ces documents au sein de l'environnement OLAP.

L'ensemble de la problématique de ce mémoire de thèse est résumé ainsi :

Comment fournir au décideur un environnement permettant l'analyse en ligne de 100% des données des systèmes d'information avec des méthodes et des moyens qu'il maîtrise ?

Comment permettre l'exécution de l'environnement d'analyse OLAP sur des données principalement textuelles, alors que les solutions actuelles reposent principalement sur des données numériques pour produire les résultats d'analyse ?

Comment permettre la représentation conceptuelle d'analyses multidimensionnelles basées sur des données issues de documents en maintenant compatible l'environnement avec les analyses classiques ?

Comment des processus d'intégration des données issues des documents pourront être associés à l'environnement d'analyse multidimensionnel ?

Ce mémoire de thèse vise à répondre à cette problématique.

5 Plan du mémoire

Le mémoire s'articule autour de trois chapitres de propositions. Ces trois chapitres sont précédés d'un état de l'art global, et suivi d'un chapitre décrivant l'implantation d'un prototype permettant de valider les travaux de recherche exposés.

Le **chapitre II** présente l'état de l'art associé à cette thèse, présente le mariage entre le format XML et les différents éléments de l'environnement multidimensionnel. Ce chapitre commence par présenter les travaux concernant les entrepôts de données. Il se poursuit sur les magasins de données en présentant la manipulation multidimensionnelle associée. Le chapitre se termine sur l'analyse multidimensionnelle de données issues de documents ainsi que la synthèse d'information associée. Cette synthèse d'information se fait par le biais de fonctions d'agrégations.

Le **chapitre III** présente un modèle conceptuel multidimensionnel adapté pour l'analyse de documents principalement composés de données textuelles. Ce modèle est basé sur un unique concept de dimension. Le modèle, qui généralise le concept de constellation, est nommé modèle en galaxie. Il permet une représentation des structures multidimensionnelles que le décideur pourra manipuler pour spécifier des analyses.

Le **chapitre IV** présente une fonction d'agrégation adaptée aux données textuelles. Cette fonction permet d'effectuer une opération d'agrégation de mots-clefs en générant des mots-clefs plus généraux. Le chapitre se poursuit sur la présentation des opérations de manipulation adaptées aux structures du modèle conceptuel multidimensionnel en galaxie. A l'instar des

opérations algébriques multidimensionnelles, ces opérations permettent au décideur de spécifier ses analyses.

Le **chapitre V** présente des éléments méthodologiques pour une démarche d'intégration de données issues de documents XML au sein d'un magasin de données. Le chapitre commence par la spécification d'un schéma multidimensionnel conceptuel en fonction de besoins d'analyse et adapté aux sources de données disponibles. Il se termine par l'alimentation des structures multidimensionnelles du magasin par les données des sources.

Le **chapitre VI** valide nos travaux sous la forme de la réalisation d'un prototype. Ce prototype étend et adapte un outil d'analyse multidimensionnel : GraphicOLAPSQL [Tournier, 2004], afin de pouvoir effectuer des analyses de documents XML.

6 Références

[Abelló et al., 2001a] Alberto Abelló, José Samos, Fèlix Saltor, "Understanding Facts in a Multidimensional Object-Oriented Model", *4th ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 32–39, 2001.

[Abelló et al., 2001b] Alberto Abelló, José Samos, Fèlix Saltor, "Understanding Analysis Dimensions in a Multidimensional Object-Oriented Model", *3rd Intl. Workshop on Design and Management of Data Warehouses* (*DMDW*), CEUR Workshop proceedings vol.39, CEUR-WS.org, p. 4.1–4.9, 2001.

[Abiteboul, 2003] Serge Abiteboul, "Managing an XML Warehouse in a P2P Context", *15th Intl. Conf. on Advanced Information Systems Engineering* (*CAiSE*), LNCS 2681, Springer, p. 4–13, 2003.

[Abiteboul, 2006] Serge Abiteboul, "Entrepôts de contenu autour de XML et des services Web", *2ème journées francophones sur les Entrepôts de Données et Analyse en ligne* (*EDA*), Revue des Nouvelles Technologies de l'Information (RNTI), numéro spécial, vol.RNTI-B-2, Cépaduès Editions, conference invite, p. 1, 2006.

[Chaudhuri & Dayal, 1997] Surajit Chaudhuri, Umeshwar Dayal, "An Overview of Data Warehousing and OLAP Technology", *ACM SIGMOD Record*, vol.26(1), ACM Press, p. 65–74, mars 1997.

[Codd, 1993] E.F. Codd, S.B. Codd, C.T. Salley, *Providing OLAP (On Line Analytical Processing) to user analyst: an IT mandate*, rapport technique, E.F. Codd and associates, (white paper de Hyperion Solutions Corporation), 1993.

[Colliat, 1996] George Colliat, "OLAP, relational, and multidimensional database systems", *ACM SIGMOD Record*, vol.25(3), ACM Press, p. 64–69, septembre 1996.

[Dudouet et al., 2005] François-Xavier Dudouet, Ioana Manolescu, Benjamin Nguyen, Pierre Senellart, "XML Warehousing Meets Sociology", *IADIS Intl. Conf. WWW/Internet* (*ICWI*), IADIS Press, Lisbonne, Portugal, Octobre 2005.

[Fankhauser & Klement, 2003] Peter Fankhauser, Thomas Klement, "XML for Data Warehousing Chances and Challenges" (Extended Abstract), *5th Intl. Conf. on Data Warehousing and Knowledge Discovery* (*DaWaK*), LNCS 2737, Springer, p.1-3, 2003.

[Fuhr & Großjohann, 2001] Norbert Fhur, Kai Großjohann, "XIRQL: A Query Language for Information Retrieval in XML Documents", *24th ACM Conf. on Research and development in information retrieval* (*SIGIR*), ACM Press, p.172–180, 2001.

[Inmon, 1996] W. H. Immon, *Building the Data Warehouse*, John Wiley and sons, New York, NY, ISBN : 0764599445, 1996 (2ème ed.), 4ème ed. 2005.

[Kimball, 1996] Ralph Kimball, *The data warehouse toolkit: Practical Techniques for Building Dimensional Data Warehouses*, John Wiley and Sons, ISBN : 0-471-15337-0, 1996, 2ème ed. : Ralph Kimball, Margaery Ross, *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling*, *2nd Edition*, John Wiley & Sons, 2002.

[Ravat et al., 2007e] Franck Ravat, Olivier Teste, Ronan Tournier, Gilles Zurfluh, "Algebraic and graphic languages for OLAP manipulations", *Intl. Journal of Data Warehousing and Mining* (*ijDWM*), Idea Publishing Group (IGP), 2007 (à paraître).

[Sullivan, 2001] Dan Sullivan, *Document Warehousing and Text Mining*, Wiley John & Sons, ISBN: 0471399590, 2001.

[Teste, 2000] Olivier Teste, *Modélisation et Manipulation d'Entrepôts de Données Complexes et Historisées*, thèse de doctorat, Université Paul Sabatier, Toulouse 3 (France), décembre 2000.

[Torlone, 2003] Riccardo Torlone, "Conceptual Multidimensional Models", Chapitre III de *Multidimensional Databases: Problems and Solutions*, Maurizio Rafanelli (Ed.), Idea Publishing Group (IGP), ISBN 1-59140-053-8, p. 69–90, 2003.

[Tournier, 2004] Ronan Tournier, *Bases de données multidimensionnelles : étude et implantation d'un langage graphique*, rapport de master de recherche, IRIT, Université Paul Sabatier, Toulouse 3 (France), juin 2004.

[Tseng & Chou, 2006] Frank S.C. Tseng, Annie Y.H. Chou, "The concept of document warehousing for multi-dimensional modeling of textual-based business intelligence", *journal of Decision Support Systems* (*DSS*), vol.42(2), Elsevier, p. 727–744, novembre 2006.

[W3C-XML, 2006] *Extensible Markup Language (XML) 1.0 (Fourth Edition)*, recommandation du W3C (29/09/2006), http://www.w3.org/TR/xml/

[W3C-XSL, 1999] *XSL Transformations (XSLT) version 1.0*, recommandation du W3C (19/09/1999) http://www.w3.org/TR/xslt

[W3C-XSchema, 2006] *XML Schema 1.1 Part 1 : Structures*, document de travail du W3C (31/08/2006) http://www.w3.org/TR/xmlschema11-1/ et *XML Schema 1.1 Part 2: Datatypes*, document de travail du W3C (17/02/2006) http://www.w3.org/TR/xmlschema11-2/

[W3C-XQuery, 2007] *XQuery 1.0 and XPath 2.0 Formal Semantics*, recommandation du W3C (23/01/2007) http://www.w3.org/TR/xquery-semantics/

CHAPITRE II
État de l'art

Résumé du chapitre

Ce chapitre présente l'état de l'art du mariage entre les systèmes d'aide à la prise de décision et le format XML. Le chapitre commence par présenter l'architecture des systèmes d'aide à la décision. Les sections du chapitre suivent les différents niveaux de cette architecture (entrepôts, magasins, manipulation et analyse/restitution). Premièrement, les entrepôts XML sont présentés. Les magasins de données sont abordés ensuite, en commençant pas un tour d'horizon de la modélisation conceptuelle multidimensionnelle. La section se poursuit sur l'ajout du format XML dans l'environnement des magasins. Une analyse des différents langages de manipulation multidimensionnelle avec une comparaison de l'ensemble des différents opérateurs proposés est exposée avant de présenter les opérateurs qui ont été spécifiés pour la gestion du format XML dans la manipulation. Enfin la restitution et l'analyse sont abordées. Premièrement, les précédentes propositions permettant l'analyse de documents sont détaillées, puis les travaux de synthétisation d'information XML nécessaire à l'analyse de données issues de documents sont présentés. Le chapitre se conclut par un résumé de la problématique et un positionnement des propositions qui vont suivre.

Sommaire

CHAPITRE II : État de l'art

« Le savoir que l'on ne complète pas chaque jour diminue… »

— Proverbe chinois.

Ce chapitre présente l'état de l'art des travaux associés à ce mémoire de thèse et présente le mariage entre le langage XML et les systèmes d'aide à la décision. Une présentation détaillée est faite des travaux de recherche qui présentent l'analyse de données issues de documents XML.

1 Systèmes d'aide à la décision et XML

Dans le chapitre précédent, le format d'échange XML a été présenté. Ce langage semi-structuré est devenu de facto un standard pour l'échange de données entre applications [Tourwé et al., 2003]. Grâce au développement, entre autres, des services Web, les données au format XML se sont multipliées et peu à peu, le besoin d'analyse de ces données s'est fait ressentir. Par conséquent, les systèmes d'aide à la décision ont cherché à exploiter ces données, faisant apparaître leur inadaptation à ce nouveau format.

Cette section présente l'architecture des systèmes d'aide à la prise de décision et le positionnement général du format XML vis-à-vis de cette architecture.

1.1 Architecture décisionnelle

La Figure 2 rappelle notre architecture d'un système décisionnel qui se décompose en quatre niveaux : les sources de données, l'entrepôt de données, le magasin de données et l'interface de restitution et d'analyse. Une section de l'état de l'art sera dédiée à chaque niveau. Dans certaines propositions l'emploi d'un entrepôt de données est parfois optionnel, ainsi dans certaines architectures, les sources sont intégrées directement dans des magasins de données.

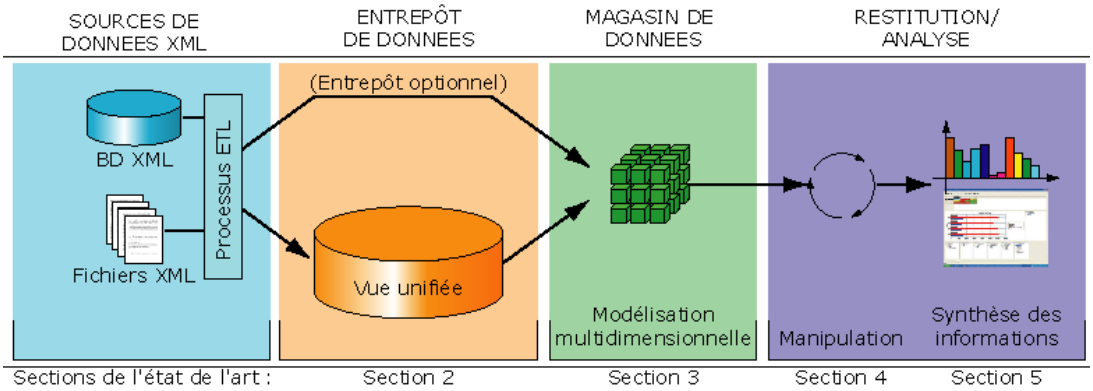


Figure 7 - Architecture d'un système d'aide à la prise de décision.

Parmi les différentes sources envisageables, il est possible de rencontrer des bases données XML natives, des fichiers au format XML ou encore des combinaisons des deux (cas des sites Web).

Originellement le format XML n'était présent que dans les sources, mais peu à peu, il s'est directement invité au cœur des systèmes décisionnels.

1.2 Positionnement : XML et les systèmes décisionnels

Depuis peu la technologie XML s'est insérée dans les architectures décisionnelles (cf. Tableau 2 et Figure 8). Ainsi il est possible de retrouver la technologie XML dans :

- dans les sources de données,
- dans les espaces de stockage (entrepôt ou magasin),
- au niveau de l'analyse et de la restitution (OLAP XML).

Tableau 2- Intrusion du XML à différents niveaux.

Sources	Entrepôt	Magasin	Analyse	Description
XML	-	-	-	ETL traditionnel
XML	XML	-	-	Entrepôt de données ou de contenu
XML	XML[1]	XML	-	Cubes XML
XML	XML[1]	XML[1]	XML	Analyse en ligne XML (OLAP XML)

[1]: composant optionnel

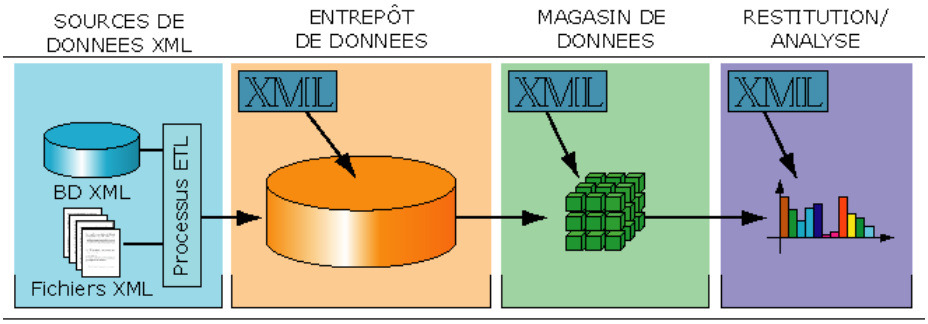


Figure 8 – Intrusion du format XML dans les différents niveaux de l'architecture.

Il faut noter aussi l'emploi de XML autour du niveau du magasin en tant que moyen de communication entre différents systèmes décisionnels hétérogènes : [Hümmer et al., 2003], [Nguyen et al., 2001], [Nguyen et al., 2003] et une proposition d'un consortium d'entreprises, XML for Analysis[3] (cf. Figure 9).

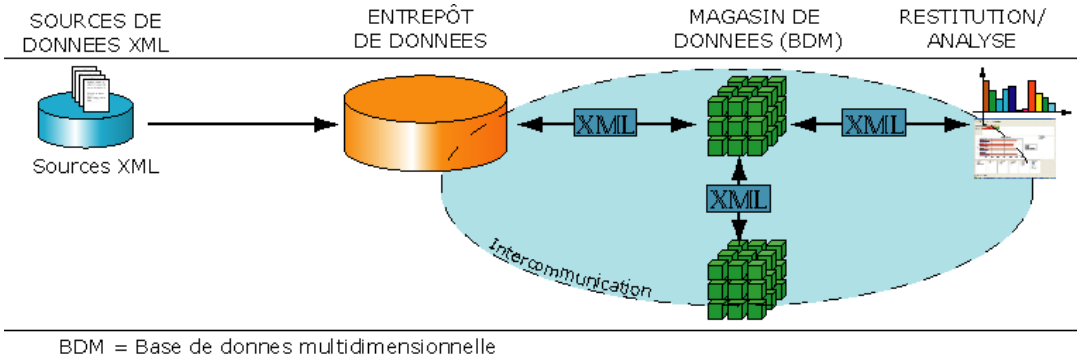
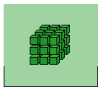


Figure 9 – XML en tant que langage d'interopérabilité entre systèmes hétérogènes.

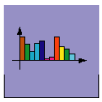
[3] XMLA : XML for Analysis de http://www.xmlforanalysis.com/

1.3 Plan

Les deux premières sections de l'état de l'art présentent les travaux relatifs aux niveaux de stockage de l'architecture décisionnelle.

	La section suivante (section 2) présente le deuxième niveau de l'architecture : l'entrepôt de données XML. C'est-à-dire les propositions concernant l'introduction du XML dans l'environnement de stockage centralisé du système décisionnel, l'entrepôt.
	La section 3 poursuit sur le troisième niveau : les magasins de données. La section commence par présenter les solutions de modélisation multidimensionnelle. Puis, expose les apports du format XML dans les magasins. Notamment en terme d'intégration de données au sein de magasins de données non XML et au sein de magasins de données XML natifs.

Les deux sections qui suivent exposent les travaux relatifs au niveau d'analyse et de restitution de l'architecture décisionnelle.

	La section 4, commence par un état de l'art de la manipulation: il s'agit d'opérateurs permettant la manipulation des structures multidimensionnelles afin de spécifier des analyses. La section se poursuit sur les méthodes employées pour synthétiser les informations lors des analyses. Premièrement, les fonctions d'agrégation sont présentées suivies des modifications apportées pour permettre la gestion de données textuelles et XML.
	La section 5 présente les rares travaux d'analyse de données issues de documents dans un environnement OLAP. Plusieurs approches sont envisageables, mais, seules quelques unes ont été étudiées.

Enfin, la section 6 conclut par un bilan sur les apports à l'environnement des systèmes d'aide à la décision avec l'inclusion du XML. Cette section se termine sur un résumé des problèmes non résolus vis-à-vis de notre problématique et expose les objectifs du mémoire.

2 Les entrepôts

Comme dit en introduction, les entrepôts [Kimball, 1996] permettent de présenter une vision uniformisée des données qui serviront à alimenter les magasins qui sont plus spécifiques et adaptés selon des perspectives d'analyses bien précises. Un état de l'art peut être trouvé dans [Ravat, 2007].

Cette section se focalise sur les « XML warehouses » ou entrepôts XML. Il s'agit d'entrepôts de données dont le stockage est au format XML. Les données ne sont pas nécessairement traitées de manière similaire à un entrepôt de données classique. Nous distinguons :

- l'entrepôt de données XML (cf. section 2.1),
- l'entrepôt de documents XML (cf. section 2.2).

Ces deux types d'entrepôts dépendent du type de données qui y seront stockées. L'entrepôt de données XML est un système de stockage de documents XML très structurés et qui ne diffère que peu d'un entrepôt de données classique hormis son format : XML. Le stockage de

documents XML orientés documents dans un entrepôt est aussi appelé entrepôt de documents ou encore entrepôt de contenu [Abiteboul, 2003]. Les entrepôts de contenu sont destinés à un stockage de données [Abiteboul, 2006] et non destinés à l'OLAP.

Ces deux systèmes répondent à des problématiques et des buts différents bien qu'ils emploient une terminologie similaire, voir identique.

2.1 Entrepôt de données XML

But. Le but de ce type d'entrepôts de données est de fournir une vision uniforme des sources de données XML, favorisant ainsi leur intégration au sein des systèmes d'analyse.

Caractéristiques. Au sein de ce type d'entrepôts, les données XML sont des données principalement issues de documents orientés données. La structure de l'entrepôt est stockée via la structure du format XML. Grâce à ce format et aux outils de transformation employant XSLT ou encore des requêtes XQuery, il est possible de manière simple d'uniformiser la collection de données XML et ainsi de faciliter leur intégration dans le système de stockage.

L'environnement est identique à un entrepôt de données classique, hormis le format XML omniprésent, et possède les mêmes caractéristiques :

- données non-volatiles,
- données intégrées,
- données archivées et variant selon le temps,

Cet environnement sert d'intermédiaire pour obtenir des données orientées analyse. Ainsi, dans [Nassis et al., 2004], les données XML sont stockées sous la forme de documents XML structurés selon une grammaire XML au niveau logique et représentés par un diagramme de classes UML au niveau conceptuel. Un tel entrepôt sert de base pour permettre, par la suite, l'alimentation de structures orientées analyse telles que celles d'un magasin de données.

Avantages/Inconvénients. Le principal intérêt de cet environnement est d'éviter de coûteuses conversions entre des sources XML et un format spécifique d'entrepôt. Il s'agit d'une solution simple et disponible, notamment avec l'ajout récent de la gestion du format XML dans des SGBD tels qu'Oracle[4] ou encore MySQL[5]. Néanmoins, ce gain est relatif car les bases de données XML sont moins matures que des formats plus anciens tels que le relationnel. De plus, si l'environnement décisionnel ne dispose pas d'un magasin avec une gestion XML intégrée, des conversions sont nécessaires lors de l'alimentation des structures du magasin.

Le stockage dans un entrepôt de documents XML orientés document, à savoir des documents XML essentiellement constitués de données textuelles peu structurées à première vue, a donné naissance à un nouveau type d'entrepôt : l'entrepôt de documents.

2.2 Entrepôt de documents XML

But. Le but de ce type d'entrepôt est de fournir un environnement de stockage de données peu structurées. Les principales préoccupations d'un entrepôt de document sont 1) le stockage avec une uniformisation des données et 2) la restitution de fragments jugés pertinents par l'utilisateur. En d'autres termes, il s'agit d'une problématique de recherche d'information.

[4] Oracle database server (11*g*) : http://www.oracle.com/database/index.html
[5] MySQL server (5.0) : http://dev.mysql.com/downloads/mysql/5.0.html

L'analyse multidimensionnelle est une problématique secondaire car un tel entrepôt est inadapté pour des traitements de style OLAP [Dudouet et al., 2005].

Dans cette catégorie on trouve les entrepôts de données Web. Ces systèmes ont muté vers une gestion XML de leur contenu, facilitant leur exploitation et leur utilisation. En effet le format HTML, généralement employé pour les données Web, ne fourni pas une grammaire assez expressive pour permettre une description détaillée de la structure de ces données.

Caractéristiques. Le stockage des données est au format XML et un document équivaut à un fragment de données. Les documents de l'entrepôt sont principalement des documents composés de fragments de textes peu structurés (en comparaison avec des données relationnelles par exemple). Un moteur de recherche est souvent associé à ce type d'entrepôt. Il existe peu, voire pas, d'interfaçage avec un environnement d'analyse en ligne (OLAP) permettant l'analyse multidimensionnelle du contenu de l'entrepôt.

Parmi les quelques systèmes existants, deux exemples d'application se détachent dans le cadre d'entrepôts de documents de part leur ambition et les travaux de recherche associés :

- Xyleme[6] [Abiteboul et al., 2002] : dont le système est désormais commercialisé et le serveur sert à de nombreuses compagnies pour entreposer leur documentation.
- Whoweda[7] : dont le projet est d'entreposer une grande partie du contenu du web. Toutefois, à ce jour, le projet n'est plus maintenu.

Avantages/Inconvénients. Ce type d'entrepôts permet une uniformisation en terme de structure des documents XML orientés documents. Ces entrepôts répondent à des priorités de restitution des données ou de recherche d'information. Mais la restitution n'est pas orientée décideur car elle ne prend pas en compte la synthèse des informations employée par l'analyse en ligne OLAP. L'un des pères des entrepôts de documents, Dan Sullivan, suggère plutôt l'emploi de la fouille de texte [Sullivan, 2001] sur les entrepôts de documents.

2.3 Bilan : l'analyse de documents n'est pas favorisée

En résumé, dans le cadre des documents orientés données, les entrepôts de données XML permettent une structuration homogène de leur contenu tout en évitant de coûteux processus de transformation qui seraient nécessaires pour la conversion des données XML dans un format natif d'un entrepôt de données classique. Toutefois, il faut noter le manque de maturité de ce type d'entrepôt.

[6] Xyleme XML Server : http://www.xyleme.com/xml_server
[7] Whoweda : warehousing of web data de http://www.cais.ntu.edu.sg:8000/~whoweda/ ; références des publications associées disponibles sur http://mandolin.cais.ntu.edu.sg/~whoweda/publications.htm

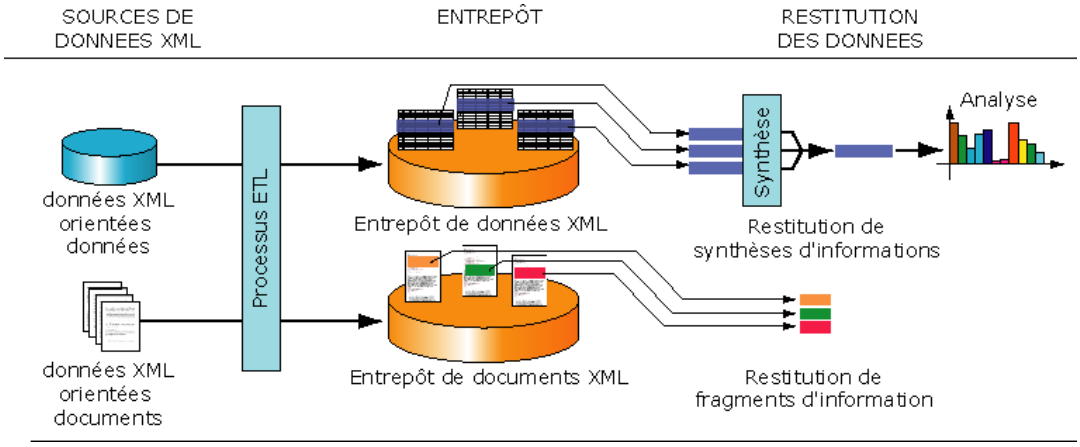
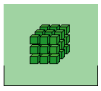


Figure 10 – Comparaison de principes de restitution des entrepôts de données et de documents.

Lors de l'emploi d'un entrepôt de données XML issues de documents, telles que des données textuelles, l'entrepôt de documents ne permet pas une mise en valeur des données pour un environnement OLAP.

3 Magasins de données



Les magasins de données correspondent à la structuration multidimensionnelle des données en vue d'analyses. Cette section est subdivisée en quatre parties :

- la modélisation conceptuelle multidimensionnelle,
- l'intégration de données XML dans le magasin,
- l'intégration de données avec un stockage multidimensionnel XML,
- l'intégration des données virtuelle.

La première section aborde la définition des structures multidimensionnelles qui composent les magasins de données classiques. Dans la seconde section, un magasin classique, structuré de manière multidimensionnelle (avec un schéma en étoile par exemple), est employé. Un processus d'alimentation permet l'intégration des données issues de sources XML qui sont alors restructurées pour pouvoir être insérées dans les structures multidimensionnelles du magasin. La troisième section présente l'intégration directe dans un magasin également au format XML, ce qui évite un processus de conversion des données une fois restructurées. La dernière section expose l'intégration virtuelle avec par exemple un mécanisme de vue sur des sources ou un entrepôt XML. Ce type de magasin n'est pas matérialisé.

3.1 Modélisation multidimensionnelle

La modélisation multidimensionnelle conceptuelle au sein d'un magasin de données permet une représentation des concepts indépendamment de toute contrainte d'implantation logique ou physique. Toutefois, les données issues de documents, principalement composés de texte, ont quelques spécificités que le modèle devrait pouvoir représenter :

- données structurées de manières hiérarchiques,
- données essentiellement textuelles,
- liaisons intra ou inter documents (références, citations, liens hypertextes…).

L'idéal serait de disposer d'un modèle multidimensionnel qui permette de modéliser l'ensemble de ces caractéristiques tout en préservant les bénéfices déjà acquis par dix années de recherche en modélisation de bases de données multidimensionnelles. Toutefois, il n'existe toujours pas de standard [Ravat, 2007]. Les concepts et les structures existent sans base théorique stable et standardisée [Niemi Ti. et al., 2003] et [Rizzi et al., 2006].

Initialement, les premiers modèles proposés reposent sur la métaphore de *cube* ou d'*hypercube* (cf. chapitre 1). Le sujet d'analyse est modélisé en tant que valeurs contenues dans une cellule d'un cube et chaque « arête » du cube représente un axe d'analyse. Un état de l'art détaillé peut être trouvé dans [Ravat et al., 2007e]. Mais ces modèles ont une faiblesse dans la modélisation du fait (le sujet d'analyse), pas ou peu de modélisation conceptuelle des dimensions, pas de représentation explicite des structures hiérarchiques des dimensions et pas de séparation entre structure et contenu.

Pour répondre à ces problèmes, des modèles multidimensionnels sémantiquement plus riches furent conçus. Ces modèles conceptuels peuvent être regroupés en trois catégories en fonction du paradigme employé pour représenter les concepts :

- modèles basés sur le paradigme entité/association : [Tryfona et al., 1999], [Sapia et al., 1998] et [Hahn et al., 2000] ;
- modèles basés sur le paradigme objet : [Luján-Mora, 2005] et [Luján-Mora et al., 2006], [Pedersen T.B., 2000] et [Pedersen T.B. et al., 2001], [Abelló, 2002] et [Abelló et al., 2006], [Nguyen et al., 2000] et [Bruckner et al., 2001] ;
- modèles multidimensionnels : [Golfarelli et al., 1998], [Cabibbo & Torlone, 1998] et [Cabibbo & Torlone, 2000], [Tournier, 2004], [Ravat et al., 2007e], [Schneider, 2003] et [Schneider, 2007].

Les modèles reposant sur le paradigme entité/association emploient le même formalisme et reposent sur des relations d'associations entre sujets (faits) et axes d'analyse (dimensions). Les modèles basés sur le paradigme objet reposent sur les notations du diagramme de classes d'UML[8]. Enfin, les modèles multidimensionnels reposent sur une notation permettant la capture de la structure multidimensionnelle pour l'utilisateur, indépendamment de notations préexistantes. Un état de l'art détaillé de ces propositions est présenté dans [Ghozzi, 2004]. Bien que l'ensemble de ces travaux modélisent des sujets d'analyses (faits) et des axes d'analyse (dimensions), ces modèles représentent la structure des dimensions de manière très différentes [Torlone, 2003].

Il faut noter que ces modèles ne peuvent pas gérer la composition essentiellement textuelle des documents et encore moins la représentation des liaisons intra ou inter documents. Ces travaux de modélisation ne prennent pas en compte la gestion de documents au format XML. Toutefois, certains travaux ont proposé une approche visant à intégrer des données au format XML dans un magasin modélisé avec une modélisation multidimensionnelle.

3.2 Intégration de données XML

But. Le but de ce type de magasin standard est d'intégrer à partir d'une définition commune au format DTD (ou XSchema), les données issues des sources XML. Il s'agit de documents XML orientés données.

[8] UML : Unified Modelling Language, Langage de spécification objet de l'OMG (Object Management Group), http://www.uml.org/

Principe. Les données XML sont transformées selon une structuration multidimensionnelle en générant par exemple des fichiers intermédiaires. Ainsi les données XML sont reformatées selon un schéma multidimensionnel (étoile, flocon…). Par la suite, ces données intermédiaires sont converties au format natif du magasin de données pour en alimenter les structures.

C'est dans cette catégorie que se retrouve l'essentiel des travaux de recherches du domaine. Denis Pedersen regroupe ces propositions en deux catégories [Pedersen D. et al., 2002] :

- intégration physique,
- intégration logique, avec : l'approche fédérative et l'approche de contextualisation.

Dans le cadre de l'intégration physique, les données sources XML sont intégrées dans un magasin de données et sont stockées selon le format natif du magasin. L'intégration logique survient lorsque l'insertion de données XML au sein d'un magasin de données est impossible. Dans ce contexte, la solution consiste à conserver les données XML d'un côté et le magasin de l'autre et un système médiateur permet la fusion logique des deux systèmes.

3.2.1 Intégration physique

But. Le but de l'intégration physique est d'alimenter un magasin de données classique avec des sources XML en employant des processus ETL.

Caractéristiques. Les sources sont des documents XML orientés données (très structurés) décrits par une ou plusieurs DTD. Les données XML sont transformées pour être intégrées dans une structure multidimensionnelle au format XML qui est décrite par une DTD. Un processus de transformation permet la conversion, par exemple par l'intermédiaire de XSLT ou XQuery et des fichiers XML intermédiaires. Le résultat des différentes méthodes est similaire : des données XML prêtes à être chargées dans un magasin de données classique moyennant un processus d'adaptation spécifique au format du magasin.

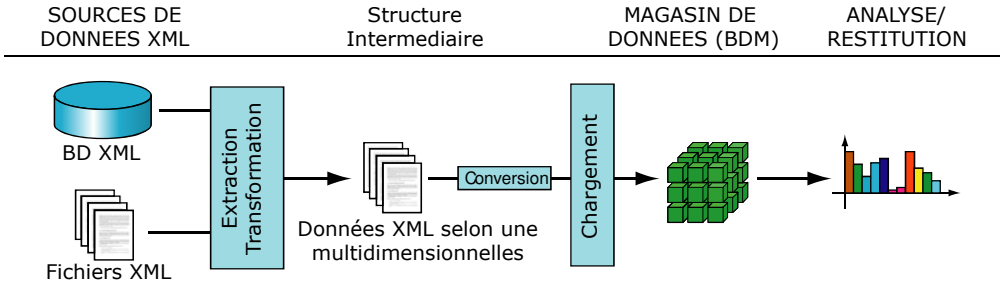


Figure 11 - Intégration physique de sources XML (la structure intermédiaire est optionnelle).

Dans [Golfarelli et al., 2001], la structure des sources est décrite par une DTD. Un processus de transformation extrait les données des sources puis les intègrent dans une structure multidimensionnelle XML décrite elle aussi par une DTD. De manière similaire, [Pokorný, 2001] propose l'extraction de données XML à partir de plusieurs DTD complémentaires. L'auteur propose un schéma en flocon dont chaque niveau est décrit par au moins une DTD. [Vrdoljak et al., 2003], propose d'opérer à partir du formalisme XSchema[9] (à la place de la DTD) avec un outil d'intégration [Vrdoljak et al., 2006]

Dans [Niemi Ta. et al., 2002], les auteurs proposent d'assembler des cubes de données XML issus d'un environnement réparti. Ces cubes sont construits à la demande de l'utilisateur à

[9] XSchema de http://www.w3.org/XML/Schema

partir de sources disposées sur une grille. Dans le cadre d'un système décisionnel déployé sur Internet, dans [Huang & Su, 2002], les auteurs encapsulent une extraction d'un magasin de données classique dans une structure multidimensionnelle au format XML. S'attaquant aux problèmes de volumes et de temps consommés lors de la construction de magasins de données, les auteurs de [Zhang et al., 2003] assemblent des magasins à partir de sources XML distribuées. Le système sélectionne les données selon les accès utilisateurs au sein de chaque source distribuée. Ceci permet l'élimination du magasin des données peu usitées.

Employant le langage de requête XQuery, dans [Rusu et al., 2004], les auteurs proposent un canevas de requêtes types pour sélectionner des données dans les sources et générer en sortie des fichiers XML structurés de manière multidimensionnelle. La méthode produit un document XML par dimension et un autre par fait. Des index permettent de lier les données dimensionnelles et factuelles.

Avantages/Inconvénients. L'intégration physique permet d'employer les langages de transformation et de restructuration associés au XML pour assembler de manière multidimensionnelle les données et bénéficier ainsi de processus cohérents. Néanmoins cette approche repose sur des sources fortement structurées (documents orientés données). En outre, à ce jour, l'intégration de données issues de documents XML orientés documents n'a pas été prise en compte.

3.2.2 Intégration logique fédérative

But. L'*approche fédérative* est employée lorsque une partie des données sources ne peut pas être intégrée au sein du magasin pour des raisons techniques. Par exemple, dans le cas de données fortement dynamiques (cotations en bourse…) qui ne permettent pas un processus d'alimentation et de rafraîchissement du magasin sans une perte majeure de performance.

Caractéristiques. L'approche fédérative [Jensen et al., 2001] est constituée de trois éléments : un magasin de données classique, un ensemble de données XML et des liens entre le magasin et les données XML [Yin & Pedersen T.B., 2004]. Les données XML peuvent être dimensionnelles [Pedersen D. et al., 2002] ou factuelles [Pedersen D. et al., 2004]. Dans cette approche, les données restent séparées et ne sont fusionnées qu'à la phase de restitution/analyse. Un opérateur spécifique, « décoration » [Yin & Pedersen T.B., 2004], permet d'ajouter des données XML complémentaires à une dimension du magasin lors d'une analyse. Le schéma du magasin est alors « décoré » par les nouvelles données XML.

Par exemple, il est possible d'ajouter à un attribut d'une dimension tel que *Ville*, des données complémentaires issues d'Internet (population, superficie, pluviométrie…).

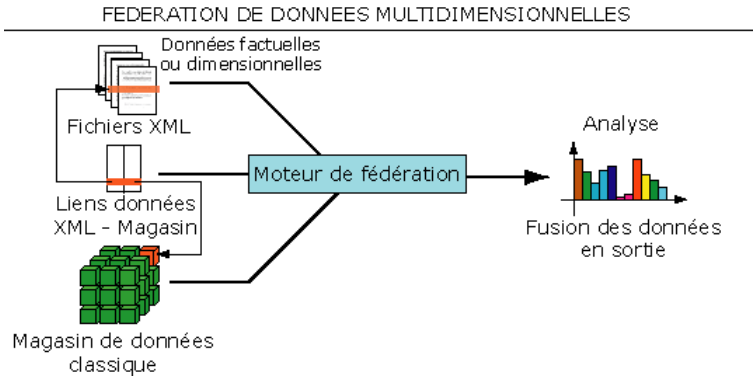


Figure 12 - Féderation de données XML avec un magasin.

Avantages/Inconvénients. La fédération permet de pallier le fait que la technologie optimisée des magasins de données peut souffrir de processus de rafraîchissement coûteux. Toutefois, le temps de réponse du système est ralentit par la fusion des informations en provenance des deux sources ainsi que l'écart de performance entre le magasin traditionnel et le système XML (moins mature). Dans le cadre de cette approche, seul des documents fortement structurés sont traités (documents orientés données).

3.2.3 Intégration logique de contextualisation

De son côté, l'approche de contextualisation emploie une intégration logique nécessaire des suites de la forte hétérogénéité entre les données du magasin (fortement structurées et orientées sujet) et des données XML complémentaires (faiblement structurées).

But. Le but de l'*approche de contextualisation* est de fournir un complément d'information pour une analyse en cours. Il s'agit d'extraire le contexte de l'analyse et d'effectuer de la recherche d'informations relative à ce contexte au sein d'une base de documents XML.

Caractéristiques. L'approche de contextualisation consiste à considérer des documents XML orientés documents comme une source complémentaire d'informations relative à un contexte d'analyse en cours [Pérez et al., 2005] et [Pérez et al., 2007]. Le système est composé de deux éléments : un magasin de données classique et un entrepôt de documents. Un moteur de recherche d'information restitue à l'utilisateur les documents jugés pertinents vis-à-vis du contexte de l'analyse.

Par exemple, lors d'une analyse de quantité de ventes d'un certain type de photocopieur dans le sud de la France durant l'année 2005, le système y associera les documents (articles de presse, rapports…) concernant le photocopieur, la région, et l'année considérée dans l'analyse.

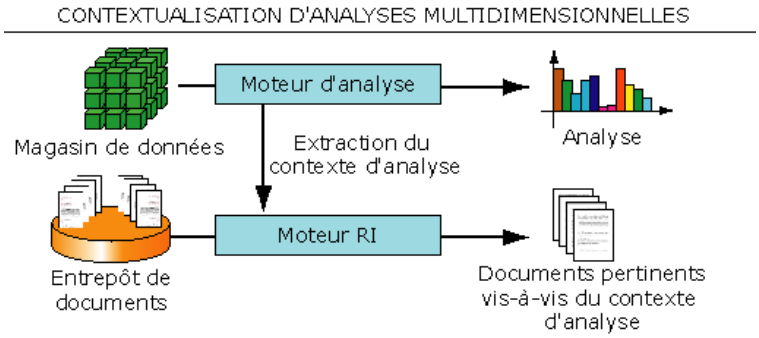


Figure 13 - Contextualisation d'un système d'analyse multidimensionnelles avec des documents XML.

Avantages/Inconvénients. Ce système permet d'aider le décideur dans sa recherche d'informations concernant le contexte d'une analyse. Néanmoins, les documents retournés doivent tous être traités manuellement. L'entrepôt de document n'intègre aucune fonctionnalité d'analyse et le décideur doit sélectionner puis lire les documents et risque fort d'être surchargé d'informations.

3.2.4 Bilan : l'intégration ne considère pas les documents textuels

Chaque alternative pour l'intégration multidimensionnelle de données XML permet de pallier à un problème précis. L'intégration physique est abordée comme un processus ETL supplémentaire pour permettre la gestion du nouveau type de données. L'intégration logique a pour but de ne pas directement intégrer les données XML soit pour des raisons de performance, soit pour des raisons d'hétérogénéité. La plupart des travaux d'intégration logique ou physique se basent sur des données fortement structurées, à savoir des documents XML orientés données.

En conclusion, dans le cadre de l'intégration de données dans un environnement multidimensionnel, XML reste un format adapté à l'intégration de données. Aucune proposition ne permet l'analyse de données issues de documents XML essentiellement composés de texte et l'approche de contextualisation se contente de restituer les données documentaires sans en permettre l'analyse.

Les technologies XML devenant peu à peu plus matures et les performances s'améliorant, la solution suivante fut de spécifier des magasins XML natifs.

3.3 Stockage XML multidimensionnel

But. Le but du stockage en XML d'un magasin de données est de disposer d'un environnement uniformisé entre les sources, l'éventuel entrepôt et le magasin de données. Ainsi le processus d'alimentation du magasin est simplifié.

Caractéristiques du magasin. Différents schémas multidimensionnels sont employés : étoile, flocon [Kimball, 1996] ou xFACT [Nassis et al., 2004]. Le schéma est défini au moyen d'une DTD (ou XSchema). La structuration arborescente des données XML permet une représentation aisée des dimensions où les données sont disposées de manière hiérarchique.

Caractéristiques du niveau physique. Le schéma multidimensionnel est stocké en XML avec une alternative au niveau de la gestion des instances multidimensionnelles :

- *instances stockées séparément* : dans ce cas, les instances factuelles et dimensionnelles sont stockées de manière séparée.
- *instances stockées ensembles* : chaque instance d'un fait est stocké avec les instances dimensionnelles auxquelles il est associé.

Le stockage physique des instances de manière séparée repose sur un fichier par fait contenant l'ensemble des valeurs du fait et un fichier par dimension contenant l'ensemble des valeurs des dimensions. Une structure intermédiaire d'index lie les instances du fait à celles des dimensions. Dans [Park et al., 2005], les auteurs utilisent la structure d'entrepôt du xFACT [Nassis et al., 2004] (cf. la sous-section 3.4). La structure hiérarchique des documents XML est employée pour structurer les dimensions. Il s'agit de la seule proposition qui considère des documents XML orientés documents. Néanmoins, aucun modèle conceptuel formel n'est défini et le modèle logique proposé ne capture pas les spécificités des documents telles que leur structure. De plus, bien que ce modèle permette l'analyse de données textuelles, il ne propose aucune modélisation de la structure principale de l'analyse : la mesure.

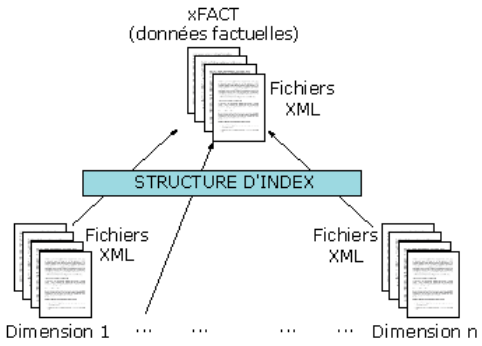


Figure 14 – Magasin de données répartis entre plusieurs fichiers XML liés par un index [Park et al., 2005].

Lorsque les instances sont stockées ensemble, le magasin correspond à un fichier XML par fait contenant l'ensemble des valeurs du fait et des dimensions. Chaque valeur du fait est associée aux valeurs correspondantes de chaque dimension, entraînant une grande redondance dans le stockage des instances des dimensions. Dans le cadre de l'analyse de données complexes avec des méthodes de fouille de données, une modélisation en flocon de données multidimensionnelles XML est proposée dans [Boussaid et al., 2006] et [Messaoud, 2006]. Les propositions permettent l'analyse de données complexes, mais ne sont pas adaptées pour l'analyse de données textuelles issues de documents XML.

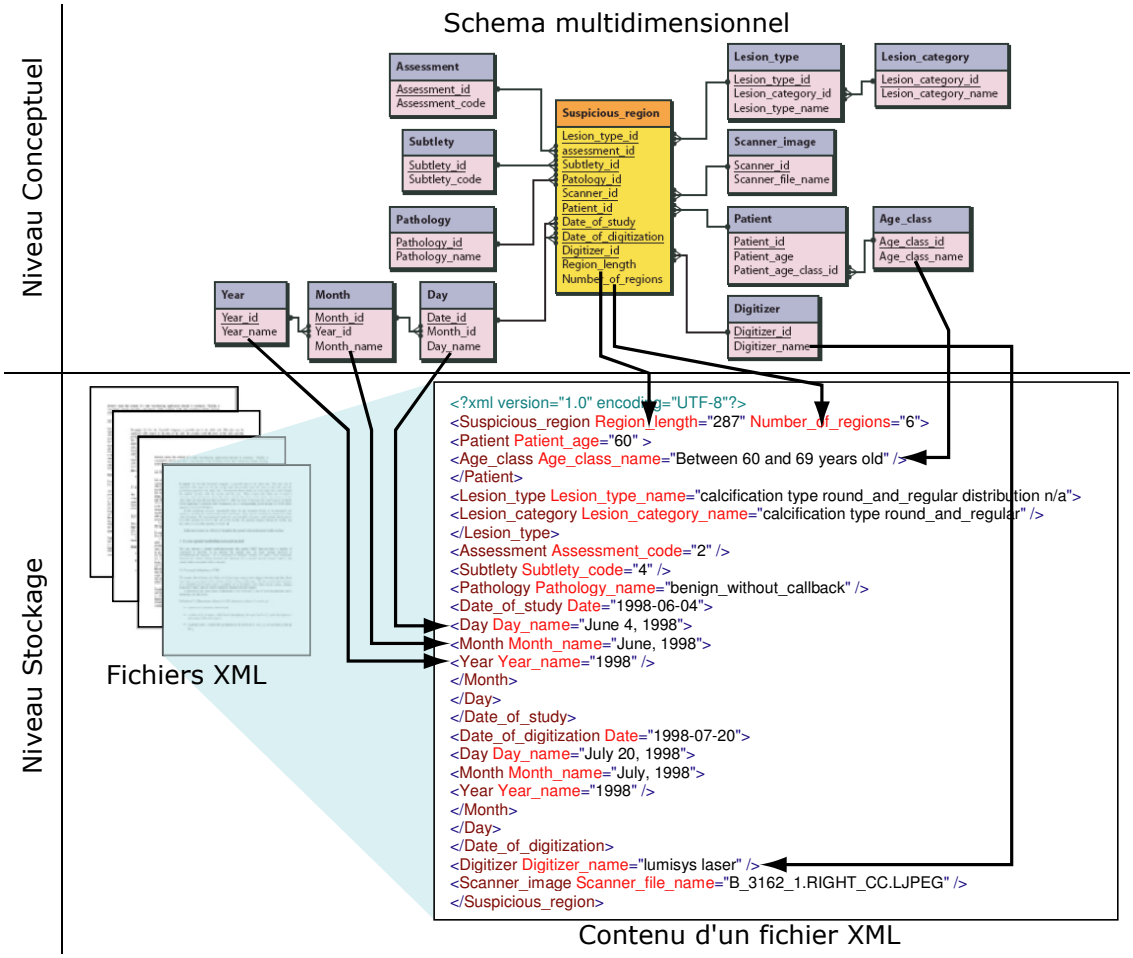


Figure 15 - Schéma en flocons reposant sur des fichiers XML [Messaoud, 2006].

Avantages/Inconvénients. Un magasin de données au format XML permet de fournir plus de facilités quant à l'intégration de sources de données XML. Grâce à la flexibilité du format XML, un tel magasin est capable de gérer des données complexes [Boussaid et al., 2006]. La structure arborescente du XML permet, au sein des dimensions, une modélisation des données selon des hiérarchies strictes [Malinowski & Zimányi, 2006].

Le format physique de stockage du XML est volumineux (UTF-8), surtout dans le cadre d'une approche avec stockage des instances ensemble qui introduit une grande redondance. La solution alternative de stocker les instances séparément est très similaire à une architecture multidimensionnelle relationnelle (ROLAP).

En conséquence, les magasins XML sont loin d'être matures et devront faire face à des problèmes de performances, de matérialisations et de volume. Toutefois il faut noter qu'associé à un environnement de restitution adapté, ce type de magasin permet la gestion des documents XML textuels.

Face au problème de volume, une solution est l'emploi de vues XML similaires avec les vues des SGBD relationnels.

3.4 Stockage multidimensionnel virtuel

But. Le but du stockage virtuel des données multidimensionnelles est d'éviter principalement les redondances des données sources, mais aussi de disposer d'une plus grande souplesse dans la spécification d'un magasin.

Caractéristiques. Le stockage virtuel s'appuie sur une correspondance, ou « mapping », entre un schéma multidimensionnel virtuel et les données sous jacentes (soit les sources, soit l'entrepôt).

Dans le cadre de documents XML orientés données, dans [Li Y. & An, 2005], les auteurs proposent de représenter via un schéma UML le schéma multidimensionnel de leurs sources de données XML (cf. Figure 16). Les sources sont décrites par un formalisme XSchema, converties automatiquement en diagrammes de classes UML. S'inspirant des méthodes objet multidimensionnelles [Luján-Mora et al., 2002], la représentation UML des sources est convertie manuellement en un schéma multidimensionnel. Un moteur OLAP se charge alors de traduire les requêtes multidimensionnelles vers les sources.

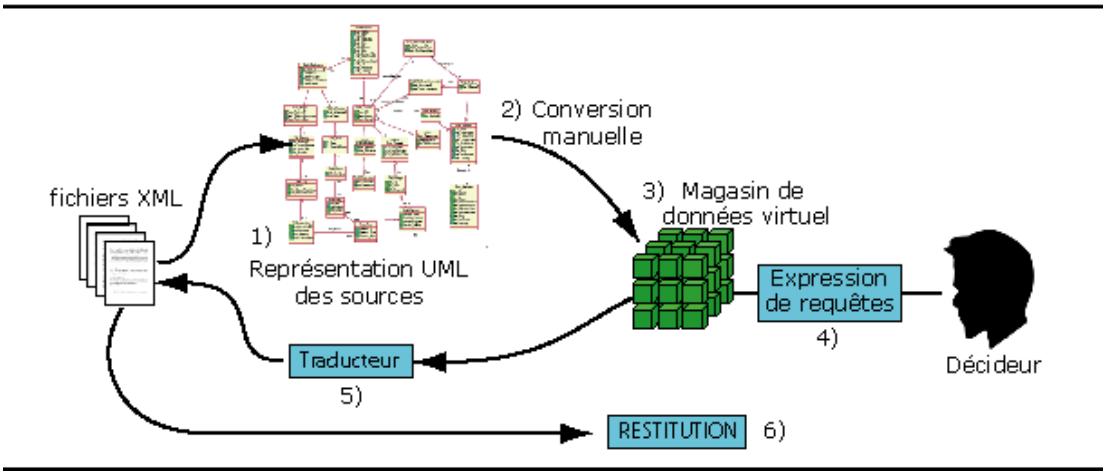


Figure 16 – magasin virtuel directement construit sur des sources XML [Li Y. & An, 2005].

Dans [Nassis et al., 2004], les auteurs présentent un modèle multidimensionnel reposant sur un xFACT. Dans cette approche, un entrepôt de données est modélisé au moyen d'un diagramme de classes UML. Des vues XML [Rajugan et al., 2003] sont définies virtuellement sur l'entrepôt pour modéliser les dimensions : VDim. La structure multidimensionnelle est bâtie au moyen de vues sur un entrepôt de documents XML. Dans ce modèle, les indicateurs sont plus complexes que dans un modèle traditionnel. Concrètement, il s'agit d'un sous-ensemble de classes du diagramme objet représentant l'entrepôt. Les auteurs ne précisent pas les méthodes d'agrégation employées ni la spécification des mesures au sein du xFACT. Les dimensions sont reliées aux mesures via les liaisons présentes dans le diagramme UML.

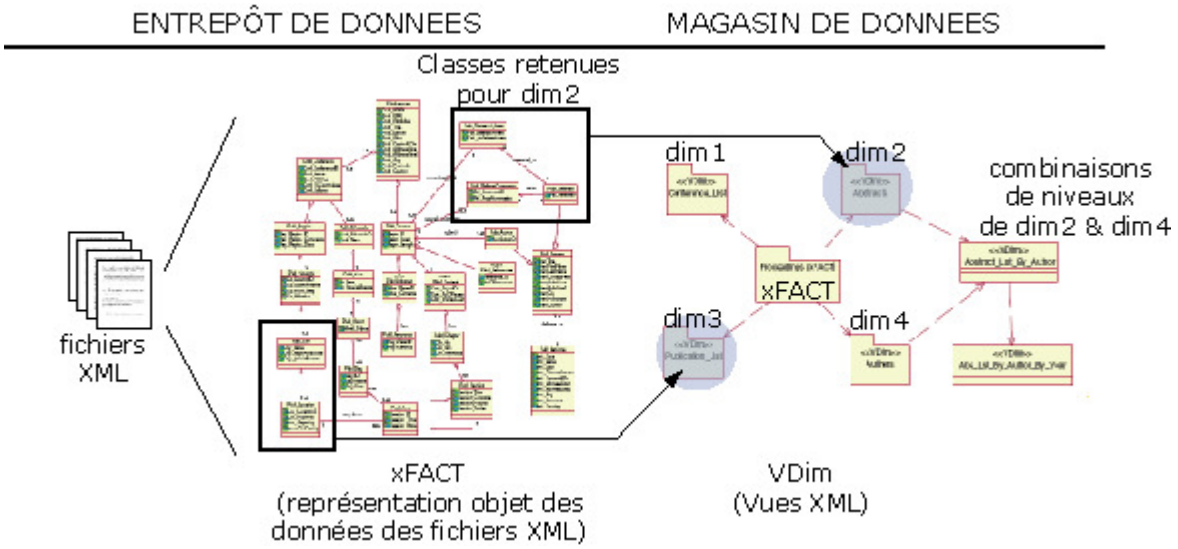


Figure 17 - Constitution des dimensions virtuelles à partir du xFACT [Nassis et al., 2004] .

Avantages/Inconvénients. Cette approche a l'avantage d'éviter une conversion nécessaire pour l'intégration des données XML dans un entrepôt classique avec un format de données spécifique. En outre, l'absence de matérialisation du magasin permet de résoudre d'éventuels problèmes de volumes de données. Néanmoins, l'absence de structuration multidimensionnelle des sources ralentit les traitements et l'expression des requêtes multidimensionnelles sur les sources est très complexe et difficilement appréhendable.

3.5 Bilan

3.5.1 Les modèles ne permettent pas la gestion de documents

En terme de modélisation, les modèles basés sur le cube ont atteint leurs limites il y a quelques années, ils ne sont d'actualités que pour des analyses simples où une modélisation claire des axes d'analyse n'est pas nécessaire. Les modèles conceptuels multidimensionnels, plus récents, permettent de faire abstraction des contraintes logiques et physiques et d'obtenir une vision orientée décideur [Golfarelli et al., 2002]. Ils intègrent la spécification d'axes d'analyses à multiples perspectives fournissant différents niveaux de granularité pour permettre des analyses plus poussées.

Néanmoins pour la modélisation de l'analyse de données issues de documents XML orientés documents, les modèles proposés atteignent leurs limites. Aucun des modèles précédemment cités ne répondent simultanément aux trois objectifs fixés initialement :

- représentation de la structuration hiérarchique des documents,
- faciliter l'analyse d'un contenu essentiellement textuel,
- représentation des liaisons intra ou inter documents.

De part la nature des modèles multidimensionnels, la structure d'un document se retrouve subdivisée en plusieurs éléments conceptuels (mesures et dimensions).

La représentation des liens internes des documents est perdue tandis que la structuration multidimensionnelle des niveaux logiques associés aux modèles ne facilite pas la tâche. En l'état actuel, il est impossible de naviguer au sein des données à analyser via les liens internes. Ainsi l'analyse classique des citations de publications scientifiques nécessite des magasins adaptés avec une redondance dans les données non négligeable.

L'analyse de données textuelles (données qualitatives) ne fournit pas nécessairement un résultat aussi clair et précis que l'analyse de données quantitatives. Ainsi pour l'analyse de données issues de documents orientés documents, il est nécessaire de disposer d'une certaine flexibilité dans la désignation du sujet d'analyse. La spécification de faits dans un schéma conceptuel multidimensionnel se traduit par une performance accrue du système grâce à une matérialisation des vues d'analyses [Harinarayan et al., 1996]. Néanmoins, ceci contraint l'analyste à des solutions d'analyse spécifiées au préalable et pouvant devenir limitatives dans le cadre d'analyses de données textuelles.

3.5.2 La gestion de XML dans les magasins est insuffisante pour les documents

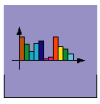
Le stockage de données multidimensionnelles en XML génère une grande redondance dans les données des dimensions. Une solution alternative est le stockage des données factuelles et des données dimensionnelles dans des fichiers séparés les liaisons étant spécifiées par l'intermédiaire d'index, tels que les techniques REF / IDREF [Golfarelli et al., 2001]. Mais cette solution n'a que peu d'intérêt pour des données très structurées par rapport au relationnel.

Les SGBD XML natifs ne sont pas encore complètement matures. Ainsi les magasins XML font face à des problèmes de performance.

Le bénéfice de la structure hiérarchique du format de données est contrebalancé par deux inconvénients : 1) la difficulté à modéliser des dimensions composées de hiérarchies complexes et 2) le parcours de données arborescentes implique une gestion récursive des requêtes. Ainsi le langage tout désigné pour spécifier des requêtes sur des données XML : XQuery imbrique de nombreuses boucles récursives pour parcourir les arborescences des données. Il faut toutefois noter que la constitution d'un magasin virtuel permet de s'affranchir de certains de ces inconvénients en terme de modélisation de dimensions complexes.

Une fois encore, dans le cadre des magasins de données XML, la majorité des travaux de recherche considèrent principalement les données issues de documents XML orientés données (des données fortement structurées) et laissent de côté les documents XML principalement composés de texte.

4 Manipulation OLAP



La présence de données XML dans un système décisionnel ouvre de nouvelles perspectives. Cette section est divisée en deux : premièrement, cette section s'intéresse à la manipulation multidimensionnelle en général. Deuxièmement, la section présente les possibilités offertes par le format XML en terme de manipulation multidimensionnelle.

4.1 Manipulation multidimensionnelle

4.1.1 Opérations de manipulation

En terme de manipulation multidimensionnelle, les premiers travaux sur les manipulations OLAP ont étendu les opérateurs de l'algèbre relationnelle pour le modèle en cube [Gray et al., 1996]], [Agrawal et al., 1997] (une transcription SQL des opérations est disponible dans [Agrawal et al., 1995]), [Li C. & Wang, 1996], [Gyssens & Lakshmanan, 1997] et [Datta & Thomas, 1999]. Pour contourner l'inadaptabilité de l'algèbre relationnelle en ce qui concerne la manipulation de structures multidimensionnelles dans un contexte OLAP, de nombreux travaux ont proposé des opérateurs et des opérations pour spécifier et manipuler un cube [Cabibbo & Torlone, 1997] et [Cabibbo & Torlone, 1998], [Abelló et al., 2003], [Pedersen T.B. et al., 2001] et [Franconi & Kamble, 2004].

De manière plus complète, dans [Ravat et al., 2007e], nous avons défini un langage algébrique de manipulation complétant ces propositions. Notons aussi un début de formalisation de langage de manipulation multidimensionnel appliqué à la spécification d'analyses de données complexes dans [Messaoud, 2006].

Malgré l'absence de standard en terme de modèle et d'accord sur un noyau d'opérateurs de manipulation OLAP, la plupart des propositions offrent un support partiel pour les catégories d'opérations suivantes :

- **Opérations de forage**. Ces opérations permettent la navigation au moyen de la structure hiérarchique des axes d'analyses, afin de permettre l'analyse d'un indicateur avec plus ou moins de précisions. Le forage vers le haut (*roll-up*) consiste à analyser les données en fonction d'un niveau de granularité moins détaillé. L'inverse, le forage vers le bas (*drill-down*) permet d'analyser les données avec un niveau plus fin.
- **Opérations de sélection**. Ces opérations permettent à un utilisateur de restreindre l'ensemble des données analysées. La spécification d'une « tranche de cube » (*slice*) consiste à exprimer une restriction sur une des données de l'un des axes d'analyse. La spécification d'un « sous-cube » (*dice*) consiste à exprimer une restriction sur les données d'un indicateur d'analyse.
- **Opérations de rotation**. Ces opérations permettent la réorientation d'une analyse. Elles permettent de changer l'un des axes d'analyse en cours d'utilisation (*rotation de dimension*), de changer le sujet de l'analyse (*rotation de fait* ou *drill-across*) et de changer de perspective d'analyse (*rotation de hiérarchie*).

Des auteurs ont aussi proposé des opérations additionnelles :

- **Opérations de modifications du sujet d'analyse**. Ces opérations permettent d'ajouter ou de retirer un indicateur d'analyse d'une analyse en cours.

- **Opérations de modifications d'une dimension**. Ces opérations permettent l'ajout d'attributs en tant qu'indicateur d'analyse (*push*) ou de convertir un indicateur d'analyse en attribut (*pull*).
- **Opérations d'ordonnancements**. Ces opérations permettent de changer l'ordre des valeurs des attributs des dimensions (*order*) ou d'insérer un attribut dans une autre hiérarchie (*nest*).
- **Opérateurs binaires**. Certains auteurs proposent aussi l'emploi des opérations binaires ensemblistes : *union*, *différence* et *intersection*. Certains travaux ont aussi proposé la notion de jointure (*join*) inspirée de la jointure relationnelle, mais d'un intérêt limité dans un environnement multidimensionnel. Il est intéressant de noter que les opérateurs binaires sont des opérateurs nécessitant de très fortes contraintes. Par exemple, l'union entre deux structures multidimensionnelles nécessite une compatibilité presque complète des deux structures [Ravat et al., 2007e].

A partir de ces catégories d'opérations il est possible d'effectuer une comparaison entre les différents opérateurs proposés.

4.1.2 Comparaison des opérateurs et bilan

Afin de comparer le positionnement des opérateurs des différentes propositions, les opérations sont présentées en fonction de la catégorie à laquelle ils sont associés (cf. Tableau 3). Au sein de chaque cellule du tableau, le nom spécifique de l'opération est présenté. Une liste d'une partie de ces opérations peut être trouvée dans [Rafanelli, 2003], mais sans comparaison.

Dans le tableau, la ligne *structure du modèle* résume la structure de sortie des opérations définies par les auteurs. Il est possible de trouver, le cube, des tables bidimensionnelles ou encore une structure multidimensionnelle spécifique. Dans [Cabibbo & Torlone, 1997] et [Cabibbo & Torlone, 1998], les auteurs emploient une structure multidimensionnelle nommée *f-table*. Le support de l'opérateur Cube [Gray et al., 1996] et [Gray et al., 1997] n'est pas présenté. Toutefois la plupart des propositions supportent cet opérateur d'agrégation généralisée ([Agrawal et al., 1997], [Gyssens & Lakshmanan, 1997], [Ravat et al., 2007e]…).

Certaines propositions complètent leur offre par quelques opérateurs additionnels ou encore un langage de calcul associé aux opérateurs algébriques. Ce langage de calcul est destiné à permettre la spécification d'un langage déclaratif, l'algèbre étant destinée au système en tant que langage procédural d'optimisation.

Tableau 3 - Comparaison des différents langages multidimensionnels.

Travaux de Recherche / Operations		(Grouping Algebra) Li C. & Wang 1996	Agrawal *et al.*, 1997	Gyssens & Lakshmanan,1997	(MD) Cabibbo & Torlone 1997, 1998	Lehner, 1998	Datta & Thomas, 1999	Pedersen *et al.*, 2001	(YAM²) Abelló *et al.*, 2002, 2006	(GMD) Franconi *et al.*, 2004	Ravat *et al.*, 2006	Ravat *et al.*, 2007e
Forage	**Niveau plus détaillé**	Roll ,Cube	Join			DrillDown[2], Split	(Push, Pull) + Join		DrillDown		Drill Down	DrillDown
	Niveau moins détaillé	Roll, Aggregation	Merge	Summerization	RollUp, Aggregation	RollUp, Merge, Aggregation	-	Aggregation	RollUp		RollUp	RollUp
Sélection	**Valeurs factuelles**			Slice (Selection)	Selection		Restriction + Pull		Dice, Projection			Select
	Valeurs dimensionnelles		Restriction	Dice (Selection)	Selection		Restriction	Selection		Slice, Multi-Slice[3]		Select
Rotation	**Fait**								DrillAcross		FRotate	FRotate
	Dimension								ChangeBase		DRotate	Rotate
	Hiérarchie										HRotate	HRotate
Modification de fait	**Ajout d'une mesure**		Projection	Projection						Derived measures		AddM
	Suppression d'une mesure		Projection	Projection								DelM
Modification de dimension	**Réduction de dimension**	Cube Aggregation	Projection, Destroy-Dimension		Simple Projection		Partition	Projection		Projection		Display
	Push		Push	Fold[4]			Push					Push
	Pull		Pull	Unfold			Pull					Pull
Ordonnancement	**Oder**			Classification								Switch, Order
	imbrication	Transfer									Nest	Nest
Opérateurs binaires	**Union**	Union[6]	Union[6]	Union[6]			Union[5]	Union[5]	Union[6]	Union[6]		Union
	Intersection		Intersection	Intersection			2 Difference			Intersection		Intersect
	Différence		Difference	Difference			Difference	Difference		Difference		Difference
	Jointure	RC-Join (Relation vers dimension)	join cubes	join cubes	Join[1]		Cartesian product + Restriction	Identity-based Join, Group		join cubes		
Structure du modèle		Cube	Cube	2D-Table	MD (f-table)	MD	Cube	MD	Cube	Cube	2D-Table (MT)	2D-Table (MT)
Autres opérations		Add dimension		Produit cartésien	Produit cartésien		Produit cartésien					
Commentaires				définit un langage de calcul	définit un langage graphique + "query calculus"				définit une traduction en SQL		définit un langage assertionnel	définit un langage graphique

MD=Multidimensionnel; [1]=sans restriction; [2]=pas de conservation de hiérarchies; [3]=spécifié sur une zone; [4]=push généralisé; [5]=sur dimensions; [6]=cubes identiques seulement;

Quasiment aucune proposition n'offre un support complet pour toutes les opérations de manipulations multidimensionnelles. Toutefois, nous avons proposé dans [Ravat et al., 2007e] un support de l'ensemble des catégories d'opérations. Cette proposition est basée sur un modèle en constellation et une structure de visualisation de table multidimensionnelle.

Dans le contexte d'analyse documents XML orientés documents, une reformulation de ces opérateurs sera nécessaire pour prendre en charge les spécificités des documents orientés documents.

4.2 Manipulation multidimensionnelle XML

La manipulation OLAP orientée XML permet de s'affranchir de la structure multidimensionnelle classique telle que le schéma en étoile [Kimball, 1996]. Il est ainsi possible d'exploiter la structure arborescente spécifique au XML, voire de gérer des données dans un format semi-structuré.

But. Le but d'effectuer des manipulations OLAP sur un espace de stockage XML est de disposer de la souplesse équivalente à un entrepôt de données classique mais au format XML.

Caractéristiques. Dans un tel environnement, les données sont dans un format XML et sont hiérarchiquement structurées. Les requêtes multidimensionnelles sont exprimées avec un langage adapté, tel que XQuery.

Dans [Khrouf & Soulé-Dupuy, 2004] les auteurs présentent un système capable d'effectuer des analyses multidimensionnelles sur un entrepôt de documents. Des documents XML orientés documents sont stockés dans l'entrepôt et classifiés selon leurs structures. Le système permet d'effectuer des analyses multidimensionnelles sur les structures.

Bien que ce ne soit pas le cas dans [Khrouf & Soulé-Dupuy, 2004], l'expression de requêtes multidimensionnelles sur des données XML s'exprime dans un langage de requête adapté. Tout comme en SQL, l'expression de telles requêtes s'avère très complexe. Deux propositions issues des laboratoires d'IBM, [Bordawekar & Lang, 2005] et [Beyer et al., 2005], présentent l'ajout de l'opérateur Group By au langage de requête XQuery pour simplifier l'expression des requêtes multidimensionnelles.

Avantages/Inconvénients. La solution d'analyser directement les données d'un entrepôt de données XML est similaire à l'approche de magasin de données virtuelles (cf. section 3.4). Ces approches sont une première étape vers un environnement décisionnel XML. Néanmoins, bien que l'utilisateur puisse exprimer des requêtes multidimensionnelles, leur complexité et l'absence de schéma multidimensionnel rendent la tâche difficile.

Bien que la gestion de documents XML orientés documents soit abordée dans [Khrouf & Soulé-Dupuy, 2004], des analyses OLAP sur le contenu des documents n'est pas possible.

4.3 Bilan

Un ensemble d'opérations multidimensionnelles ont été proposées depuis une dizaine d'années. Le format XML nécessite des opérateurs multidimensionnels permettant la gestion de semi-structuré. Des opérateurs de bas niveau ont été proposés (tels que le GROUP BY). Toutefois, le langage XQuery est moins mature que son grand frère SQL et par conséquent l'expression d'analyses multidimensionnelles est plus difficile.

La majeure partie des propositions s'est appuyée sur l'analyse de données issues de documents XML très structurés (documents orientés données). Seul les auteurs de [Khrouf & Soulé-Dupuy, 2004] se sont focalisés sur des documents orientés documents, mais ils ne favorisent pas l'analyse du contenu. En revanche, dans [Park et al., 2005], les auteurs proposent une introduction à l'analyse du contenu de documents, mais l'environnement proposé n'est que peu détaillé et aucune opération de manipulation n'est proposée.

La manipulation multidimensionnelle permet la spécification d'analyses à partir des concepts disponibles. Ces analyses synthétisent les informations à partir de fonctions d'agrégations.

5 Analyse OLAP

Cette section présente les travaux de recherche qui ont proposé l'analyse de données issues de documents. Toutefois, ces analyses reposent toutes sur des mesures très simples qui consistent à compter des documents. La section se poursuit en présentant les principes employés pour obtenir la synthèse de informations avec des fonctions d'agrégations. Enfin elle se termine sur les fonctions avancées adaptées aux XML et aux données textuelles.

5.1 Analyse OLAP de documents

Dans son ouvrage [Sullivan, 2001], Dan Sullivan, suggère l'emploi de la fouille de texte, « text mining », pour l'analyse du contenu d'un entrepôt de document. Toutefois, notre idée est de pouvoir analyser le contenu de documents tout en restant dans un environnement OLAP ; c'est-à-dire, de disposer non seulement d'entrepôt de documents orientés analyse, mais aussi de disposer d'outils d'analyse OLAP adaptés.

Définition. Par *analyse multidimensionnelle de documents*, nous entendons l'analyse dans un environnement OLAP des sources de données textuelles telles que des documents orientés documents.

Quatre travaux ont proposé une analyse multidimensionnelle OLAP de documents. Pour pouvoir comparer les différents travaux, les schémas multidimensionnels sont représentés au moyen d'un même formalise, proposé dans [Ravat et al., 2007e].

Les trois figures suivantes, issues des propositions de [Tseng & Chou, 2006], représentent des propositions qui évaluent le contenu de documents orientés document en comptant les occurrences des documents en fonction de critères disposés en tant que dimensions (dates, auteurs…). Chaque instance du fait correspond à un certain nombre de documents. Le système conserve un lien avec les documents d'origine, stockés à part pour permettre à l'utilisateur une éventuelle consultation de ces documents. Au sein de ces approches, les auteurs n'hésitent pas à employer des dimensions avec un seul niveau de détail, à savoir un unique paramètre. Par exemple, la dimension *CREATEUR* dans la Figure 18.

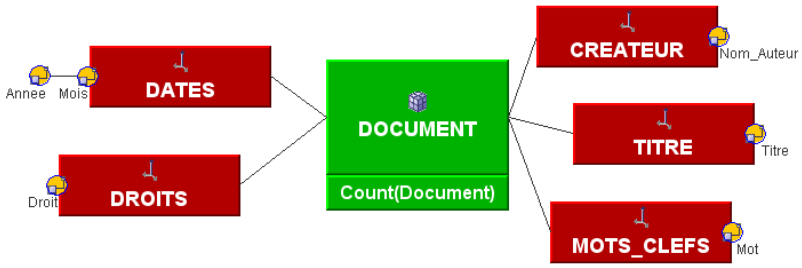


Figure 18. [Tseng & Chou, 2006] Comptage de documents généraux (mails, articles, pages Web…) avec des dimensions issues des méta-données des recommandations du Dublin Core [Dublin Core, 2007].

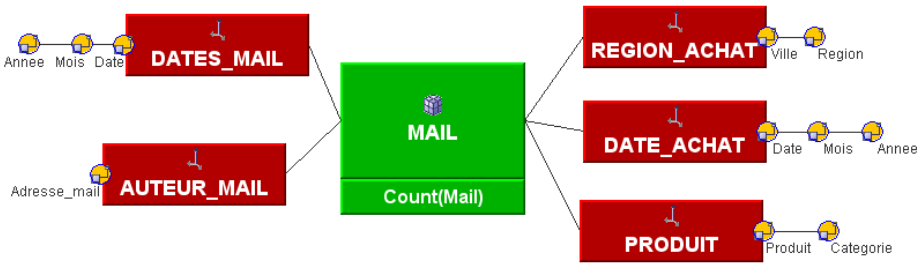


Figure 19. [Tseng & Chou, 2006] Comptage de mails de plaintes d'acheteurs de produits.

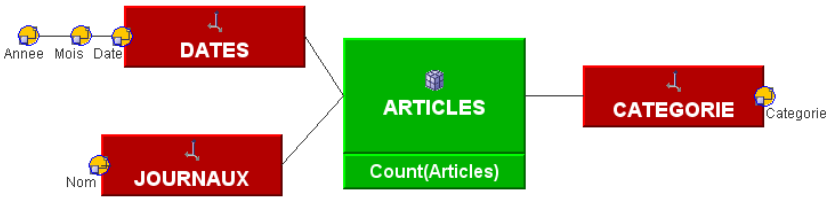


Figure 20. [Tseng & Chou, 2006] Magasin de données bâti sur un entrepôt de document contenant des publications scientifiques issues de journaux (Elsevier, ACM…).

Il est à noter que dans les exemples de la Figure 18, la dimension MOTS_CLEFS et dans la Figure 20, la dimension CATEGORIE sont *non-strictes* [Malinowski & Zimányi, 2006]. C'est-à-dire qu'un document peut être caractérisé par plusieurs mots-clefs et un article peut

appartenir à plusieurs catégories. En l'état actuel, les notations de notre modèle [Ravat et al., 2007e] ne permettent pas la différenciation de ce type de hiérarchies.

Dans [Mothe et al., 2003] les auteurs proposent l'analyse présentée en Figure 21 avec l'environnement DocCube. Le but est de voir la répartition des documents en fonction des mots-clefs employés dans ces documents, ceci dans le but de permettre des requêtes de recherche d'information plus précises par un emploi des mots-clefs les mieux adaptés.

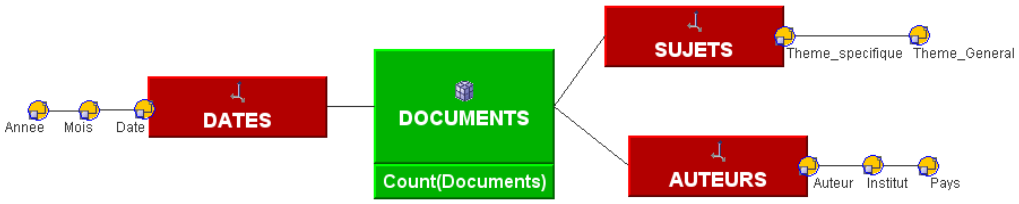


Figure 21. [Mothe et al., 2003] Analyse de l'emploi de mots clés dans des documents.

Les exemples suivants ne travaillent plus sur un comptage de documents, mais un comptage de mots clefs. Le principe est identique, mais l'analyse est légèrement différente puisque cette fois il est possible d'obtenir l'importance de l'utilisation d'un mot-clef dans un document.

Dans l'exemple présenté en Figure 22, [Keith et al., 2005] proposent d'utiliser l'environnement OLAP pour générer des matrices de co-occurrence de termes en fonction d'ouvrages pour comparer l'emploi des termes.

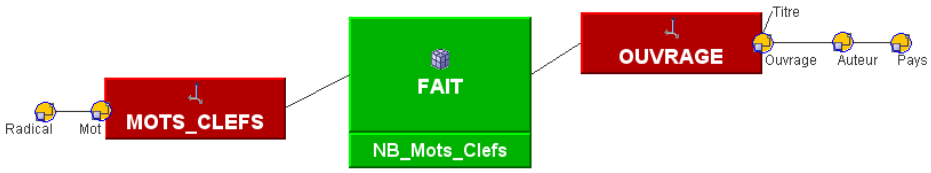


Figure 22. [Keith et al., 2005] Analyse bibliométrique de l'utilisation de mots clés dans des ouvrages.

L'exemple en Figure 23 et extrait de [McCabe et al., 2000], est très similaire à celui de DocCube, mais est destiné à analyser des archives gouvernementales, d'où la dimension localisation permettant de retrouver le lieu d'entreposage du document d'origine.

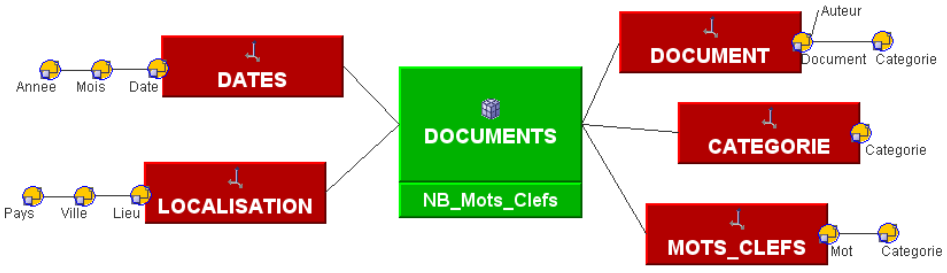


Figure 23. [McCabe et al., 2000] Analyse de documents gouvernementaux.

Avec ces travaux, des possibilités d'analyse de documents ont été proposées. Mais, basées sur des comptages de documents ou de mots-clefs, ces analyses restent limitées. De plus aucune ne permet une analyse plus en détails du contenu des documents. Les plus poussées se limitent aux méta-données des documents telles que celles spécifiées dans les recommandations du Dublin Core [Dublin Core, 2007].

Dans les précédentes propositions exposées, la synthèse de l'information au sein des analyses est numérique. Toutefois le contenu des documents est textuel. Ainsi des fonctions d'agrégations doivent être adaptées au XML et au texte…

5.2 Synthèse d'informations (fonctions d'agrégation)

Les fonctions d'agrégation sont un élément important de la génération de rapports sur des bases de données [Klug, 1982]. La spécification de fonctions d'agrégation dans les bases de données relationnelles a été une problématique active depuis la définition de l'algèbre et du calcul relationnel dans [Codd, 1970] et [Codd, 1972]. Visionnaire, Anthony Klug [Klug, 1982] s'est penché sur la spécification de fonctions d'agrégation, principe qui était encore mal compris à l'époque. Ces travaux furent repris et étendus dans [Özsoyoglu et al., 1985] et [Özsoyoglu et al., 1987] pour une application au sein des bases de données statistiques en se reposant sur la notion de « *summary tables* », tables bidimensionnelles à l'origine des tableaux croisés dynamiques. Enfin, Bültzingsloewen [Bültzingsloewen, 1987] a proposé une adaptation de la spécification formelle des fonctions présentée dans [Klug, 1982] afin d'être conforme avec le langage de requête SQL.

Plus complexe, l'agrégation multidimensionnelle a commencé avec la spécification d'agrégats et le principe d'agrégation dans les bases de données statistiques avec les propositions de [Özsoyoglu et al., 1985] (le lecteur est invité à consulter [Lenz & Shoshani, 1997], [Shoshani, 2003] et [Torlone, 2003] pour plus de détails concernant les bases de données statistiques). Cette problématique fut reprise avec l'apparition des premiers modèles cubes pour OLAP : [Li C. & Wang, 1996] [Agrawal et al., 1997] et [Gyssens & Lakshmanan, 1997]. Par la suite, l'apparition de notion de hiérarchisation dans les données représentant les axes d'analyse a aussi donné lieu à de nouvelles propositions [Lenz & Shoshani, 1997], [Jagadish et al., 1999], [Pourrabas & Rafanelli, 2000] et [Pourrabas & Rafanelli, 2003].

5.2.1 Fonctions d'agrégation classiques

Les SGBD relationnels sont accompagnés d'une série de fonctions d'agrégation classiques. Il s'agit de fonctions simples qui regroupent un ensemble de valeurs en une valeur unique. Parmi ces fonctions on trouve généralement les cinq fonctions suivantes :

- somme (SUM) : cette fonction retourne la somme numérique de l'agrégat ;
- comptage (COUNT) : cette fonction compte le nombre d'instances dans un agrégat ;
- minimum (MIN) : cette fonction retourne la plus petite valeur d'un agrégat ;
- maximum (MAX) : cette fonction retourne la plus grande valeur d'un agrégat ;
- moyenne (AVERAGE) : cette fonction retourne la valeur moyenne d'un agrégat.

Ce jeu de fonctions a été augmenté de fonctions statistiques pour permettre la génération de rapports plus complets dans l'environnement des bases de données statistiques [Shoshani, 2003]. Il s'agit essentiellement de fonctions statistiques, telles que le calcul de *moyennes mobiles*, de *barycentres* ou encore de *médianes* ou d'*écart type*. De plus les SGBD récents (par exemple Oracle[10]) offrent une interface de programmation permettant à un programmeur de spécifier ses propres fonctions d'agrégation.

[10] Oracle Database 11*g* http://www.oracle.com/database/index.html

5.2.2 Fonctions d'agrégation avancées

Récemment diverses fonctions d'agrégation évoluées ont vue le jour. Ces fonctions proviennent de différents domaines :

- le multidimensionnel,
- les systèmes d'information géographiques (SIG),
- la fouille de données.

Le décisionnel a vu la création d'un opérateur de regroupement CUBE [Gray et al., 1996] et [Gray et al., 1997], qui emploie de manière intensive les fonctions d'agrégation dans un environnement décisionnel. Il s'agit d'un opérateur calculant les totaux généralisés d'une sélection de données.

Dans le cadre des SIG, un domaine décisionnel apparu, notamment avec le SOLAP (Spatial-OLAP) [Rivest et al., 2005]. Des fonctions spécifiques adaptées aux données géographiques virent le jour [Stefanovic et al., 2000]. Les données géographiques étant stockées sous les forme de points, segments et surfaces, les fonctions adaptées se chargent de permettre un regroupement de ces types de données (avec par exemple le barycentre de plusieurs points, la surface moyenne,…).

Récemment des fonctions d'agrégation issues de la fouille de données commencent à voir le jour au sein de l'environnement OLAP. L'une des plus importantes en matière de production scientifique depuis le cube de Jim Gray est la fonction d'agrégation SKYLINE [Börzsönyi et al., 2001]. Il s'agit d'une fonction cherchant à résoudre le problème de maximisation de vecteurs (Vector Maximisation Problem—VMP) [Kung et al., 1975] et de rechercher une solution maximale ou minimale pour un problème à (au moins) deux variables. Par exemple : la fonction permet de rechercher les hôtels les moins chers en fonction de leur distance à la plage voisine. Dans ce cas précis, il s'agit de trouver les hôtels qui minimisent le coût et la distance.

Dans la même lignée, des opérateurs de classification issus de fouille de données ont été inclus dans les systèmes OLAP. Parmi ceux-ci, *OpAC* [Messaoud et al., 2004] est un opérateur regroupant les instances selon une classification ascendante hiérarchique (CAH). Néanmoins, ces fonctions ne sont guère utiles dans l'environnement OLAP car elles détruisent systématiquement l'agencement hiérarchique des axes d'analyses, réduisant à néant les possibilités de forage par la suite. Ainsi les résultats obtenus par ce type de fonction d'agrégation sortent de l'environnement OLAP car de nombreuses opérations de manipulation ne peuvent plus s'y appliquer.

5.2.3 Agrégation et données XML

Lorsqu'un magasin est matérialisé dans un environnement XML, les fonctions d'agrégation classique opèrent sur des données fortement structurées. Toutefois, ceci est loin d'être le cas de tous les types de documents XML. Ainsi des propositions ont vue le jour dans le cadre d'agrégation de données XML. Ces propositions peuvent être regroupées en deux catégories :

- agrégation avec exploitation de la structure XML (documents très structurés),
- agrégation pour des données textuelles.

La première approche regroupe les spécifications d'opérateurs permettant d'agréger des données en exploitant la structure arborescente XML. Ces opérateurs s'inspirent des opérateurs d'agrégation de l'environnement OLAP tels que AGGREGATE [Gyssens &

Lakshmanan, 1997], ROLL [Li C. & Wang, 1996] ou encore CUBE [Gray et al., 1996]. Dans [Wang et al., 2003] et [Wang et al., 2005], les auteurs présentent un opérateur d'agrégation de structure XML : XAGGREGATION. Cet opérateur permet le regroupement de données XML selon leur structure arborescente dans le document par fusion d'arbres XML. Cette fonction agrége systématiquement toute valeur de la mesure sous un nœud jugé commun à plusieurs valeurs de mesure. Cette valeur agrégée est alors retournée un nombre de fois proportionnel au nombre des différentes valeurs du paramètre sélectionné pour l'agrégation en fonction du paramétrage de la fonction d'agrégation. Cette fonction a comme problème de fusionner tous les résultats en un seul ou bien de retourner de multiples fois la même valeur. Ceci peut provoquer bon nombre d'incohérences d'agrégation [Horner et al., 2004]. Ainsi la structure obtenue reste difficilement exploitable. Très récemment, dans [Wiwatwattana et al., 2007], les auteurs redéfinissent l'opérateur CUBE [Gray et al., 1996] : X^3 pour agréger des données XML semi-structurées. L'opérateur X^3 emploie la relaxation de chemins arborescents [Amer-Yahia et al., 2002] pour permettre une plus grande flexibilité quand à la spécification d'agrégation sur des données n'ayant pas tout à fait la même structure.

Toutefois ces opérateurs effectuent des agrégations de structures XML et s'appuient sur les fonctions d'agrégation classiques (sum, avg, min,…) pour effectuer l'agrégation des données analysées. Par conséquent l'agrégation de données non numériques (comme par exemple le contenu de documents textuels) ne peut-être prise en charge par ces environnements. Globalement ces fonctions correspondent à la réécriture du principe de regroupement des données (instruction GROUP BY en relationnel).

Dans la seconde approche, [Park et al., 2005], les auteurs suggèrent l'emploi de techniques de fouille de texte pour l'analyse du contenu de documents. Plusieurs fonctions sont proposées : SUMMARY, permet la génération d'un résumé du texte agrégé ; TOP_KEYWORDS, sélectionne les n principaux mots-clefs du texte a agréger ; TOPIC, extrait le sujet d'un bloc de texte ; et CLUSTERING est une fonction qui partitionne des textes en fonction de leur contenu.

Les auteurs ne donnent aucune précision concernant le fonctionnement de ces fonctions ou leur implantation.

5.3 Analyse OLAP de données textuelles est envisageable

Depuis l'apparition des systèmes d'aide à la décision, des travaux de recherche ont proposés l'analyse de documents avec un environnement classique. Mais les analyses restent simples et se limitent aux capacités des fonctions d'agrégation disponibles. Ces analyses ne permettent pas l'analyse du contenu de documents.

Les fonctions d'agrégation ont évoluées de leur côté, avec notamment :

- des fonctions de statistiques (médiane,…),
- des fonctions de fouilles de données (SKYLINE…),
- des fonctions de classification (OPAC,…),
- des fonctions spécifiques pour XML,
- quelques rares fonctions de fouille de texte (TOP_KEYWORD, SUMMARY…).

Dans le cadre de cette thèse, les fonctions d'agrégations suggérées dans [Park et al., 2005] sont les plus intéressantes car elles permettent une solution à notre problématique, à savoir l'analyse des 80% des données issues du système d'information des entreprises qui reste hors de portée des systèmes OLAP. Toutefois, dans [Park et al., 2005], aucune des fonctions proposées n'est présentée en détails.

6 Bilan de l'état de l'art

Après avoir résumé les trois principaux poins abordés dans cet état de l'art, cette section se poursuit sur une présentation détaillée de la problématique et des objectifs de la thèse.

6.1 Conclusion

Les trois sections précédentes ont exposé l'intégration de la technologie XML dans les niveaux de l'architecture des systèmes décisionnels avec :

- entrepôts des données XML,
- magasins de données XML,
- restitution et manipulation OLAP à partir de données XML.

Les entrepôts de données XML permettent une homogénéité avec les sources XML de données. Les magasins XML bénéficient de la structure arborescente XML pour représenter les données de manière multidimensionnelle. Enfin certains opérateurs ont été adaptés pour permettre une gestion du format XML.

Le Tableau 4 montre la répartition des différents travaux qui ont abordé les entrepôts et la technologie XML. En ligne, on retrouve les différentes sections de ce document. Les deux dernières lignes permettent de visualiser le type de document XML géré par les différentes propositions. Les travaux fusionnés en une seule colonne sont des travaux qui se suivent.

Tableau 4 - Comparaison des différentes propositions.

	Golfarelli et al., 2001	Vrdoljak et al., 2003	Vrdoljak et al., 2006	Pokorny, 2001	Niemi et al., 2002	Zhang et al., 2003	Pokorny et al., 2002	Nassis et al., 2004	Rusu et al., 2005	Li Y. & An, 2005	Park et al., 2005	Boussaid et al., 2006	Baril & Bellashene, 2004	Jensen et al., 2001	Pedersen D. et al., 2002	Yin & Pedersen T.B, 2004	Khrouf & Soule-Dupuy, 2004	Beyer et al., 2005	Bordawekar & Lang, 2005	Wang et al., 2003	Wang et al., 2005	Wiwatwattana et al., 2007	Sullivan, 2001
Entrepôts XML																							
Entrepôts de données XML	x		x	x	x	x	x	x	x	x		x		x				x		x	x	x	x
Entrepôts de documents XML											x		x				x						
Magasin de données XML																							
Intégration multidimensionnelle	x		x	x	x			x						x									
Sockage XML multidimensionnel	(x)					x		x		x	x									(x)	(x)		
Stockage multidimensionnel virtuel							x		x														
Manipulation OLAP																							
Stockage multidimensionnel											x									x	x		
Stockage non multidimensionnel																	x	x					
Type de document XML																							
Document orienté données	x		x	x	x	x	x			x		x		x				x		x	x	x	
Document orienté documents								x	x		x		x				x						x

(x) = sous entendu dans la proposition

Un état de l'art alternatif peut être trouvé dans [Pérez et al., 2006]. Il s'agit d'une vision orienté recherche d'information positionnant les propositions [Pérez et al., 2005][Pérez et al., 2007] vis-à-vis des travaux concernant XML et les systèmes décisionnels.

L'emploi d'entrepôts pour le stockage de données XML a prouvé son intérêt pour les entrepôts de contenus avec des solutions telles que Xyleme. Ceci permet une solution pour entreposer des documents XML principalement constitués de données textuelles. Néanmoins,

un entrepôt XML n'a que peu d'avantages par rapport à un entrepôt classique pour des documents XML orientés données, hormis d'éviter un processus de conversion dans le format natif de l'entrepôt classique (relationnel, objet...). Il faut aussi noter le manque de maturité des entrepôts XML ainsi qu'un volume important dû au format de stockage du XML.

De leur côté, les magasins de données XML ont un intérêt pour l'analyse de données complexes, car ils permettent de gérer ce type de données alors qu'un magasin classique ne pourrait le faire avec son format natif. De plus, malgré de nombreuses propositions de modélisation multidimensionnelle, aucune n'est adapté pour la modélisation d'analyses de contenus de documents essentiellement composés de données textuelles.

D'un point de vue restitution, bien que quelques travaux se soient portés sur des opérateurs adaptés, ces travaux se limitent à la manipulation de données issues de documents XML orientés données. Toutefois il faut noter une tentative d'analyser des documents XML textuels au moyen de fonctions d'agrégation spécifiques, mais aucun détail concernant cette proposition n'est fourni par les auteurs.

Il existe clairement deux catégories de travaux concernant XML et les systèmes d'aide à la décision : ceux qui prennent en compte les documents XML principalement constitués de texte (orientés documents) et ceux qui ne les prennent pas en compte. La grande majorité des travaux se reposent sur des documents très structurés (orientés données).

Parmi les propositions prenant en compte les documents XML composés de texte (orientés documents) très peu suggèrent l'analyse du contenu des documents. Un début d'analyse en ligne de documents a néanmoins été proposé dans certains travaux. Ces travaux se limitent tout de même à des indicateurs quantitatifs (le comptage de documents ou de mots-clefs). Seule la proposition de [Park et al., 2005] est la plus avancée, mais elle n'expose pas de modèle conceptuel permettant de représenter les spécificités des documents.

Ainsi, avec un processus d'analyse en ligne adapté, de nouvelles perspectives s'ouvrent sur l'analyse multidimensionnelle de données en incluant désormais les documents.

Désormais, à la vue de la disponibilité des sources XML au sein des entreprises ou bien sur le Web, les documents structurés ou semi-structurés représentent une source de données envisageable pour les processus OLAP. L'intégration de **documents orientés données** a déjà fait l'objet de nombreuses études telles que [Jensen et al., 2001], [Golfarelli et al., 2001]. Ainsi cette thèse se focalise sur l'ajout de **documents orientés documents** (contenu constitués de données textuelles) dans l'analyse en ligne au sein de systèmes d'aide à la décision.

6.2 Problématique et objectifs de la thèse

Afin de fournir des capacités d'analyse plus exhaustives, les systèmes d'aide à la décision devraient pouvoir permettre l'exploitation de 100% des données des systèmes d'information des entreprises. Ainsi, les analystes devraient pouvoir intégrer des documents ou des données issues du Web directement dans leur processus d'analyse. Ne pas prendre en compte ces données mènent inévitablement à l'omission d'informations pertinentes durant un processus de prise de décision important voire même l'inclusion de données non pertinentes générant ainsi des analyses approximatives ou erronées [Tseng & Chou, 2006].

En allant au-delà des propos de Fankhauser [Fankhauser & Klement, 2003], nous pensons que la technologie XML permet d'envisager l'intégration de documents textuels dans un environnement OLAP. Ainsi, notre problématique s'articule autour de l'analyse

multidimensionnelle du contenu de documents principalement constitués de données textuelles.

L'environnement OLAP actuel ne prend pas en compte l'analyse de données textuelles et ces données disposent d'une structure et d'un contenu actuellement inexploités faute de moyens.

6.2.1 Sujet d'analyse : différentes approches

Le contenu de documents textuels est largement constitué de données textuelles. A partir de ce type de données, trois approches possibles pour l'analyse OLAP de documents sont envisageables :

- L'emploi de *mesures numériques classiques* : dans ce cadre, les mesures correspondent toutes à des comptages d'instances (de documents, de mots-clefs…). Ce cadre autorise l'emploi de l'environnement OLAP classique mais ne permet que des analyses limitées.
- L'emploi de *mesures numériques élaborées* (ou calculées) : dans ce cadre l'environnement se ramène à l'emploi de mesures numériques pour analyser du texte (par exemple des vecteurs). Bien que ce type de mesure permette des analyses poussées, il est difficilement interprétable. De plus il nécessite un environnement d'analyse multidimensionnel adapté.
- L'emploi de *mesures textuelles* : dans ce cadre, l'environnement emploie directement du texte extrait des documents en guise de mesures. Ce type de mesure permet des analyses poussées, tout en restant interprétables. Néanmoins cela nécessite un environnement d'analyse multidimensionnel adapté.

Dans le cadre de nos travaux, nous proposons de gérer les mesures numériques classiques et les mesures textuelles. Cette approche a l'avantage de permettre :

- au minimum, les mêmes analyses que la première approche,
- l'emploi de mesures ne nécessitant pas une interprétation complexe (en comparaison avec la seconde approche, les mesures numériques élaborées),
- des analyses poussées ouvrant de nouvelles perspectives en terme d'analyse en ligne.

Le but étant de permettre l'emploi de processus d'analyse en ligne sur une architecture stockant des documents orientés texte. Notre problématique est donc d'intégrer aussi les spécificités de ces documents.

6.2.2 Intégration des caractéristiques des documents

La modélisation multidimensionnelle conceptuelle au sein d'un magasin de données permet une représentation des concepts indépendamment de toute contrainte d'implantation logique ou physique. Les données issues de documents orientés documents ont quelques spécificités :

- données essentiellement textuelles,
- données structurées de manières hiérarchiques,
- liaisons intra ou inter documents (références, citations, liens hypertextes…).

L'idéal serait de disposer d'un modèle multidimensionnel qui permette de modéliser l'ensemble de ces caractéristiques tout en préservant les bénéfices déjà acquis par dix années de recherche en modélisation de bases de données multidimensionnelles (BDM). BDM, pour

lesquelles nous devons constater qu'il n'existe toujours pas de standard [Ravat, 2007]. Au sein des systèmes d'aide à la décision, les concepts et les structures existent sans base théorique stable et standardisée [Niemi Ti. et al., 2003] et [Rizzi et al., 2006].

6.2.3 Analyse OLAP de documents

L'ajout de documents pour une analyse en ligne OLAP a un impact sur l'architecture. Dans notre approche, cet impact concerne deux niveaux précis : le magasin de données et la restitution/analyse. Dans ce cadre, ce mémoire de thèse s'articule autour de :

- la modélisation conceptuelle multidimensionnelle au sein des magasins de données,
- la manipulation OLAP des magasins pour permettre la restitution et l'analyse,
- les méthodes d'agrégation de données multidimensionnelles par des fonctions d'agrégations pour la synthèse de l'information dans le cadre de la restitution.

La modélisation multidimensionnelle conceptuelle au sein des magasins de données, permet de découvrir des sujets et des axes d'analyse. La manipulation des concepts du modèle via des opérations permet la spécification d'analyses. L'agrégation est un élément essentiel des systèmes d'aide à la décision. C'est par ce principe qu'un grand volume de données peut être réduit à quelques valeurs plus facilement appréhendables par un utilisateur.

Le but de cette thèse étant de fournir un environnement adapté, la première étape est de définir un modèle conceptuel permettant l'analyse de données issues de documents orientés documents.

Pour permettre le support de l'analyse de données issues de documents textuels (orientés documents), les opérations de manipulation nécessiteront d'être redéfinies. Ces opérations étant connues des décideurs, il est important que le principe et le fonctionnement des opérations ne diffèrent que peu des opérations de manipulation présentées précédemment. Toutefois, les opérations auront besoin d'être étendues pour prendre en compte les spécificités des documents pour permettre leur analyse.

L'essentiel des travaux d'analyse du contenu de documents actuels repose sur des analyses numériques faute de méthode d'agrégation adaptée pour des données textuelles. En effet, l'analyse de données textuelles nécessite une gestion particulière de la structure des documents. De plus, une telle analyse devrait reposer sur une mesure textuelle, inexistant dans les environnements actuels. L'analyse de données repose sur des processus de synthétisation des informations. Il s'agit de fonctions d'agrégation. Toutefois, ces fonctions bien adaptées à des données numériques ne peuvent gérer des données textuelles.

Références

[Abelló, 2002] Alberto Abelló, *YAM^2: a multidimensional conceptual model*, Thèse de doctorat, Université Polytechnique de Catalogne (Espagne), avril 2002.

[Abelló et al., 2003] Alberto Abelló, José Samos, Fèlix Saltor, "Implementing operations to navigate semantic star schemas", *6th ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 56–62, 2003.

[Abelló et al., 2006] Alberto Abelló, José Samos, Fèlix Saltor, "YAM^2: a multidimensional conceptual model extending UML", *Information Systems* (*IS*), vol.31(6), Elsevier, p. 541–567, septembre 2006.

[Abiteboul et al., 2002] Serge Abiteboul, Sophie Cluet, Guy Ferran, Marie-Christine Rousset, "The Xyleme project", *Computer Networks*, vol.39(3), Elsevier, p. 225–238, 2002.

[Abiteboul, 2003] Serge Abiteboul, "Managing an XML Warehouse in a P2P Context", *15th Intl. Conf. on Advanced Information Systems Engineering* (*CAiSE*), LNCS 2681, Springer, p. 4–13, 2003.

[Abiteboul, 2006] Serge Abiteboul, "Entrepôts de contenu autour de XML et des services Web", *2ème journées francophones sur les Entrepôts de Données et Analyse en ligne* (*EDA*), Revue des Nouvelles Technologies de l'Information (RNTI), numéro spécial, vol.RNTI-B-2, Cépaduès Editions, conference invite, p. 1, 2006.

[Agrawal et al., 1995] Rakesh Agrawal, Ashish Gupta, Sunita Sarawagi, *Modeling Multidimensional Databases*, IBM Research Report, http://rakesh.agrawal-family.com/papers/icde97olap_rj.pdf 1995.

[Agrawal et al., 1997] Rakesh Agrawal, Ashish Gupta, Sunita Sarawagi, "Modeling Multidimensional Databases", *13th Intl. Conf. on Data Engineering* (*ICDE*), IEEE Computer Society, p. 232–243, 1997.

[Agrawal et al., 2000] Rakesh Agrawal, Roberto J. Bayardo Jr., Ramakrishnan Srikant, "Athena: Mining-Based Interactive Management of Text Database", *7th Intl. Conf. on Extending Database Technology* (*EDBT*), LNCS 1777, Springer, p. 365–379, 2000.

[Amer-Yahia et al., 2002] Sihem Amer-Yahia, SungRan Cho, Divesh Srivastava, "Tree Pattern Relaxation", *Advances in Database Technology, 8th Intl. Conf. on Extending Database Technology* (*EDBT*), LNCS 2287, Springer, p. 496–513, 2002.

[Beyer et al., 2005] Kevin S. Beyer, Donald D. Chamberlin, Latha S. Colby, Fatma Özcan, Hamid Pirahesh, Yu Xu, "Extending XQuery for Analytics", *ACM SIGMOD Intl. Conf. on Management of Data* (*SIGMOD*), ACM Press, p. 503–514, 2005.

[Bordawekar & Lang, 2005] Rajesh Bordawekar, Christian A. Lang, "Analytical processing of XML documents: opportunities and challenges", *ACM SIGMOD Record*, vol.34(2), ACM Press, p. 27–32, mars 2005.

[Börzsönyi et al., 2001] Stephan Börzsönyi, Donald Kossmann, Konrad Stocker, "The Skyline Operator", *17th Intl. Conf. on Data Engineering* (*ICDE*), IEEE Computer Society, p. 421–430, 2001.

[Boussaid et al., 2006] Omar Boussaid, Riadh Ben Messaoud, Rémy Choquet, Stéphane Anthoard, "X-Warehousing: An XML-Based Approach for Warehousing Complex Data", *10th East European Conf. on Advances in Databases and Information Systems* (*ADBIS*) LNCS 4152, Springer, p. 39–54, 2006.

[Bruckner et al., 2001] Robert M. Bruckner, Beate List, Josef Schiefer, A. Min Tjoa, "Modeling Temporal Consistency in Data Warehouses", *1st Intl. Workshop on Knowledge Extraction for Enterprise Services* (KEES), *12th Intl. Workshop on Database and Expert Systems Applications* (*DEXA Workshop*), IEEE Computer Society, p. 901–905, 2001.

[Bültzingsloewen, 1987] Günter von Bültzingsloewen, "Translating and Optimizing SQL Queries Having Aggregates", *13th Intl. Conf. on Very Large Data Bases* (*VLDB*), Morgan Kaufmann, p. 235–243, 1987.

[Cabibbo & Torlone, 1997] Luca Cabibbo, Riccardo Torlone, "Querying Multidimensional Databases", *6th Intl. Workshop Database Programming Languages* (*DBPL*), LNCS 1369, Springer, p. 319–335, 1997.

[Cabibbo & Torlone, 1998] Luca Cabibbo, Riccardo Torlone, "From a Procedural to a Visual Query Language for OLAP", *10th Intl. Conf. on Scientific and Statistical Database Management* (*SSDBM*), IEEE Computer Society, p. 74–83, 1998.

[Cabibbo & Torlone, 2000] Luca Cabibbo, Riccardo Torlone "The Design and Development of a Logical System for OLAP", *2nd Intl. Conf. on Data Warehousing and Knowledge Discovery* (*DaWaK*), LNCS 1874, Springer, p. 1–10, 2000.

[Chakrabarti et al., 1998] Soumen Chakrabarti, Byron Dom, Rakesh Agrawal, Prabhakar Raghavan, "Scalable Feature Selection, Classification and Signature Generation for Organizing Large Text Databases into Hierarchical Topic Taxonomies", *The VLDB Journal*, vol.7(3), Springer, p. 163–178, août 1998.

[Codd, 1970] E.F. Codd, "A Relational Model of Data for Large Shared Data Banks", *Communications of the ACM*, vol.13(6), ACM Press, juin 1970.

[Codd, 1972] E. F. Codd, "Relational Completeness of Data Base Sublanguages", *Database Systems*, R. Rustin (ed.), Prentice Hall & IBM Research Report RJ 987, p. 65–98, 1972.

[Datta & Thomas, 1999] Anindya Datta, Helen Thomas, "The cube data model: a conceptual model and algebra for on-line analytical processing in data warehouses", *Decision Support Systems* (*DSS*), vol.27(3), Elsevier, p. 289–301, décembre 1999.

[Dublin Core, 2007] *The Dublin Core Metadata Initiative* de http://dublincore.org/ (Dublin Core Metadata Element Set, version 1.1) en date de mai 2007.

[Dudouet et al., 2005] François-Xavier Dudouet, Ioana Manolescu, Benjamin Nguyen, Pierre Senellart, "XML Warehousing Meets Sociology", *IADIS Intl. Conf. WWW/Internet* (*ICWI*), IADIS Press, Lisbonne, Portugal, Octobre 2005.

[Fankhauser & Klement, 2003] Peter Fankhauser, Thomas Klement, "XML for Data Warehousing Chances and Challenges" (Extended Abstract), *5th Intl. Conf. on Data Warehousing and Knowledge Discovery* (*DaWaK*), LNCS 2737, Springer, p.1-3, 2003.

[Franconi & Kamble, 2004] Enrico Franconi, Anand Kamble, "The GMD Data Model and Algebra for Multidimensional Information", *16th Intl. Conf. on Advanced Information Systems Engineering* (*CAiSE*), LNCS 3084, Springer, p. 446–462, 2004.

[Ghozzi, 2004] Faiza Ghozzi, *Conception et manipulation de bases de données dimensionnelles à contraintes*, Thèse de doctorat, Université Paul Sabatier, Toulouse 3 (France), novembre 2004.

[Golfarelli et al., 1998] Matteo Golfarelli, Dario Maio, Stefano Rizzi, "The Dimensional Fact Model: A Conceptual Model for Data Warehouses", invited paper, *Intl. Journal of Cooperative Information Systems* (*IJCIS*), vol.7(2-3), World Scientific Publishing, p. 215–247, juin & septembre1998.

[Golfarelli et al., 2001] Matteo Golfarelli, Stefano Rizzi, Boris Vrdoljak, "Data Warehouse Design from XML Sources", *4th ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 40–47, 2001.

[Golfarelli et al., 2002] Matteo Golfarelli, Stefano Rizzi, Ettore Saltarelli, "WAND: A CASE Tool for Workload-Based Design of a Data Mart", *Decimo Convegno Nazionale su Sistemi Evoluti per Basi di Dati* (*SEBD*), p. 422–426, 2002.

[Gray et al., 1996] Jim Gray, Adam Bosworth, Andrew Layman, Hamid Pirahesh, "Data Cube: A Relational Aggregation Operator Generalizing Group-By, Cross-Tab, and Sub-

Total", *12th Intl. Conf. on Data Engineering* (*ICDE*), IEEE Computer Society, p. 152–159, 1997.

[Gray et al., 1997] Jim Gray, Surajit Chaudhuri, Adam Bosworth, Andrew Layman, Don Reichart, Murali Venkatrao, Frank Pellow, Hamid Pirahesh, "Data Cube: A Relational Aggregation Operator Generalizing Group-by, Cross-Tab, and Sub Totals", *Data Mining and Knowledge Discovery* , vol.1(1), Springer, p. 29–53, mars 1997.

[Gyssens et al., 1996] Marc Gyssens, Laks V. S. Lakshmanan, Iyer N. Subramanian, "Tables as a Paradigm for Querying and Restructuring", *15th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems* (*PODS*), ACM Press, p. 93–103, 1996.

[Gyssens & Lakshmanan, 1997] Marc Gyssens, Laks V. S. Lakshmanan, "A Foundation for Multi-dimensional Databases", *23rd Intl. Conf. on Very Large Data Bases* (*VLDB*), Morgan Kaufmann, p. 106–115, 1997.

[Hahn et al., 2000] Karl Hahn, Carsten Sapia, Markus Blaschka, "Automatically Generating OLAP Schemata from Conceptual Graphical Models", *3rd ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 9–16, 2000.

[Harinarayan et al., 1996] Venky Harinarayan, Anand Rajaraman, Jeffrey D. Ullman, "Implementing Data Cubes Efficiently", *ACM Intl. Conf. on Management of Data* (*SIGMOD*), ACM Press, p. 205–216, 1996.

[Horner et al., 2004] John Horner, Il-Yeol Song, Peter P. Chen, "An analysis of additivity in OLAP systems", *7th ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 83–91, 2004.

[Huang & Su, 2002] Shi-Ming Huang, Chun-Hao Su, "The Development of an XML-Based Data Warehouse System", *3rd Intl. Conf. on Intelligent Data Engineering and Automated Learning* (*IDEAL*), LNCS 2412, Springer, p. 206–212, 2002.

[Hümmer et al., 2003] Wolfgang Hümmer, Andreas Bauer, Gunnar Harde, "XCube: XML for data warehouses", *6th ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 33–40, 2003.

[Jagadish et al., 1999] H. V. Jagadish, Laks V. S. Lakshmanan, Divesh Srivastava, "What can Hierarchies do for Data Warehouses?", *25th Intl. Conf. on Very Large Data Bases* (*VLDB*), Morgan Kaufmann, p. 530–541, 1999.

[Jensen et al., 2001] Mikael R. Jensen, Thomas H. Møller, Torben Bach Pedersen, "Specifying OLAP Cubes On XML Data", *13th Intl. Conf. on Scientific and Statistical Database Management* (*SSDBM*), IEEE Computer Society, p. 101–112, 2001.

[Johnson & Chatziantoniou, 1999] Theodore Johnson, Damianos Chatziantoniou, "Extending Complex Ad-Hoc OLAP", *8th Intl. Conf. on Information and Knowledge Management* (*CIKM*), ACM Press, p. 170–179, 1999.

[Keith et al., 2005] Steven Keith, Owen Kaser, Daniel Lemire, "Analyzing Large Collections of Electronic Text Using OLAP", *APICS 29th Conf. in Mathematics, Statistics and Computer Science*, Acadia University, p. 17–26, 2005.

[Kimball, 1996] Ralph Kimball, *The data warehouse toolkit: Practical Techniques for Building Dimensional Data Warehouses*, John Wiley and Sons, ISBN : 0-471-15337-0, 1996, 2ème ed. : Ralph Kimball, Margaery Ross, *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling*, *2nd Edition*, John Wiley & Sons, 2002.

[Klug, 1982] Anthony C. Klug, "Equivalence of Relational Algebra and Relational Calculus Query Languages Having Aggregate Functions", *Journal of the ACM* (*JACM*), vol.29(3), ACM Press, p. 699–717, juillet 1982.

[Khrouf & Soulé-Dupuy, 2004] Kaïs Khrouf, Chantal Soulé-Dupuy, "A Textual Warehouse Approach: A Web Data Repository", Chapitre de *Intelligent Agents for Data Mining and Information Retrieval*, Masoud Mohammadian (Ed.), Idea Publishing Group (IGP), ISBN : 1-59140-277-8, p. 101–124, 2004.

[Kung et al., 1975] H. T. Kung, E Luccio, E P. Preparata, "On finding the maxima of a set of vectors", *Journal of the ACM* (*JACM*), ACM Press, vol.22(4), p. 469–476, 1975.

[Lassila & McGuiness, 2001] Ora Lassila, Deborah L. McGuinness, "The Role of Frame-Based Representation on the Semantic Web", Knowl-edge Systems Laboratory Report KSL-01-02, Stanford University, 2001 (publié aussi dans *Computer and Information Science*, vol.6(5), Linköping University, 2001).

[Lenz & Shoshani, 1997] Hans-Joachim Lenz, Arie Shoshani, "Summarizability in OLAP and Statistical Data Bases", *9th Intl. Conf. on Scientific and Statistical Database Management* (*SSDBM*), IEEE Computer Society, p. 132–143, 1997.

[Lehner, 1998] Wolfgang Lehner, "Modelling Large Scale OLAP Scenarios", *6th Intl. Conf. on Extending Database Technology - Advances in Database Technology* (*EDBT*), LNCS 1377, Springer, p. 153–167, 1998.

[Li C. & Wang, 1996] Chang Li, Xiaoyang Sean Wang, "A Data Model for Supporting On-Line Analytical Processing", *5th Intl. Conf. on Information and Knowledge Management* (*CIKM*), ACM Press, p. 81–88, 1996.

[Li Y. & An, 2005] Yu Li, Aijun An, "Representing UML Snowflake Diagram from Integrating XML Data Using XML Schema", *Intl. Workshop on Data Engineering Issues in E-Commerce* (*DEEC*), IEEE Computer Society, p. 103–111, 2005.

[Luján-Mora et al., 2002] Sergio Luján-Mora, Juan Trujillo, Il-Yeol Song, "Extending the UML for Multidimensional Modeling", *5th Intl. Conf. on The Unified Modeling Language* (*UML*), LNCS 2460, Springer, p. 290–304, 2002.

[Luján-Mora, 2005] Sergio Luján-Mora, *Data Warehouse Design with UML*, Thèse de doctorat, Université d'Alicante (Espagne), juin 2005.

[Luján-Mora et al., 2006] Sergio Luján-Mora, Juan Trujillo, Il-Yeol Song, "A UML profile for multidimensional modeling in data warehouses", *Data & Knowledge Engineering* (*DKE*), vol.59(3), Elsevier, p. 725–769, décembre 2006.

[Malinowski & Zimányi, 2006] Elzbieta Malinowski, Esteban Zimányi, "Hierarchies in a multidimensional model: From conceptual modeling to logical representation", *Data & Knowledge Engineering* (*DKE*), Elsevier, vol.59(2), p. 348–377, November 2006.

[McCabe et al., 2000] Catherine McCabe, Jinho Lee, Abdur Chowdhury, David A. Grossman, Ophir Frieder, "On the design and evaluation of a multi-dimensional approach to information retrieval", *23rd Intl. ACM Conf. on Research and Development in Information Retrieval* (*SIGIR*), ACM Press, p. 363–365, 2000.

[Messaoud et al., 2004] Riadh Ben Messaoud, Omar Boussaid, Sabine Rabaséda, "A new OLAP aggregation based on the AHC technique", *7th ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 65–72, 2004.

[Messaoud, 2006] Riadh Ben Messaoud, *Couplage de l'analyse en ligne et de la fouille de données pour l'exploration, l'agrégation et l'explication des données complexes*, thèse de doctorat, Université Lumière Lyon 2 (France), novembre 2006.

[Mothe et al., 2003] Josiane Mothe, Claude Chrisment, Bernard Dousset, Joël Alau, "DocCube: Multi-dimensional visualisation and exploration of large document sets", *Journal of the American Society for Information Science and Technology* (*JASIST*), vol.54(7), Wiley Periodicals, p. 650–659, mai 2003.

[Nassis et al., 2004] Vicky Nassis, Rajagopal Rajugan, Tharam S. Dillon, J. Wenny Rahayu, "Conceptual Design of XML Document Warehouses", *6th Intl. Conf. on Data Warehousing and Knowledge Discovery* (*DaWaK*), LNCS 3181, Springer, p. 1–14, 2004.

[Nguyen et al., 2000] Thanh Binh Nguyen, A. Min Tjoa, Roland Wagner, "An Object Oriented Multidimensional Data Model for OLAP", *1st Intl. Conf. on Web-Age Information Management* (*WAIM*), LNCS 1846, Springer, p. 69-82, 2000.

[Nguyen et al., 2001] Thanh Binh Nguyen, A. Min Tjoa, Oscar Mangisengi, "Meta Cube-X: An XML Metadata Foundation for Interoperability Search among Web Data Warehouses", *3rd Intl. Workshop on Design and Management of Data Warehouses* (*DMDW*), CEUR Workshop Proceedings vol.39, CEUR-WS.org, p. 8.1–8.8, 2001.

[Nguyen et al., 2003] Thanh Binh Nguyen, A. Min Tjoa, Oscar Mangisengi, "MetaCube XTM: A Multidimensional Metadata Approach for Semantic Web Warehousing Systems", *5th Intl. Conf. Data Warehousing and Knowledge Discovery* (*DaWaK*), LNCS 27370 Springer, p. 76–88, 2003.

[Niemi Ta. et al., 2002] Tapio Niemi, Marko Niinimäki, Jyrki Nummenmaa, Peter Thanisch, "Constructing an OLAP cube from distributed XML data", *5th ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 22–27, 2002.

[Niemi Ti. et al., 2003] Timo Niemi, Lasse Hirvonen, Kalervo Järvelin: Multidimensional Data Model and Query Language for Informetrics. *Journal of the American Society for Information Science and Technology* (*JASIST*), vol.54(10), Wiley Periodicals, p. 939–951, mai 2003.

[Özsoyoglu et al., 1985] Gültekin Özsoyoglu, Zehra Meral Özsoyoglu, Francisco Mata, "A Language and a Physical Organization Technique for Summary Tables", *ACM Intl. Conf. on Management of Data* (*SIGMOD*), *ACM SIGMOD Record*, vol.14(4), ACM Press, p. 3–16, décembre 1985.

[Özsoyoglu et al., 1987] Gültekin Özsoyoglu, Zehra Meral Özsoyoglu, Victor Matos, "Extending Relational Algebra and Relational Calculus with Set-Valued Attributes and Aggregate Functions", *ACM Transactions on Database Systems* (*TODS*), vol.12(4), ACM Press, p. 566–592, 1987.

[Park et al., 2005] Byung-Kwon Park, Hyoil Han, Il-Yeol Song, "XML-OLAP: A Multidimensional Analysis Framework for XML Warehouses", *7th Intl. Conf. on Data Warehousing and Knowledge Discovery* (*DaWaK*), LNCS 3589, Springer, p. 32–42, 2005.

[Pedersen D. et al., 2002] Dennis Pedersen, Karsten Riis, Torben Bach Pedersen, "Query optimization for OLAP-XML federations", *5th ACM Intl. workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 57–64, 2002.

[Pedersen D. et al., 2004] Dennis Pedersen, Jesper Pedersen, Torben Bach Pedersen, "Integrating XML Data in the TARGITOLAP System", industrial paper, *20th Intl. Conf. on Data Engineering* (*ICDE*), IEEE Computer Society, p. 778–781, 2004.

[Pedersen T.B., 2000] Torben Bach Pedersen, *Aspects of Data Modeling and Query Processing for Complex Multidimensional Data*, Thèse de doctorat, Université d'Aalborg (Danemark), 2000.

[Pedersen T.B. et al., 2001] Torben Bach Pedersen, Christian S. Jensen, Curtis E. Dyreson, "A foundation for capturing and querying complex multidimensional data", *Information Systems* (*IS*), vol.26(5), Elsevier, p. 383–423, juillet 2001.

[Pérez et al., 2005] Juan Manuel Pérez, Rafael Berlanga Llavori, María José Aramburu, Torben Bach Pedersen, "A relevance-extended multi-dimensional model for a data warehouse contextualized with documents", *8th ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 19–28, 2005.

[Pérez et al., 2006] Juan Manuel Pérez-Martínez, Rafael Belanga, María José Aramburu, Torben Bach Pedersen, *Integrating Data Warehouses with Web Data : A Survey*, DB technical report, TR-18, http://www.cs.aau.dk/DBTR, 2006.

[Pérez et al., 2007] Juan Manuel Pérez-Martínez, Rafael Belanga-Llavori, María José Aramburu-Cabo, Torben Bach Pedersen, "Contextualizing data warehouses with documents", *Decision Support Systems* (*DSS*), Elsevier, (In Press) disponible en ligne depuis février 2007.

[Pokorný, 2001] Jaroslav Pokorný, "Modelling Stars Using XML", *4th ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 24–31, 2001.

[Pourrabas & Rafanelli, 2000] Elaheh Pourabbas, Maurizio Rafanelli, "Hierarchies and Relative Operators in the OLAP Environment", *ACM SIGMOD Record*, vol.29(1), ACM Press, p. 32–37, 2000.

[Pourrabas & Rafanelli, 2003] Elaheh Pourabbas, Maurizio Rafanelli, "Hierarchies", Chapitre IV, *Multidimensional Databases: Problems and Solutions*, Maurizio Rafanelli (Ed.), Idea Publishing Group (IGP), ISBN 1-59140-053-8, p. 91–115, 2003.

[Rafanelli, 2003] Maurizio Rafanelli, "Operators for Multidimensional Aggregate Data", Chapitre V, *Multidimensional Databases: Problems and Solutions*, Maurizio Rafanelli (Ed.), Idea Publishing Group (IGP), ISBN 1-59140-053-8, p. 116–165, 2003.

[Rajugan et al., 2003] Rajagopal Rajugan, Elizabeth Chang, Tharam S. Dillon, Ling Feng, "XML Views: Part 1", *14th Intl. Conf. on Database and Expert Systems Applications* (*DEXA*), LNCS 2736, Springer, p. 148–159, 2003.

[Ravat et al., 2006] Franck Ravat, Olivier Teste, Gilles Zurfluh, "Algèbre OLAP et langage graphique", *Actes du XXIVème Congrès INFormatique des ORganisations et Systèmes d'Information et de Décision* (*INFORSID*), Inforsid (Ed.), ISBN 2-906855-22-7, p. 1039–1054, 2006.

[Ravat et al., 2007e] Franck Ravat, Olivier Teste, Ronan Tournier, Gilles Zurfluh, "Algebraic and graphic languages for OLAP manipulations", *Intl. Journal of Data Warehousing and Mining* (*ijDWM*), Idea Group Publishing (IGP), juin 2007 (à paraître).

[Ravat, 2007] Franck Ravat, *Outils pour la conception et la manipulation de systèmes d'aide à la décision*, habilitation à diriger les recherches (HDR), Université de Toulouse 1 (France), à paraître, 2007.

[Rivest et al., 2005] Sonia Rivest, Yvan Bédard, Marie-Josée Proulx, Martin Nadeau, Frederic Hubert, Julien Pastor, "SOLAP technology: Merging business intelligence with geospatial technology for interactive spatio-temporal exploration and analysis of data", *ISPRS Journal of Photogrammetry and Remote Sensing* (*JPRS*), Elsevier, vol.60(1), p. 17–33, December 2005.

[Rizzi et al., 2006] Stefano Rizzi, Alberto Abelló, Jens Lechtenbörger, Juan Trujillo, "Research in data warehouse modeling and design: dead or alive?", *9th ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 3–10, 2006.

[Rusu et al., 2004] Laura Irina Rusu, J. Wenny Rahayu, David Taniar, "On Building XML Data Warehouses", *5th Intl. Conf. on Intelligent Data Engineering and Automated Learning* (*IDEAL*), LNCS 3177, Springer, p. 293–299, 2004.

[Sapia et al., 1998] Carsten Sapia, Markus Blaschka, Gabriele Höfling, Barbara Dinter, "Extending the E/R Model for the Multidimensional Paradigm", *Advances in Database Technologies, ER '98 Workshops on Data Warehousing and Data Mining, Mobile Data Access, and Collaborative Work Support and Spatio-Temporal Data Management* (*ER Workshops*), LNCS 1552, Springer, p. 105–116, 1998.

[Schneider, 2003] Michel Schneider, "Well-formed data warehouse structures", *5th Intl. Workshop on Design and Management of Data Warehouses* (*DMDW*), CEUR Workshop Proceedings vol.77, CEUR-WS.org, p. 2.1–2.13, 2003.

[Schneider, 2007] Michel Schneider, "A general model for the design of data warehouses", *Intl. Journal of Production Economics*, Elsevier, disponible en ligne 2007 (à paraître).

[Shoshani, 2003] Arie Shoshani, "Multidimensionality in Statistical, OLAP, and Scientific Databases. Multidimensional Databases", Chapitre II, *Multidimensional Databases: Problems and Solutions*, Maurizio Rafanelli (Ed.), Idea Publishing Group (IGP), ISBN 1-59140-053-8, p. 46–68, 2003.

[Stefanovic et al., 2000] Nebojsa Stefanovic , Jiawei Han , Krzysztof Koperski, "Object-Based Selective Materialization for Efficient Implementation of Spatial Data Cubes", *IEEE Transactions on Knowledge and Data Engineering* (*TKDE*), IEEE Computer Society, vol.12(6), p. 938–958, november 2000.

[Sullivan, 2001] Dan Sullivan, *Document Warehousing and Text Mining*, Wiley John & Sons, ISBN: 0471399590, 2001.

[Torlone, 2003] Riccardo Torlone, "Conceptual Multidimensional Models", Chapitre III, *Multidimensional Databases: Problems and Solutions*, Maurizio Rafanelli (Ed.), Idea Publishing Group (IGP), ISBN 1-59140-053-8, p. 69–90, 2003.

[Tournier, 2004] Ronan Tournier, *Bases de données multidimensionnelles : étude et implantation d'un langage graphique*, Rapport de Master de Recherche, IRIT, juin 2004.

[Tourwé et al., 2003] Tom Tourwé, Luk Stoops, Stijn Decneut, "Automated support for data exchange via XML", *5th Intl. Symposium on Multimedia Software Engineering* (*ISMSE*), IEEE Computer Society, p. 70–77, 2003.

[Tryfona et al., 1999] Nectaria Tryfona, Frank Busborg, Jens G. Borch Christiansen, "starER: A Conceptual Model for Data Warehouse Design", *2nd ACM Intl. Workshop on Data Warehousing and OLAP* (DOLAP), ACM Press, p. 3–8, 1999.

[Tseng & Chou, 2006] Frank S.C. Tseng, Annie Y.H. Chou, "The concept of document warehousing for multi-dimensional modeling of textual-based business intelligence",

journal of Decision Support Systems (*DSS*), vol.42(2), Elsevier, p. 727–744, novembre 2006.

[Vrdoljak et al., 2003] Boris Vrdoljak, Marko Banek, Stefano Rizzi, "Designing Web Warehouses from XML Schemas", *5th Intl. Conf. on Data Warehousing and Knowledge Discovery* (*DaWaK*), LNCS 2737, Springer, p. 89–98, 2003.

[Vrdoljak et al., 2006] Boris Vrdoljak, Marko Banek, Zoran Skocir, "Integrating XML Sources into a Data Warehouse", *2nd Intl. Workshop on Data Engineering Issues in E-Commerce and Services* (*DEECS*), LNCS 4055, Springer, p. 133–142, 2006.

[Wang et al., 2003] Hongzhi Wang, Jianzhong Li, Zhenying He, Hong Gao, "Xaggregation: Flexible Aggregation of XML Data", *4th Intl. Conf. on Advances in Web-Age Information Management* (*WAIM*), LNCS 2762, Springer, p. 104–115, 2003.

[Wang et al., 2005] Hongzhi Wang, Jianzhong Li, Zhenying He, Hong Gao, "OLAP for XML Data", *5th Intl. Conf. on Computer and Information Technology* (*CIT*), IEEE Computer Society, p. 233–237, 2005.

[Wiwatwattana et al., 2007] Nuwee Wiwatwattana, H. V. Jagadish, Laks V. S. Lakshmanan, Divesh Srivastava "X^3: A Cube Operator for XML OLAP", *23rd Intl. Conf. on Data Engineering* (*ICDE*), IEEE Computer Society, p. 916–925, 2007.

[Yin & Pedersen T.B., 2004] Xuepeng Yin, Torben Bach Pedersen, "Evaluating XML-extended OLAP queries based on a physical algebra", *7th ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 73–82, 2004.

[Zhang et al., 2003] Ji Zhang, Tok Wang Ling, Robert M. Bruckner, A. Min Tjoa, "Building XML Data Warehouse Based on Frequent Patterns in User Queries", *5th Intl. Conf. on Data Warehousing and Knowledge Discovery* (*DaWaK*), LNCS 2737, Springer, p. 99–108, 2003.

CHAPITRE III
Modèle conceptuel multidimensionnel en galaxie

Résumé du chapitre

Ce chapitre présente un modèle conceptuel multidimensionnel permettant l'analyse de données issues de documents. Après un rappel des objectifs du modèle, le modèle en galaxie est présenté avec son unique concept central de dimension. Afin de prendre en considération des propriétés de documents, le chapitre se poursuit sur la présentation des liens inter et intra-dimension qui permettent d'exploiter les liens présents dans les documents sources. Toujours selon le même objectif, la gestion des données textuelles issues de documents est présentée via le concept d'attribut et de dimension documentaire. Enfin un parallèle est présenté entre les modélisations conceptuelles multidimensionnelles en galaxie et celle basée sur une dualité fait-dimension.

Sommaire

CHAPITRE III : Modèle conceptuel multidimensionnel en galaxie

« Savoir faire abstraction d'une représentation, même si elle vient s'imposer à l'homme par l'intermédiaire des sens, est un pouvoir beaucoup plus grand que celui d'être attentif [...] »

—Emmanuel Kant, Anthropologie.

1 Introduction

La modélisation conceptuelle multidimensionnelle vise à représenter des besoins utilisateurs en terme d'analyse. Cette modélisation, indépendante de toute contrainte d'implantation logique et physique, permet d'obtenir une vision orientée décideur [Golfarelli et al., 2002] et facilite la compréhension de l'ensemble des données mises à disposition de l'analyste [Rizzi et al., 2006].

Notre but est de permettre l'analyse en ligne (OLAP) de documents XML. Afin de refléter les besoins décisionnels, nous proposons de représenter ces besoins par un modèle permettant l'analyse du contenu des documents. Notre modèle doit être assez expressif pour représenter les caractéristiques des documents, tout en facilitant les prises de décisions.

1.1 Problématique liée à l'analyse de documents XML

Les documents XML disposent de caractéristiques que nous voulons prendre en compte dans notre modèle. Les quatre principales caractéristiques des documents que nous proposons de modéliser sont les suivantes :

- la structuration hiérarchique des données textuelles des documents,
- la représentation des liens intra ou inter-documents,
- l'intégralité des données textuelles qui constituent les documents,
- les méta-données associées à un document.

Premièrement, les données textuelles sont organisées de manière hiérarchique (par exemple, des chapitres contenant des sections, elles-mêmes composées de paragraphes). Cette structure représente la disposition classique des informations d'un document.

Deuxièmement, au sein des documents, des éléments peuvent être interconnectés : les documents disposent de liens intra-document ou inter-documents qui sont réprésentés notamment par des liens hypertextes. Par exemple dans ce mémoire de thèse, il est possible de rencontrer régulièrement la citation de références. Ces interconnections entre éléments représentent, non seulement un complément d'information, mais aussi un moyen d'analyse complémentaire.

Troisièmement, l'analyse OLAP classique repose sur des indicateurs numériques, tels qu'un nombre de documents. Nous souhaitons aller au-delà des capacités des environnements traditionnels et permettre une analyse du contenu des documents. Or ce contenu est principalement constitué de données textuelles.

Enfin quatrièmement, nous souhaitons pouvoir représenter les méta-données associées aux documents XML (tel que les auteurs ou encore la date de publication d'un document). Ces méta-données constituent des informations complémentaires au contenu d'un document.

1.2 Limites des modèles actuels

Pour les analyses OLAP, les modèles multidimensionnels reposant sur la dualité faits-dimensions sont reconnus. Toutefois, au regard de nos besoins, ces modèles présentent un certain nombre d'insuffisances. Nous en avons relevé trois, détaillées dans les sections suivantes.

1.2.1 Non analyse du contenu de documents

La modélisation reposant sur les concepts de fait et de dimension associés à des indicateurs numériques permet des analyses simples de documents XML. Ces analyses reposent principalement sur le comptage de documents.

Exemple. Dans l'exemple suivant (cf. Figure 24), un décideur observe les activités de recherche d'un institut (*Inst1*) et il analyse les citations d'articles scientifiques. L'analyse consiste à compter chaque fois qu'un auteur de l'institut *Inst1* est cité par un article.

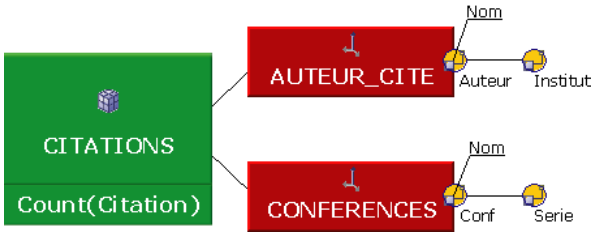


Figure 24 – Représentation multidimensionnelle d'une analyse de citations d'auteurs.

Ainsi, dans le Tableau 5, l'auteur, *Au1*, a été cité 3 fois par des articles de *DaWaK*, 2 fois par des articles de *DEXA*,… De son côté, l'auteur *Au3* a été cité une seule fois dans les articles de *DaWaK* et 2 fois par des articles de *CAiSE*.

Tableau 5 – Exemple d'analyse numérique.

ARTICLES COUNT (Document)		AUTEURS_CITÉS		
	Institut	Inst1		
	Auteur	Au1	Au2	Au3
CONFERENCES	Nom			
	DaWaK	3	2	1
	DEXA	2	-	-
	CAiSE	1	1	2

Nous proposons d'étendre les capacités d'analyse offertes aux décideurs en permettant, par exemple, d'analyser le contexte de ces citations, à savoir, de quel sujet traitent les articles qui citent les auteurs analysés. Ainsi il est possible de savoir quels sont les domaines influencés par les chercheurs de l'institut *Inst1*. Comme cette analyse ne repose plus sur des données numériques, la fonction TOP_KEYWORDS [Park et al., 2005] est employée pour afficher les deux principaux mots-clefs des documents. Ces mots-clefs sont agrégés à partir du contenu

des documents. Ils seront regroupés par conférence et donnent une liste de sujets plutôt qu'un nombre d'articles.

Exemple. Le Tableau 6 représente la même analyse que précédemment, seulement cette fois les principaux mots-clefs des articles sont présentés au lieu de compter le nombre d'articles. Ainsi, il est possible de constater que les trois citations de l'auteur *Au1* l'ont été dans le cadre de *XML* et de *Documents*. L'auteur *Au3* a toujours été cité dans le cadre de *Fouille de données*. Par rapport aux autres citations de *Au1*, les travaux de *Au3* semblent avoir une moindre portée, en terme de domaines, dans les trois conférences présentées.

Tableau 6 – Exemple d'analyse textuelle.

ARTICLES TOP_KEYWORDS (Document)		AUTEURS_CITÉS			
		Institut	Inst1		
		Auteur	Au1	Au2	Au3
CONFERENCES	Nom				
	DaWaK		XML, Docments	Fouille de données, Clustering	Fouille de données
	DEXA		XML, BD temporelles	-	-
	CAiSE		Fouille de données	XML, Entrepôts de données	Fouilles de données

Cette analyse exploite à la fois des indicateurs textuels et les liens entre les données analysées (en l'occurrence des citations au sein d'articles scientifiques). Elle serait extrêmement difficile, voire impossible à spécifier avec une modélisation classique. Ainsi les modèles actuels ne suffisent pas pour permettre l'analyse du contenu de documents XML.

1.2.2 Analyses prédéfinies et structures non flexibles

Un fait représente un sujet d'analyse prédéfini. La définition d'un fait rend la spécification d'analyses peu flexible, car le décideur se voit contraint d'employer ces faits comme sujets.

Dans une modélisation multidimensionnelle classique, les données sont représentées par des points dans un espace multidimensionnel. L'espace multidimensionnel est à la fois composé des axes d'analyses issus des dimensions et d'axes issus des mesures (les indicateurs des sujets d'analyse). Conséquence de cette optique de modélisation, le décideur dispose d'une plus grande flexibilité quant à la spécification d'analyses, car ce type de modèle n'impose pas au décideur des solutions d'analyse spécifiées à priori, qui pourraient s'avérer limitatives.

Exemple. En reprenant l'exemple d'analyse de citations, considérons l'analyse du nombre de citations en fonction des séries de conférences et des auteurs cités (cf. Figure 25). Dans cette analyse, les citations (nombre de citations) sont représentées par un axe d'analyse (disposé en profondeur).

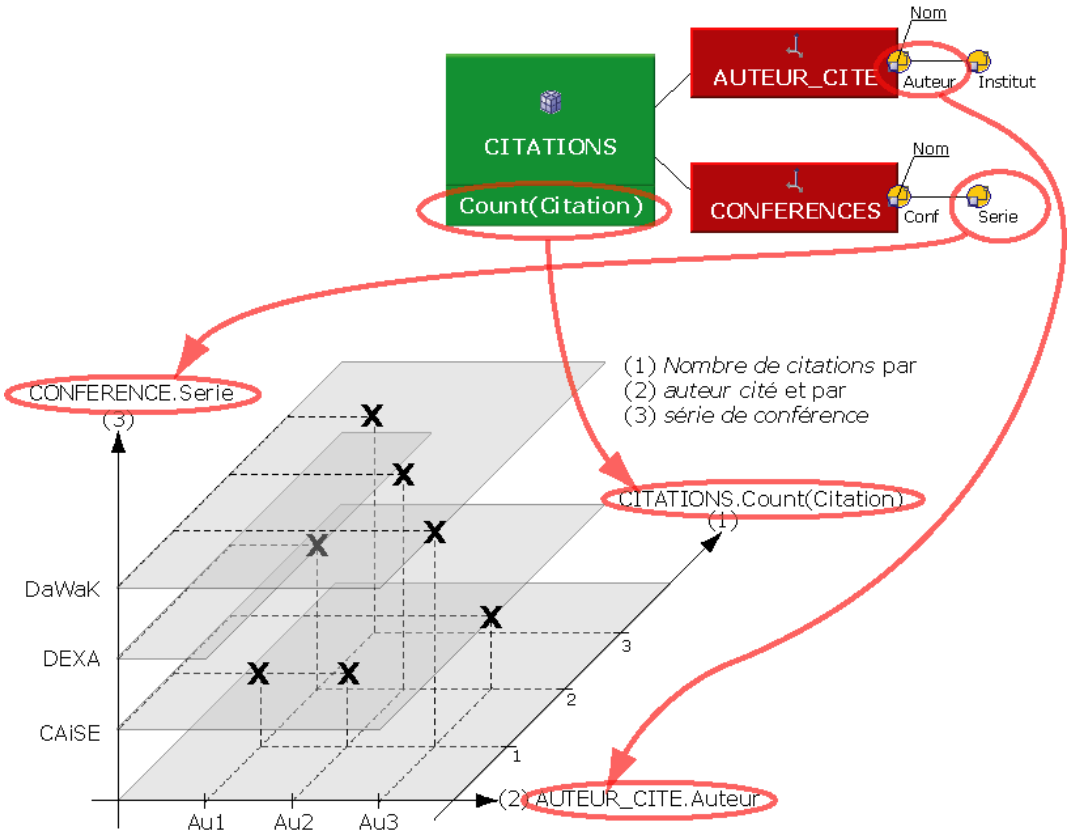


Figure 25 – Représentation d'une mesure par un axe.

1.2.3 Difficultés pour repérer les sujets d'analyse

"*There is no formal way of deciding which attributes should be made dimensions and which attributes should be made measures. It is left as a database design decision.*" [Agrawal et al., 1997]. Dix années se sont écoulées depuis cette affirmation, mais elle reste d'actualité, malgré l'existence de plusieurs méthodes comme [Kimball, 1996] ou encore [Ghozzi, 2004]. A notre connaissance, aucune n'intègre une méthode fiable pour la désignation systématique des données factuelles. La désignation des données factuelles est un élément clé de la modélisation conceptuelle mais aussi l'un des plus difficile.

En modélisant l'ensemble des structures multidimensionnelles par des axes d'analyse, non seulement, nous proposons une solution en terme de flexibilité pour l'analyse de documents mais aussi une solution à l'épineux problème de désignation des données factuelles.

1.3 Objectif et organisation du chapitre

Notre objectif est de proposer un modèle permettant une représentation des collections de documents XML facilitant les analyses OLAP. Par *collection* nous entendons un ensemble de documents homogènes en structure. Les documents sont supposés être structurés ou semi-structurés tels que des articles scientifiques au format XML.

Premièrement, nous souhaitons que notre modèle prenne en compte les caractéristiques des collections de documents XML. Ainsi les analyses OLAP exprimées à partir de notre modèle pourront prendre en considération la hiérarchisation des données, les liens au sein des documents d'une collection, leur contenu et les méta-données associées.

Deuxièmement, le modèle doit être suffisamment générique pour pouvoir gérer de manière uniforme les indicateurs numériques et les indicateurs textuels. Ces derniers sont au cœur de l'analyse de données issues de documents XML principalement constitués de données textuelles.

Enfin, troisièmement, le modèle doit fournir au décideur une flexibilité quant à la désignation des sujets lors de la spécification d'analyses multidimensionnelles. Nous désirons qu'il ne soit pas contraint par des solutions d'analyse spécifiées a priori.

Au vu des limites existantes et de nos objectifs, nous proposons de généraliser un modèle en constellation [Kimball, 1996] défini dans [Ravat et al., 2007e]. Le modèle que nous définissons emploie la notion de dimension pour modéliser les axes d'analyse. Ces dimensions qui hiérarchisent des paramètres d'analyse permettant de prendre en compte la structure hiérarchique des documents. Enfin, nous proposons de simplifier et de généraliser la représentation des éléments conceptuels multidimensionnels et de représenter un sujet d'analyse comme un axe d'analyse. Associé à des opérateurs adaptés, ceci permettra une symétrie complète entre les attributs du modèle [Agrawal et al., 1997]. Ceci a pour conséquence d'octroyer à notre modèle une grande flexibilité quant à la spécification d'analyse de la part de l'utilisateur.

Plan du chapitre. Au vu des limites existantes et de nos objectifs, nous proposons d'étendre un modèle en constellation [Kimball, 1996] défini dans [Ravat et al., 2007e]. L'extension du modèle repose sur une conception en galaxie (section 2) qui modélise l'ensemble des éléments multidimensionnels par un unique concept de dimension, n'imposant pas de sujets d'analyse définis à priori. La notion de dimension (section 3) modélise les axes d'analyse. Ces dimensions qui hiérarchisent des paramètres d'analyse permettant de prendre en compte la structure hiérarchique des documents. La galaxie modélise les liens intra et inter-documents (section 4) tout en permettant leur exploitation dans une analyse (cf. chapitre 4 sur la manipulation multidimensionnelle). Enfin, les spécificités inhérentes à la gestion d'indicateurs textuels sont prises en compte (section 5). Toutefois, le modèle n'est pas restrictif et il est toujours possible d'employer des indicateurs numériques ou bien de modéliser des besoins d'analyse classiques tels que ceux représentés par un schéma en constellation. Ce chapitre se termine donc par un parallèle entre une modélisation en constellation [Ravat et al., 2007e] et une modélisation en galaxie.

2 Galaxie

Nous généralisons le concept de constellation [Kimball, 1996] en définissant celui de galaxie. Notre approche consiste à décrire un schéma multidimensionnel par l'unique concept de dimension ; la notion de fait est supprimée. Une galaxie est un regroupement de dimensions liées entre elles par un ou plusieurs nœuds centraux ; chaque nœud modélise les dimensions compatibles pour une même analyse. De plus, les éléments constituant les dimensions peuvent être interconnectés au travers de liens.

Définition. Une *Galaxie G* est définie par (D^G, $Star^G$, Lk^G) :

- $D^G = \{D_1,\ldots, D_n\}$ est un ensemble de *dimensions*,
- $Star^G : D^G \rightarrow 2^{D^G}$ est une fonction associant chaque dimension $D_i \in D^G$ à l'ensemble des dimensions $D_{j_x \neq i} \in D^G$ compatibles pour une analyse (cf. Eq. 1)[11].

$$Star^G : \begin{array}{ccc} D^G & \longrightarrow & 2^{D^G} \\ D_i & \mapsto & \{D_{j_1},\ldots,D_{j_n}\}, D_{j_x \neq i} \end{array} \qquad \textbf{Eq. 1}$$

- $Lk^G = \{l_1,\ldots,l_u\}$ est un ensemble de *liens* qui représentent les liens intra ou inter-documents (ces liens seront définis en section 4).

Notation. la notation $D_j \in Star^G(D_i)$ représente l'association de la dimension D_i avec la dimension D_j via la fonction $Star^G$.

Formalisme graphique. Une galaxie est représentée graphiquement par un ensemble de rectangles qui représentent les dimensions reliées entre elles par des liens modélisés par la fonction $Star^G$ (cf. légende de la Figure 26).

Exemple. Afin d'observer les activités d'instituts de recherches, un décideur analyse des publications scientifiques et des rapports de projets. Ces analyses correspondent à la galaxie G_1 qui représente :

- dans sa partie supérieure : les articles publiés au sein d'une conférence à une date donnée et écrits par des auteurs ;
- dans la partie inférieure : les rapports de projets obtenus à une certaine date, pilotés par des instituts et employant des personnels scientifiques (qui sont aussi des auteurs d'articles).

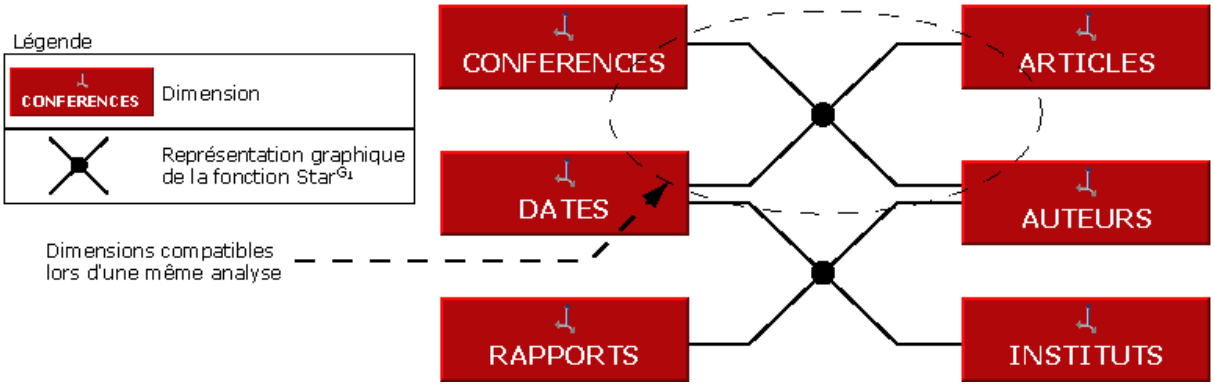


Figure 26 – Exemple simplifié de galaxie G_1.

Une représentation plus complète (avec les dimensions et hiérarchies déployées) de la galaxie est présentée en Figure 33. Suivant les spécifications formelles, la galaxie G_1 est définie comme suit :

$G_1 = (D^{G1}, Star^{G1}, Lk^{G1})$

- $D^{G1} = \{D_{CONFERENCES}, D_{ARTICLES}, D_{DATES}, D_{AUTEURS}, D_{RAPPORTS}, D_{INSTITUTS}\}$
- $Star^{G1} = \{$

[11] La notation 2^E représente la notation exponentielle de $P(E)$, l'ensemble des parties d'ensemble de l'ensemble E ; si $E=\{e_1, e_2\}$ alors $2^E = P(E) = \{\{\}, \{e_1\}, \{e_2\}, \{e_1, e_2\}\}$

$D_{CONFERENCES} \rightarrow \{D_{ARTICLES}, D_{DATES}, D_{AUTEURS}\}$,
$D_{ARTICLES} \rightarrow \{D_{CONFERENCES}, D_{DATES}, D_{AUTEURS}\}$,
$D_{DATES} \rightarrow \{D_{CONFERENCES}, D_{ARTICLES}, D_{AUTEURS}, D_{RAPPORTS}, D_{INSTITUTS}\}$,
$D_{AUTEURS} \rightarrow \{D_{CONFERENCES}, D_{ARTICLES}, D_{DATES}, D_{RAPPORTS}, D_{INSTITUTS}\}$,
$D_{RAPPORTS} \rightarrow \{D_{DATES}, D_{AUTEURS}, D_{INSTITUTS}\}$,
$D_{INSTITUTS} \rightarrow \{D_{DATES}, D_{AUTEURS}, D_{RAPPORTS}\}\}$

- La galaxie comporte aussi deux liens qui seront détaillés en section 4 : $Lk^{G1} = \{l_1, l_2\}$

L'emploi de nœuds centraux (des cercles au centre des liaisons entre les dimensions dans la Figure 26) permet une représentation graphique synthétique de $Star^G$.

La fonction $Star^G$ induit des ensembles de dimensions $Star^G(D_i) \subseteq D^G$. On pose :

- $D^{c_k} = \{D_i \mid \exists D_{j \neq i}, D_j \in Star^G(D_i) \wedge D_i \in Star^G(D_j)\}$ un ensemble de dimensions reliées entre elles deux à deux. D^{c_k} représente un sous graphe complet (ou clique[12]) du graphe définit par $Star^G$.
- D^c l'ensemble des sous ensembles D^{ck} obtenus à partir de $Star^G$, soit l'ensemble des sous-graphes complets (cliques) définis par $Star^G$.

Exemple. Nous poursuivons l'exemple précédent. A partir de la fonction $Star^{G1}$ nous obtenons $D^c = \{D^{c_1}, D^{c_2}\}$ avec :

- $D^{c_1} = D^{c_{PUBLICATIONS}} = \{D_{ARTICLES}, D_{CONFERENCES}, D_{DATES}, D_{AUTEURS}\}$
- $D^{c_2} = D^{c_{PROJETS}} = \{D_{DATES}, D_{AUTEURS}, D_{RAPPORTS}, D_{INSTITUTS}\}$.

Remarque. Nous distinguons les notions de dimensions partagées et de dimensions non partagées.

- Une *dimension non partagée* D_i est une dimension n'apparaissant que dans une seule clique D^{ck} c'est-à-dire, si $D_i \in D^{ck}$ alors $\nexists D^{cl} \in D^c \mid D_i \in D^{cl}$ $(l \neq k)$.
- Les *dimensions partagées* D_i sont des dimensions apparaissant dans au moins deux cliques D^{ck} et D^{cl}, c'est-à-dire, si $D_i \in D^{ck}$ alors $\exists D^{cl} \in D^c \mid D_i \in D^{cl}$ $(l \neq k)$.

Exemple. Par exemple, la dimension *CONFERENCES* n'appartient qu'à un seul groupe de dimensions ($D^{c_{PUBLICATIONS}}$) : elle est donc une dimension non partagée. D'un autre côté, la dimension *AUTEURS* appartient à deux groupes ($D^{c_{PUBLICATIONS}}$ et $D^{c_{PROJETS}}$), il s'agit d'une dimension partagée.

Un exemple plus détaillé est proposé dans la section suivante qui présente plus en détails les dimensions.

3 Dimensions et hiérarchies

3.1 Concept de dimension

Une dimension est caractérisée par des attributs organisés de manière hiérarchique. Chaque attribut modélise un niveau de graduation de l'axe d'analyse (ou niveau granularité). Les dimensions pouvant être considérées comme des axes d'analyse, les attributs sont autant d'indicateurs d'analyse potentiels.

[12] Dans un graphe, une *clique* est un sous-graphe complet dont tous les sommets sont reliés deux à deux par une arrête [Bondy & Murty, 1976].

On pose *ID* un ensemble d'identifiants uniques.

Définition. Une *dimension* D_i est définie par (A^{Di}, H^{Di}, I^{Di}, $IStar^{Di}$) :

- $A^{Di} = \{a^{Di}_1,\ldots, a^{Di}_r\} \cup \{Id, All\}$ est un ensemble d'*attributs*, les attributs *Id* et *All* étant précisés dans la définition des hiérarchies,
- $H^{Di} = \{H^{Di}_1,\ldots, H^{Di}_s\}$ est un ensemble de *hiérarchies*,
- $I^{Di} = \{i^{Di}_1,\ldots, i^{Di}_t\}$ est un ensemble d'*instances de dimension*. On note $\forall k \in [1..t]$ $i^{Di}_k = [Id : id_{x_k} ; a^{Di}_1 : v_{k1} ; \ldots ; a^{Di}_r : v_{kr} ; All : all]$, $id_{x_k} \in ID$, $\forall j \in [1..r]$, $v_{kj} \in \text{dom}(a^{Di}_j)$.[13]
- $\forall D^{ck} \in D^C | D_i \in D^{ck}, IStar^{D_i}_{D^{ck}} : I^{D_i} \longrightarrow 2^{\mathfrak{I}}$ où $\mathfrak{I} = \prod_{\forall D_j \in D^{ck} - \{D_i\}} I^{D_j}$ représente des fonctions associant les instances de la dimension D_i aux instances des autres dimensions auxquelles la dimension est liée, pour chacune des cliques D^{ck} auquelle elle appartient. Nous notons $IStar^{Di}$ l'ensemble de ces fonctions pour la dimension D_i.

Notations. Les notations formelles complètes a^{Di}_j, H^{Di}_k, i^{Di}_l,… seront simplifiée lors de l'absence d'ambiguïtés sur la dimension (en l'occurrence D_i) par les notations : a, H, i_x,…

Remarque. Le contenu de $IStar^{Di}$ dépend du type de dimension. $IStar^{Di}$ est un singleton, si D_i est une dimension non-partagée. Si D_i est une dimension partagée, alors, $IStar^{Di}$ est composée d'autant de fonctions que D_i appartient à des cliques différentes.

Exemple (**dimension non partagée**). Nous poursuivons l'exemple précédent en considérant la dimension non partagée *CONFERENCES*, $D_{CONFERENCES} \in D^{c_{PUBLICATIONS}}$.

$D_{CONFERENCES} = (A^{D_{CONFERENCES}}, H^{D_{CONFERENCES}}, I^{D_{CONFERENCES}}, IStar^{D_{CONFERENCES}})$

- $A^{D_{CONFERENCES}} = \{Conf, Audience, Editeur, Nom, Serie, Tx\_Acceptation, All\}$ où *Conf* est l'attribut identifiant ($Conf \in ID$) ;
- $H^{D_{CONFERENCES}} = \{HConf\}$;
- $I^{D_{CONFERENCES}} = \{i^{D_{CONFERENCES}}_1, i^{D_{CONFERENCES}}_2, i^{D_{CONFERENCES}}_3\}$ avec

$i^{D_{CONFERENCES}}_1 = [Conf : c1;$
$Audience : National ;$
$Editeur : Inforsid ;$
$Nom ; Inforsid'07 ;$
$Serie : Inforsid ;$
$Tx\_Acceptation : 33 ;$
$All : all] ;$

$i^{D_{CONFERENCES}}_2 = [Conf : c2;$
$Audience : International ;$
$Editeur : Springer ;$
$Nom ; ER'07 ;$
$Serie : ER ;$
$Tx\_Acceptation : 22 ;$
$All : all] ;$

$i^{D_{CONFERENCES}}_3 = [Conf : c3;$
$Audience : International ;$
$Editeur : Springer ;$
$Nom ; DaWaK'06 ;$
$Serie : DaWaK ;$

[13] Nous notons $dom(a_i)$ l'ensemble des valeurs possibles v_k de l'attribut a_i.

$Tx\_Acceptation : 36$;
$All : all]$;

- $IStar^{D_{CONFERENCES}} = \{IStar_{D^{c_1}}^{D_{CONFERENCES}}\}$ où :

$IStar_{D^{c_1}}^{D_{CONFERENCES}} : I^{D_{CONFERENCES}} \rightarrow 2^{I^{D_{AUTEURS}} \times I^{D_{DATES}} \times I^{D_{ARTICLES}}}$ avec :

$IStar_{D^{c_1}}^{D_{CONFERENCES}}(i^{D_{CONFERENCES}}{}_1) = \{$
$(i^{D_{AUTEURS}}{}_1 ; i^{D_{DATES}}{}_1 ; i^{D_{ARTICLES}}{}_1) \}$
$IStar_{D^{c_1}}^{D_{CONFERENCES}}(i^{D_{CONFERENCES}}{}_2) = \{$
$(i^{D_{AUTEURS}}{}_1 ; i^{D_{DATES}}{}_2 ; i^{D_{ARTICLES}}{}_2) ;$
$(i^{D_{AUTEURS}}{}_2 ; i^{D_{DATES}}{}_2 ; i^{D_{ARTICLES}}{}_2) \}$
$IStar_{D^{c_1}}^{D_{CONFERENCES}}(i^{D_{CONFERENCES}}{}_3) = \{$
$(i^{D_{AUTEURS}}{}_1 ; i^{D_{DATES}}{}_3 ; i^{D_{ARTICLES}}{}_3) ;$
$(i^{D_{AUTEURS}}{}_3 ; i^{D_{DATES}}{}_3 ; i^{D_{ARTICLES}}{}_4) ;$
$(i^{D_{AUTEURS}}{}_1 ; i^{D_{DATES}}{}_3 ; i^{D_{ARTICLES}}{}_4) \}$

La Figure 27 représente graphiquement les liaisons entre les instances définies par la fonction $IStar_{D^{c_1}}^{D_{CONFERENCES}}$. Notez que les appellations des instances ont été réduites pour éviter de surcharger la figure, ainsi $i^{D_{CONFERENCES}}{}_1 = c_1$, $i^{D_{AUTEURS}}{}_1 = Au_1$, $i^{D_{ARTICLES}}{}_1 = Art_1$, $i^{D_{DATES}}{}_1 = d_1$…

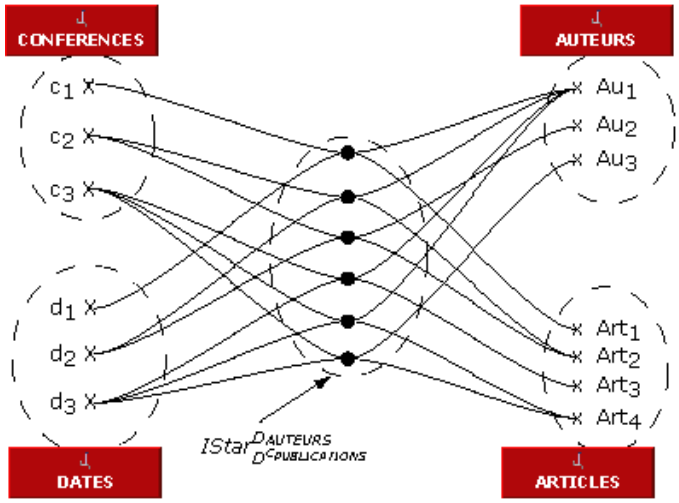


Figure 27 –Représentation graphique des instances de la dimension non partagée $D_{CONFERENCES}$.

Exemple (**dimension partagée**). Dans notre exemple la dimension *AUTEURS* est partagée car elle est reliée à deux ensembles de dimensions via $Star^G(D_{AUTEURS})$: $D_{AUTEURS} \in D^{c_{PUBLICATIONS}}$ et $D_{AUTEURS} \in D^{c_{PROJETS}}$.

$D_{AUTEURS} = (A^{D_{AUTEURS}}, H^{D_{AUTEURS}}, I^{D_{AUTEURS}}, IStar^{D_{AUTEURS}})$

- $A^{D_{AUTEURS}} = \{Auteur, Nom, Statut, Equipe, Institut, All\}$ où *Auteur* est l'attribut identifiant ($Auteur \in ID$) ;
- $H^{D_{AUTEURS}} = \{HA, HSt\}$;
- $I^{D_{AUTEURS}} = \{ i^{D_{AUTEURS}}{}_1, i^{D_{AUTEURS}}{}_2, i^{D_{AUTEURS}}{}_3\}$ avec

$i^{D_{AUTEURS}}{}_1 = [Auteur : Au1 ;$
$Nom : Ravat ;$
$Statut : MCF ;$

Equipe : *SIG* ;
Institut : *IRIT* ;
All : *all*] ;

$i^{D_{AUTEURS}}{}_4$= [*Auteur* : *Au4* ;
Nom : *Tournier* ;
Statut : *Doctorant* ;
Equipe : *SIG* ;
Institut : *IRIT* ;
All : *all*] ;

$i^{D_{AUTEURS}}{}_5$= [*Auteur* : *Au5* ;
Nom : *Annoni* ;
Statut : *Doctorant* ;
Equipe : *SIG* ;
Institut : *IRIT* ;
All : *all*] ;

- $IStar^{D_{AUTEURS}} = \left\{ IStar^{D_{AUTEURS}}_{D^{c1}} , IStar^{D_{AUTEURS}}_{D^{c2}} \right\}$ où :

$IStar^{D_{AUTEURS}}_{D^{c1}} : I^{D_{AUTEURS}} \rightarrow 2^{I^{D_{CONFERENCES}} \times I^{D_{DATES}} \times I^{D_{ARTICLES}}}$ avec :

$IStar^{D_{AUTEURS}}_{D^{c1}} (i^{D_{AUTEURS}}{}_1) = \{$
$(i^{D_{CONFERENCES}}{}_1 ; i^{D_{DATES}}{}_1 ; i^{D_{ARTICLES}}{}_1)$;
$(i^{D_{CONFERENCES}}{}_2 ; i^{D_{DATES}}{}_2 ; i^{D_{ARTICLES}}{}_2)$;
$(i^{D_{CONFERENCES}}{}_3 ; i^{D_{DATES}}{}_3 ; i^{D_{ARTICLES}}{}_3)$;
$(i^{D_{CONFERENCES}}{}_3 ; i^{D_{DATES}}{}_3 ; i^{D_{ARTICLES}}{}_4) \}$

$IStar^{D_{AUTEURS}}_{D^{c1}} (i^{D_{AUTEURS}}{}_2) = \{$
$(i^{D_{CONFERENCES}}{}_2 ; i^{D_{DATES}}{}_2 ; i^{D_{ARTICLES}}{}_2) \}$

$IStar^{D_{AUTEURS}}_{D^{c1}} (i^{D_{AUTEURS}}{}_3) = \{$
$(i^{D_{CONFERENCES}}{}_3 ; i^{D_{DATES}}{}_3 ; i^{D_{ARTICLES}}{}_4) \}$

et

$IStar^{D_{AUTEURS}}_{D^{c2}} : I^{D_{AUTEURS}} \rightarrow 2^{I^{D_{INSTITUTS}} \times I^{D_{DATES}} \times I^{D_{RAPPORTS}}}$

$IStar^{D_{AUTEURS}}_{D^{c1}} (i^{D_{AUTEURS}}{}_1) = \{$
$(i^{D_{INSTITUTS}}{}_1 ; i^{D_{DATES}}{}_4 ; i^{D_{RAPPORTS}}{}_1) \}$

$IStar^{D_{AUTEURS}}_{D^{c1}} (i^{D_{AUTEURS}}{}_2) = \{$
$(i^{D_{INSTITUTS}}{}_1 ; i^{D_{DATES}}{}_4 ; i^{D_{RAPPORTS}}{}_1) \}$

La Figure 28 représente graphiquement les liaisons entre les instances de la dimension *AUTEURS* partagée entre $D^{c_{PUBLICATIONS}}$ et $D^{c_{PROJETS}}$.

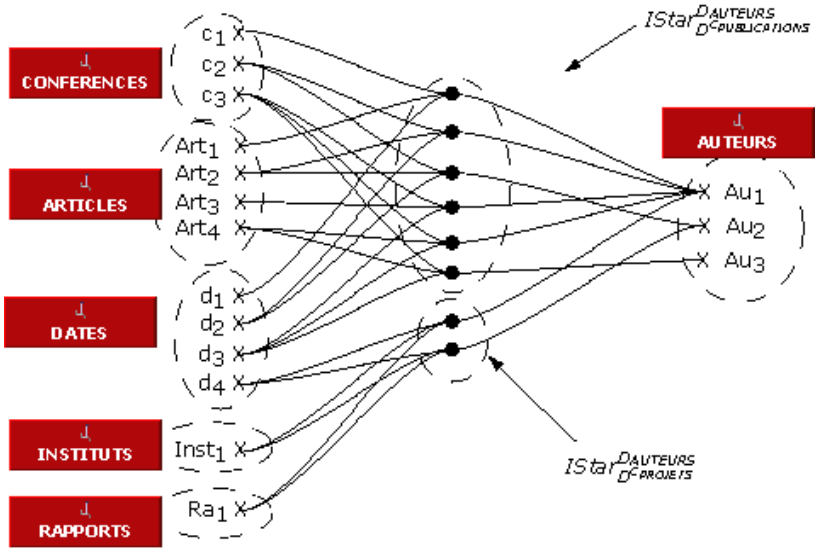


Figure 28 – Représentation des instances de la dimension partagée $D_{AUTEURS}$.

3.2 Concept de hiérarchie

Les hiérarchies constituent différentes perspectives d'analyses sur une même dimension. Une hiérarchie organise les attributs d'une dimension selon les niveaux de granularité qu'ils représentent.

Définition. Une *hiérarchie* H^{Di}_j (ou plus simplement H_j) est définie par ($Param^{Hj}$, $Weak^{Hj}$) :

- $Param^{H_j} =< Id, p_1^{H_j}, \ldots, p_{np}^{H_j}, All >$ est un ensemble ordonné d'attributs, appelés *paramètres* qui représentent les niveaux de granularité de la dimension, $\forall k \in [1..np], p^{Hj}{}_k \in A^{Di}$;
- $Weak^{H_j} : Param^{H_j} \rightarrow 2^{A^{D_i} - Param^{H_j}}$ est une fonction associant des *attributs faibles* aux paramètres, complétant la sémantique de ceux-ci (cf. Eq. 2 pour plus de détails).

$$Weak^{H_i} : Param^{H_i} \longrightarrow 2^{A^{D_i} - Param^{H_i}} \qquad p_x^{H_i} \mapsto \{a_{y_1}^{D_i}, \ldots, a_{y_n}^{D_i}\}$$

Eq. 2

Toute hiérarchie commence par l'attribut identifiant *Id*, appelé *paramètre racine*, et se termine par l'attribut générique de plus forte granularité : *All*, appelé *paramètre extrêmité*.

Formalisme graphique. Afin de représenter de manière explicite la hiérarchie des attributs d'une dimension, nous proposons un formalisme graphique (cf. Figure 29), inspiré de [Ravat et al., 2007e]. Une dimension est représentée par un rectangle rouge (gris foncé). Les attributs de la dimension sont représentés soit en tant que paramètres, par un rond jaune (gris clair) accompagné de son nom, soit en tant qu'attribut faible, par un segment rattachant son intitulé au paramètre auquel il est associé. Les attributs sont organisés selon une ou plusieurs hiérarchies.

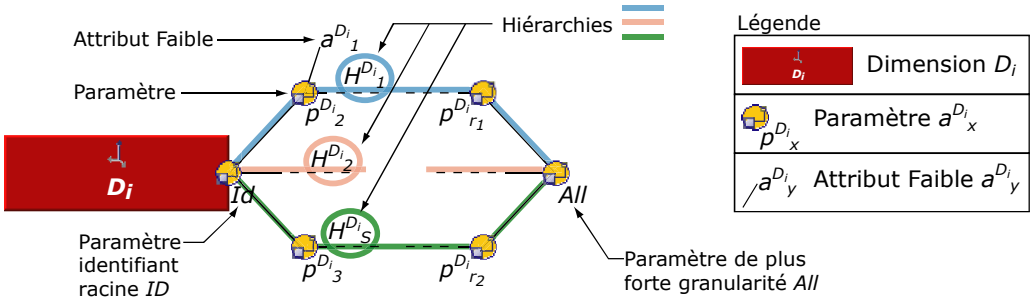


Figure 29 – Représentation graphique d'une dimension.

Remarque. Pour alléger les notations, *All* étant un paramètre système, nous ne le représenterons pas graphiquement dans les exemples suivants. Selon [Malinowski & Zimányi, 2006], ce paramètre système a même tendance à perturber la vision utilisateur du schéma conceptuel. Pour plus de clarté, le paramètre *Id* sera nommé autrement : il prendra généralement le nom au singulier de la dimension.

Exemple. Nous détaillons les hiérarchies de la dimension *AUTEURS* de la galaxie G_1. $H^{D_{AUTEURS}} = \{HA, HSt\}$. La représentation graphique est présentée en Figure 30.

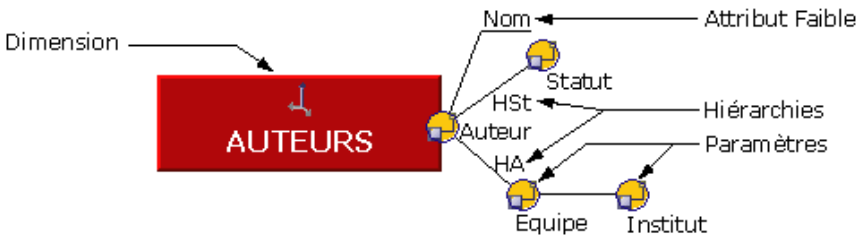


Figure 30 – Représentation graphique de la dimension *AUTEURS*.

$HA = (Param^{HA}, Weak^{HA})$

- $Param^{HA} = <Auteur, Equipe, Institut, All>$
- $Weak^{HA} = \{Auteur \rightarrow \{Nom\}\ \}$

$HSt = (Param^{HSt}, Weak^{HSt})$

- $Param^{HSt} = <Auteur, Statut, All>$
- $Weak^{HSt} = \{Auteur \rightarrow \{Nom\}\ \}$

Exemple (complet). La Figure 31 représente la galaxie G_1 spécifiée précédemment avec les dimensions complètement déployées, c'est-à-dire, avec les attributs et les hiérarchies des dimensions représentés graphiquement.

Dans cet exemple, notez la dimension *ARTICLES* dont la hiérarchie *HS* est constituée de paramètres qui représentent des parties des articles. Cette dimension est constituée des composants des articles. Le nom des dimension est une indication sur ce que ces dimensions représentent.

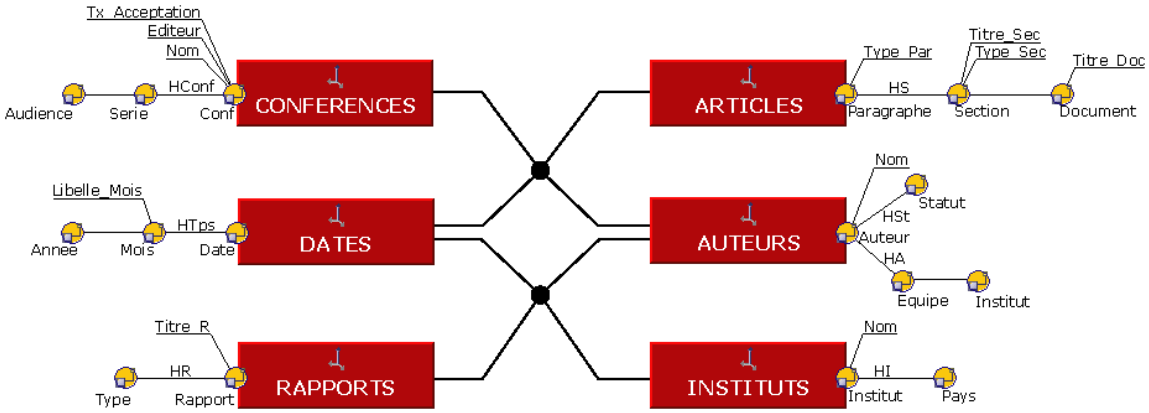


Figure 31 – Représentation de G_1 avec les dimensions déployées.

La section suivante présente plus en détails la modélisation des liens issus des documents.

4 Liens

Les sources de données telles que des documents fournissent un cadre « navigationnel » au sein d'un document ou au sein d'une collection ; par exemple les liens hypertextes contenus dans un document permettent cette navigation. Nous souhaitons conserver cette capacité à naviguer dans la base de données multidimensionnelles. En effet, lors d'une analyse, l'emploi des liens permet des perspectives d'analyses accrues.

Nous considérons un lien comme étant une liaison entre deux attributs par une relation « correspond à » entre des valeurs de ces deux attributs. Ce lien peut être matérialisé dans les documents sources, comme un renvoi hypertexte vers un autre document. Mais le lien peut également ne pas être matérialisé, en effet les références d'un article scientifique décrivent en détails des publications, mais il est très rare de trouver des liens pointant directement vers le contenu de chaque article (par exemple un lien vers un fichier pdf).

Les liens $Lk^G = \{l_1,\ldots,l_u\}$ peuvent exister au sein d'une même dimension (lien intra-dimension) ou bien entre des dimensions (lien inter-dimension).

Définition. Soient deux attributs $a^{Di}{}_u$ et $a^{Dj}{}_v$ (pouvant être de la même dimension, $i=j$). Un lien l_i qui relie les valeurs des deux attributs $a^{Di}{}_u$ et $a^{Dj}{}_v$ entre elles est défini par la fonction :

- $l_i : dom\left(a_u^{D_i}\right) \longrightarrow dom\left(a_v^{D_j}\right)$ où $dom(a^{Di}{}_u)$ (respectivement $dom(a^{Dj}{}_v)$) est l'ensemble des valeurs de $a^{Di}{}_u$ (resp. $a^{Dj}{}_v$).

Dans certains cas, l_i n'est pas défini sur son ensemble de départ $dom(a^{Di}{}_u)$, mais un sous-ensemble : seules certaines valeurs de $a^{Di}{}_u$ sont liées à des valeurs de $a^{Dj}{}_v$. La restriction suivante est alors employée :

- $l_{i|P} : dom\left(a_u^{D_i}\right) \longrightarrow dom\left(a_v^{D_j}\right)$ où $l_{i|P}$ est la restriction de la fonction l_i au sous-ensemble P, $P \subseteq dom(a^{Di}{}_u)$ tel que l_i est définie pour tout élément de P.[14]

Remarque. Le sous-ensemble P peut être spécifié par un prédicat restrictif permettant la restriction de $dom(a^{Di}{}_u)$.

[14] La notation mathématique $f_{i|P}$ représente la *restriction* de la fonction $f_i : E \rightarrow F$ dont l'ensemble de départ E est réduit à P, P étant un sous-ensemble de E ($P \subseteq E$).

Formalisme graphique. Les liens sont représentés par une flèche entre les deux attributs concernés. Les liens sont accompagnés d'un libellé descriptif.

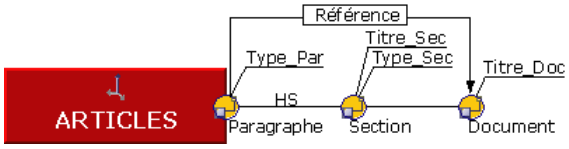


Figure 32 – Représentation graphique d'un lien au sein d'une dimension.

Exemple. Afin de compléter la galaxie G_1, et de permettre l'analyse présentée en Tableau 6, il est nécessaire de modéliser les liens représentés par la bibliographie d'articles scientifiques. De plus, afin de permettre des analyses comparatives entre des auteurs de rapports de projets appartenant aux instituts pilotant les projets et les auteurs dépendant d'instituts extérieurs, il est nécessaire de spécifier un lien entre les instituts des auteurs et les instituts pilotes de projets.

Les deux liens de la galaxie G_1 sont spécifiés par :

- Un *lien intra-dimension* qui modélise le lien entre les références d'un article (qui sont des paragraphes) et les articles correspondant à ces références :

  $l_1 = l_{Reference|Ref}$: $dom(Paragraphe) \rightarrow dom(Document)$, avec $Ref \subseteq dom(Paragraphe)$
  Les paragraphes de type références sont spécifiées par le prédicat paragraphes de type : $p_{Ref} = \{Paragraphe \mid Type\_Par = \text{'ref'}\}$

- Un *lien inter-dimension* permet de lier les instituts entre eux (ainsi les instituts des auteurs sont liés aux instituts dépositaires de rapports de projets). Les noms des dimensions sont spécifiés pour enlever toute ambiguïté :

  $l_2 = l_{Institut}$: $dom(AUTEURS.Institut) \rightarrow dom(INSTITUTS.Institut)$.

La Figure 33 représente graphiquement la galaxie G_1 avec ses deux liens $Lk^{G1} = \{l_1, l_2\}$. Le premier lien ($l_1 = l_{Référence}$) associe les paragraphes de type référence aux documents qui représentent ces références (des articles). Le second lien ($l_2 = l_{Institut}$) associe les instituts d'auteurs aux instituts qui pilotent des projets (et qui sont donc les émetteurs des rapports).

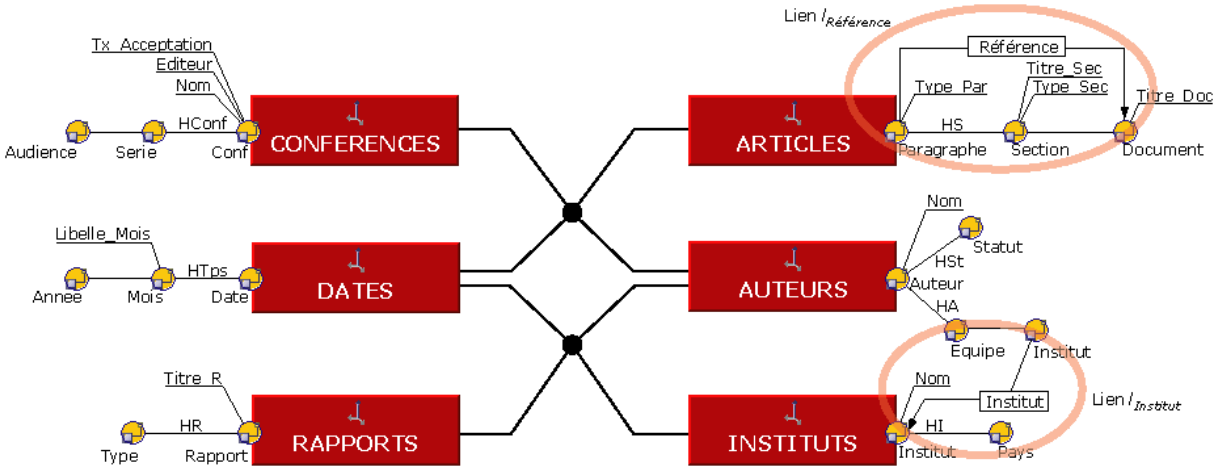


Figure 33 – Représentation complète de la galaxie G_1 (avec les liens associés).

Exemple. Sur le lien l_1, la première référence de ce chapitre est [Agrawal et al., 1997]. Ainsi l_1 associe la valeur de cette référence à l'article complet intitulé : « Modeling Multidimensional Databases ». Dans notre exemple, le lien ne concerne qu'un sous-ensemble des paragraphes : les paragraphes représentant des références sont de type référence (*Type_Par = Ref*). Ainsi le lien est la fonction réduite aux paragraphes qui sont des références.

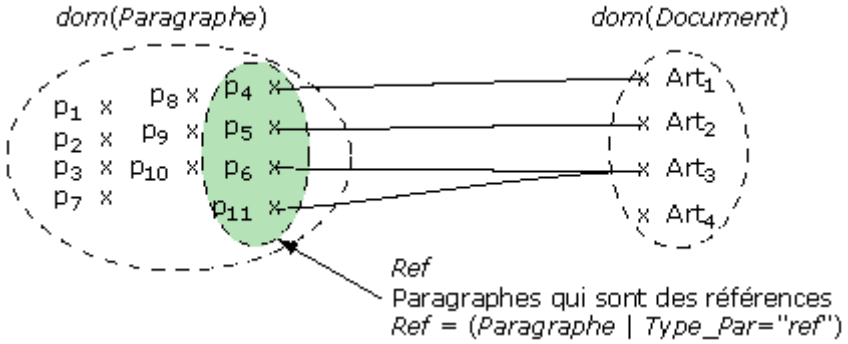


Figure 34 – Exemples de liens entre certains paragraphes et des articles.

Exemple. La Figure 35 présente quelques exemples de liens entre les instituts auxquels appartiennent les auteurs et les instituts pilotes de projets.

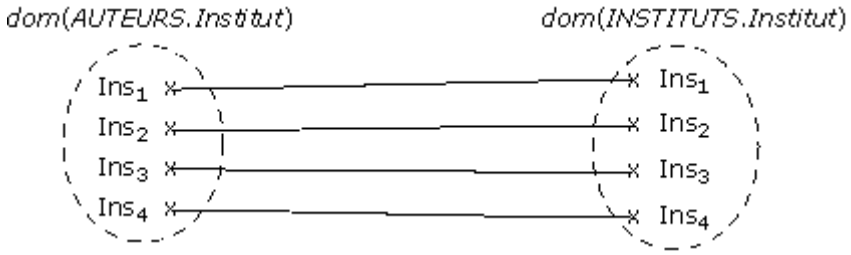


Figure 35 – Exemples de liens entre certains instituts d'auteurs et les instituts qui pilotent les projets.

Les liens au sein d'une galaxie sont des correspondances fonctionnelles (également appelées relations fonctionnelles ou encore fonctions). En informatique, ces dernières sont désignées par le terme de *fonctions partielles*. Une autre spécificité du modèle en galaxie est sa capacité à modéliser les données textuelles issues de documents XML afin de permettre leur analyse multidimensionnelle.

5 Modélisation de données textuelles

Dans le modèle en galaxie, toutes les données multidimensionnelles sont représentées par des attributs. Comme dans un modèle classique, tous les attributs ne sont pas du même type. Plus particulièrement, nous distinguons le type d'attribut documentaire et de dimension documentaire.

5.1 Types d'attributs

L'agrégation de données permet la synthèse d'informations lors d'analyses multidimensionnelles. Cette agrégation se fait par l'application de fonctions d'agrégations sur

des valeurs d'attributs numériques. Seulement, tous les attributs ne sont pas nécessairement compatibles avec les fonctions d'agrégation disponibles. On dira qu'ils ne sont pas nécessairement « agrégeables ». Toutefois, il existe deux fonctions d'agrégation qui peuvent opérer sur tout type d'attribut : COUNT, qui compte le nombre de valeurs et LIST qui donne la liste des valeurs sans les agréger.

Au sein des bases de données multidimensionnelles, nous proposons de distinguer trois catégories d'attributs :

- attributs *numériques* : ce sont des attributs composés de données numériques tels que des montants de ventes, un numéro de téléphone…
- attributs *textuels* : ce sont des attributs composés de données alphanumériques pouvant représenter un mot, un paragraphe ou encore un long texte,
- attributs *complexes* : ce sont des attributs spécifiques tels que des attributs spatiaux [Parent et al., 1999] (points, segments…) ou encore des images. Des objets complexes de manière générale.

Les attributs complexes sortent du cadre de cette thèse, aussi nous ne les aborderons pas. Les différents types d'attributs sont agrégeables selon les fonctions d'agrégation employées. Ainsi les fonctions d'agrégations sont associées aux différents attributs afin de ne pas permettre au décideur d'exprimer des analyses impossibles.

5.1.1 Attributs numériques

En fonction du type d'agrégation, il est possible de distinguer trois types d'attributs numériques [Kimball, 1996] et [Horner et al., 2004] :

- attributs numériques *additifs* : ces attributs représentent des valeurs qui peuvent être additionnées, l'ensemble des fonctions d'agrégations classiques peuvent être employées (SOMME, MOYENNE, MIN, MAX,…). Il s'agit des fonctions d'agrégation dites additives ou semi-additives ;
- attributs numériques *semi-additifs* : ces attributs représentent des niveaux, tels que des températures ou des quantités de stock. Avec ce type d'attribut la fonction d'agrégation SOMME ne peut être employée selon certaines dimensions (en général la dimension représentant le temps). Les autres fonctions d'agrégation, les fonctions dites semi-additives (telle que la MOYENNE) peuvent être employées ;
- attributs numériques *non-additifs* : ces attributs ne représentent pas des valeurs qui peuvent être sommés telles que des codes postaux ou encore des numéros de téléphone.

En l'absence de méthodes d'agrégation nouvelles et adaptées, nous considérons les attributs numériques non-additifs comme étant non-agrégeables. Toutefois, il faut noter que des fonctions génériques fonctionnent sur tout type d'attribut (agrégeables ou non). Il s'agit des fonctions effectuant des listes ou des dénombrement. En effet, par exemple, il est possible de lister ou de compter le nombre de numéros de téléphones enregistrés dans un répertoire.

5.1.2 Attributs textuels

Bien que la plupart des attributs textuels soient non-agrégeables, nous distinguons une catégorie qui peut être agrégée avec des fonctions d'agrégations adaptées : les attributs documentaires.

Définition. Les attributs *documentaires* représentent des données issues de documents textuels. Ces données sont agrégeables au moyen de fonctions telles que TOP_KEYWORD, TOPIC, SUMMARY (…) [Park et al., 2005]. Les données textuelles de ces attributs représentent des textes (tel qu'un paragraphe) et non des noms spécifiques (tels que des noms de personnes, de villes, de catégories de produits…).

Remarque. Il faut noter que les autres attributs textuels peuvent être « agrégeables » sous la condition qu'il existe une fonction adaptée permettant leur agrégation. Par exemple, des villes représentées par des positions géographiques pourraient être agrégées via des fonctions géométriques telles qu'un calcul de barycentre. Ceci permettrait l'agrégation de plusieurs points représentant différentes villes d'être agrégées en un seul point.

5.2 Dimension documentaire

Les documents analysés sont constitués d'une structure, par exemple, ce mémoire est composé de paragraphes contenus dans des sections, elles-mêmes contenues dans des chapitres. La représentation de cette structure dans le modèle en galaxie regroupe des attributs documentaires au sein de dimensions documentaire.

5.2.1 Définition

Une dimension documentaire est caractérisée par la présence d'attributs documentaires. Ces attributs sont organisés de telle manière à représenter la structure du document modélisé par la dimension. Par structure de documents, nous entendons, la *structure logique* [Mbarki, 2007], à savoir la structuration du texte en paragraphes, sections…

Définition. Une *dimension documentaire* est composée d'attributs documentaires et au moins l'une des hiérarchies de la dimension modélise la structure logique du document. Cette hiérarchie représente une vision à différents niveaux des attributs textuels. Ce type de dimension peut aussi inclure d'autre type d'attributs.

La section suivante détaille la dimension documentaire *ARTICLES* présentée dans la spécification de la galaxie G_1 (cf. Figure 33).

Remarque. D'autres types de dimensions peuvent être envisagées, mais, reposant sur des attributs complexes, elles sortent du cadre de cette thèse et ne seront pas détaillées. Parmi celles-ci, il est possible de définir des dimensions géographiques [Stefanovic et al., 2000], des dimensions temporelles…

5.2.2 Exemple

La galaxie G_1 est conçue en partie pour permettre l'analyse d'articles scientifiques. Ces articles sont structurés de manière similaires : ils sont composés de texte et ce texte est subdivisé en sections, elles-mêmes subdivisées en paragraphes (cf. Figure 36).

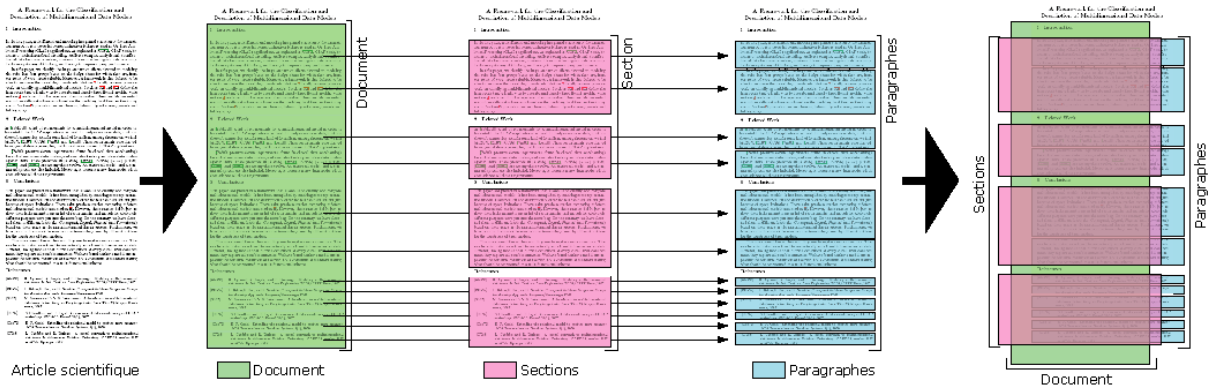


Figure 36 – Décomposition d'un article scientifique.

Ces documents, sont modélisés par une dimension pour permettre leur intégration dans un environnement d'analyse multidimensionnelle. Dans la suite, nous détaillons la dimension *ARTICLES* de la galaxie G_1 (cf. Figure 37). Cette dimension est composée d'une unique hiérarchie (*HS*) qui modélise la structure des articles. Elle fournit une vision à trois niveaux du contenu des articles, à savoir : les niveaux paragraphe, section et document (l'article complet). Chaque niveau est modélisé par un paramètre.

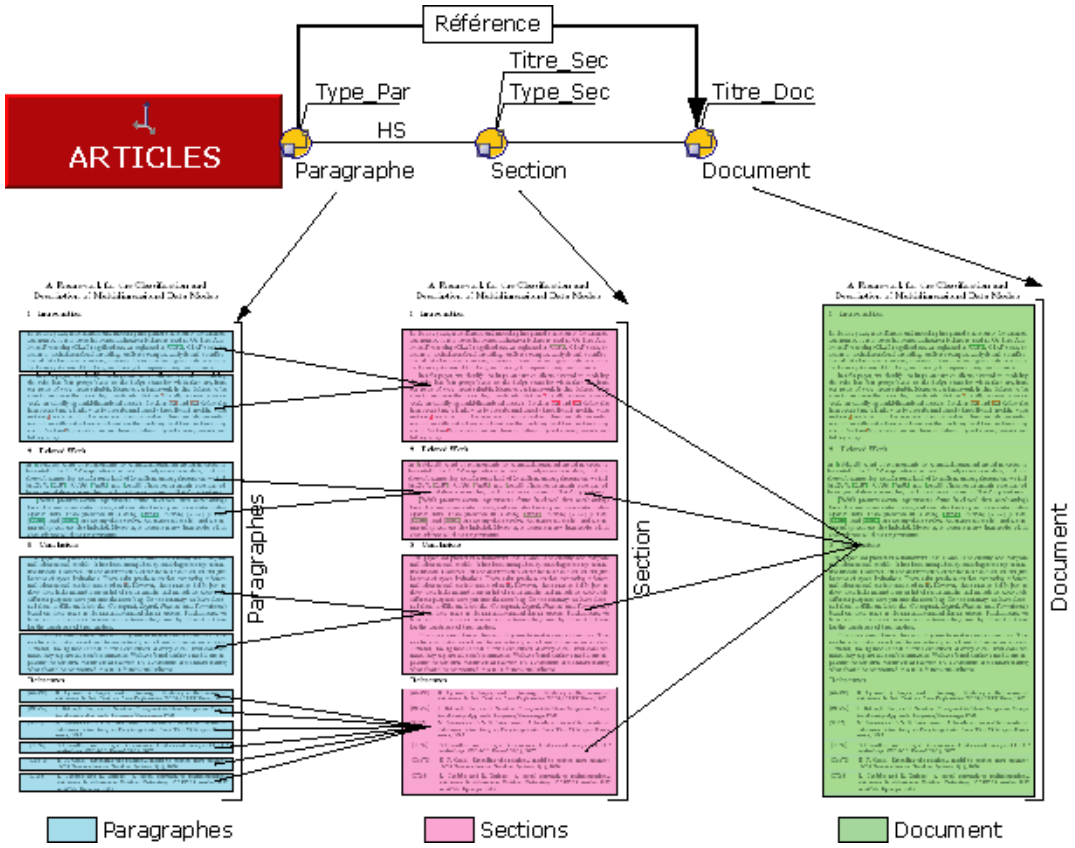


Figure 37 – Détails de la dimension *ARTICLES* de G_1 (trois différents niveaux de vision du contenu des articles : paragraphe, section et document).

Des données complémentaires extraites du contenu des articles sont associées aux données textuelles. Les paragraphes sont associés à leur type (standard, définition, théorème, référence…). Les sections, sont elles aussi associée à leur type (introduction, conclusion, bibliographie…) ainsi que leur titre. Quant au document complet il est associé à son titre. La

Figure 38 présente la décomposition des données d'un article selon la structure de la dimension *ARTICLES*. Une colonne *Id* a été ajoutée dans l'image pour permettre une identification claire de chaque instance de paragraphe.

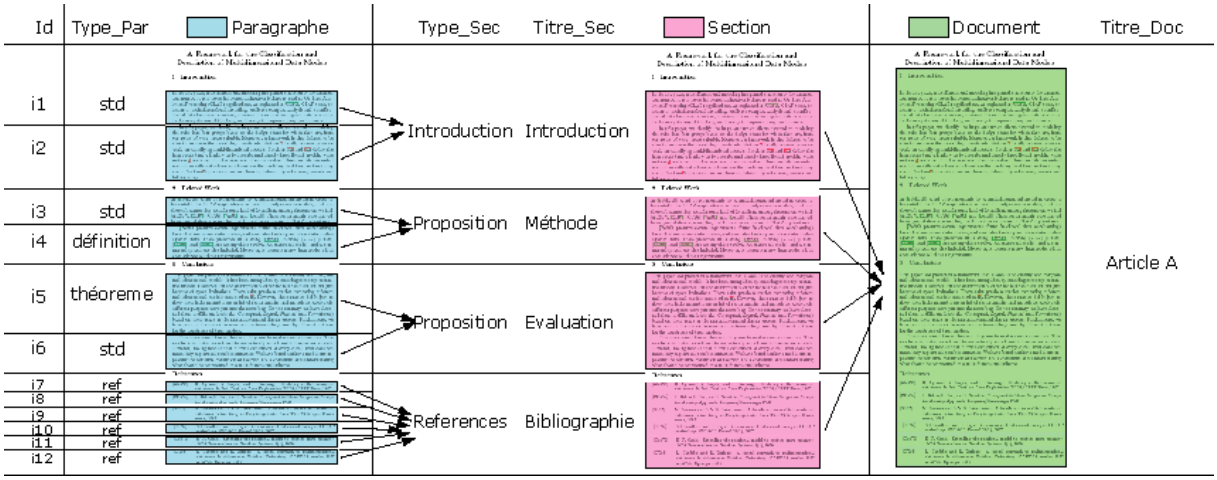


Figure 38 – Représentation des données de la dimension *ARTICLES* (*All* n'est pas représenté).

La Figure 39 présente les données avec le même formalisme de la section 3.1 sous la forme d'une vision dénormalisée par rapport à la Figure 38. Pour des raisons de place, les noms des attributs ont été réduits (*Par* pour *Paragraphe*, *Type_P* pour *Type_Par*, *Sec* pour *Section*…). Pour la même raison, le document (bloc en vert associé à une accolade) n'est représenté qu'une fois alors qu'il devrait l'être pour chaque ligne. Dans cette figure, une ligne correspond aux données d'une instance de la dimension :

[*Id* ; *Paragraphe* ; *Type_Paragraphe* ; *Section* ; *Type_Section* ; *Titre_Section* ; *Document* ; *Titre_Article* ; *All*]

Id	Paragraphe	Type_Paragraphe	Section	Type_Section	Titre_Section	Document	Titre_Doc	All
Id: i1	Par:	Type_P: std	Sec:s1	Type_S: Intro	Titre_S: Introduction	Doc:	Titre_A: Article A	All: all
Id: i2	Par:	Type_P: std						
Id: i3	Par:	Type_P: std	Sec:s2	Type_S: Prop	Titre_S: Méthode			
Id: i4	Par:	Type_P: def						
Id: i5	Par:	Type_P: theor	Sec:s3	Type_S: prop	Titre_S: Evaluation			
Id: i6	Par:	Type_P: std						
Id: i7	Par:	Type_P: ref	Sec:s4	Type_S: Ref	Titre_S: Bibliographie			
Id: i8	Par:	Type_P: ref						
Id: i9	Par:	Type_P: ref						
…								
Id: i12	Par:	Type_P: ref						

Figure 39 – Représentation formelle des données d'un article de la dimension *ARTICLES*.

Chaque ligne est composée d'un ensemble d'attributs. Les attributs textuels contiennent le texte complet du fragment correspondant : le paramètre *Paragraphe* contient le texte qui compose chaque paragraphe ; le paramètre *Section* contient, pour chaque paragraphe le texte complet de la section qui contient le paragraphe et *Document* contient pour chaque paragraphe, le texte complet de l'article qui contient le paragraphe.

Remarque. Notez bien que si le modèle duplique les données au niveau conceptuel, il n'en est pas nécessairement de même pour les autres niveaux (logique ou physique).

Au sein de la dimension *ARTICLES*, un paragraphe de type référence est en réalité un autre document. Ainsi le lien qui existe entre un tel paragraphe et l'article qu'il représente est modélisé par le lien *Référence* entre les paramètres *Paragraphe* et *Document* (cf. Figure 40). *Document* étant l'attribut qui représente le contenu complet d'un article. Concrètement, les références sont des paragraphes particuliers dont le type (*Type_Par*) vaut *ref* et le type de la section (*Type_Sec*) qui contient ces paragraphes est : *Ref*. Dans notre exemple, dans le cadre de l'article intitulé « *Article A* », la section contenant les références s'intitule « *bibliographie* ». Chaque référence est une liaison vers le contenu complet d'un autre article (modélisé par le paramètre *Document*).

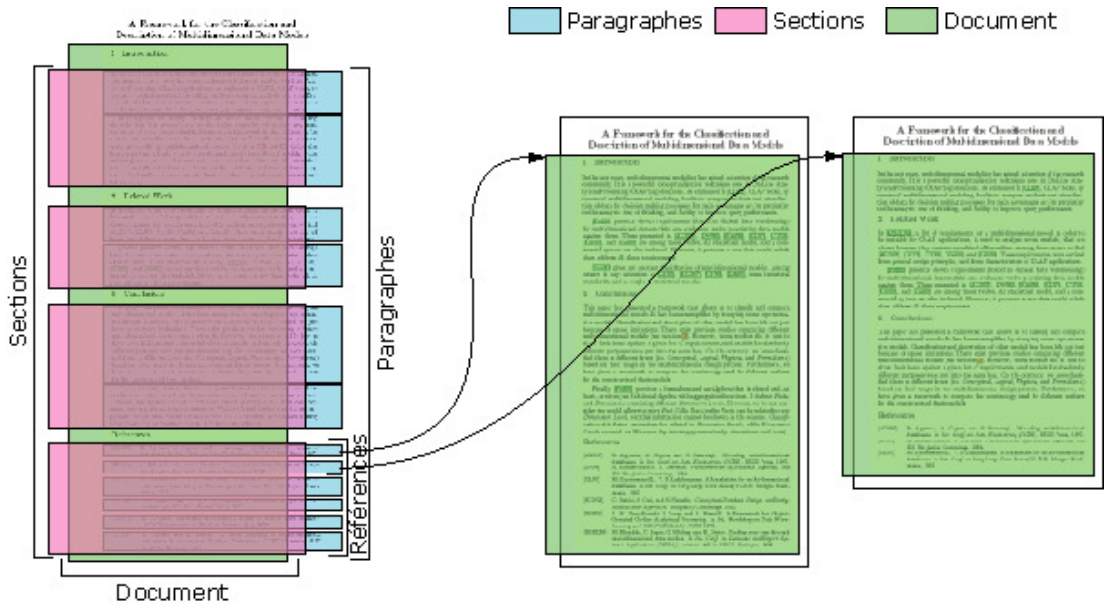


Figure 40 – Liaisons représentant les références d'articles.

Les paragraphes ne sont pas tous des références, ainsi seul un sous domaine des paragraphes représente l'ensemble de références. Formellement, il est possible d'écrire :

$$dom(références) \subseteq dom(Paragraphe)$$

Fort de ce type particulier de dimension, le modèle en galaxie permet toujours des sujets d'analyse classiques (les faits).

5.3 Cas particulier : données numériques

Le modèle en galaxie n'est pas exclusivement réservé à la modélisation de besoins d'analyse sur des données issues de documents. Il est aussi possible de modéliser des analyses classiques telles que celles modélisées par notre modèle en constellation [Ravat et al., 2007e].

5.3.1 Spécification

L'absence de fait dans le modèle en galaxie n'exclut pas pour autant les modélisations possibles avec un modèle en constellation. Le principe est de traduire la structure des faits, constitués de mesures, du modèle en constellation par une dimension qui sera constituée d'attributs numériques additifs ou semi-additifs.

La conversion d'un fait peut se faire de deux manières :

- une dimension dotée d'un unique paramètre racine *Id* et chaque mesure convertie en attribut faible rattaché au paramètre *Id*, cf. Figure 41 (a) ;
- une dimension dotée d'un paramètre racine *Id* et chaque mesure est convertie en un paramètre, chacun associé au paramètre racine par une hiérarchie différente, cf. Figure 41 (b).

La seconde solution, bien que plus complexe que la première, est intéressante pour permettre la construction de hiérarchies avec les mesures en y ajoutant de nouveaux paramètres. La notion de hiérarchisation de mesures a aussi été suggérée avec les « faits de faits » [Schneider, 2003] et [Schneider, 2007].

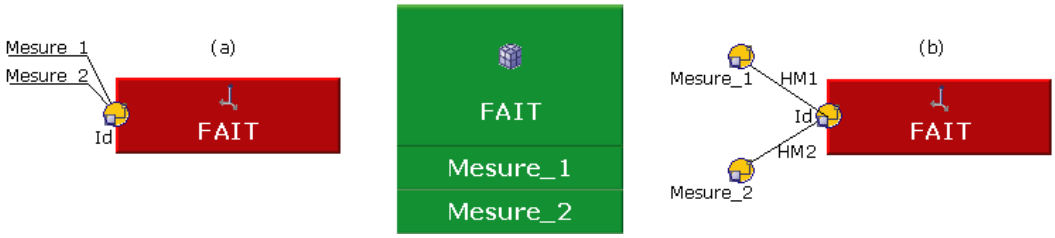


Figure 41 – Deux possibilités de conversions d'un fait : (a) par attributs faibles ; (b) par paramètres.

5.3.2 Exemple

Dans l'introduction générale du mémoire, un exemple d'analyse de ventes avait été présenté. Il est converti, ici, en un schéma en galaxie (cf. Figure 42). Nous avons opté pour la première solution pour convertir le fait *VENTES*.

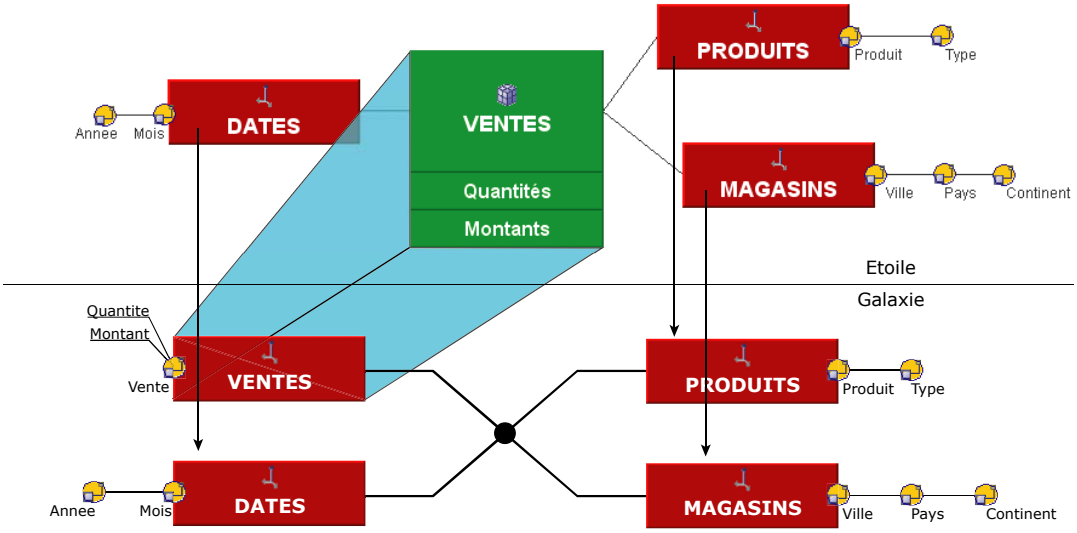


Figure 42 – Conversion du fait d'un schéma en étoile en une galaxie.

Le fait *VENTES* est converti en la dimension D_{VENTES} :

$D_{VENTES} = (A^{D_{VENTES}}, H^{D_{VENTES}}, I^{D_{VENTES}}, IStar^{D_{VENTES}})$

- $A^{D_{VENTES}} = \{Vente, Montants, Quantités, All\}$ où *Vente* est l'attribut identifiant ($Vente \in ID$) ;
- $H^{D_{VENTES}} = \{HV\}$;
- $I^{D_{VENTES}} = \{i_1, i_2, i_3\ i_4, i_5, i_6, ...\}$ avec

$i_1 = [\ Vente : v_1\ ;\ Montants : 2200\ ;\ Quantités : 1\ ;\ All : \text{all}\ ]$
$i_2 = [\ Vente : v_2\ ;\ Montants : 4400\ ;\ Quantités : 2\ ;\ All : \text{all}\ ]$
$i_3 = [\ Vente : v_3\ ;\ Montants : 4400\ ;\ Quantités : 2\ ;\ All : \text{all}\ ]$
$i_4 = [\ Vente : v_4\ ;\ Montants : 15400\ ;\ Quantités : 7\ ;\ All : \text{all}\ ]$
$i_5 = [\ Vente : v_5\ ;\ Montants : 3100\ ;\ Quantités : 1\ ;\ All : \text{all}\ ]$
$i_6 = [\ Vente : v_6\ ;\ Montants : 27900\ ;\ Quantités : 9\ ;\ All : \text{all}\ ]$
…

- $IStar^{D_{VENTES}} = \left\{IStar^{D_{VENTES}}_{D^{ca}}\right\}$ où $IStar^{D_{VENTES}}_{D^{c1}} : I^{D_{VENTES}} \rightarrow 2^{I^{D_{DATES}} \times I^{D_{PRODUITS}} \times I^{D_{MAGASINS}}}$ et :

$IStar^{D_{VENTES}}_{D^{c1}}(i_1) = \{(i^{DATES}_1, i^{PRODUITS}_1, i^{MAGASINS}_1)\}$
$IStar^{D_{VENTES}}_{D^{c1}}(i_2) = \{(i^{DATES}_1, i^{PRODUITS}_1, i^{MAGASINS}_2)\}$
$IStar^{D_{VENTES}}_{D^{c1}}(i_3) = \{(i^{DATES}_1, i^{PRODUITS}_1, i^{MAGASINS}_3)\}$
$IStar^{D_{VENTES}}_{D^{c1}}(i_4) = \{(i^{DATES}_1, i^{PRODUITS}_1, i^{MAGASINS}_4)\}$
$IStar^{D_{VENTES}}_{D^{c1}}(i_5) = \{(i^{DATES}_1, i^{PRODUITS}_2, i^{MAGASINS}_1)\}$
$IStar^{D_{VENTES}}_{D^{c1}}(i_6) = \{(i^{DATES}_1, i^{PRODUITS}_2, i^{MAGASINS}_4)\}$
…

Où i^{DATES}_1 correspond à « nov. 2005 », $i^{PRODUITS}_1$ est le produit « Dell Power Edge », $i^{MAGASINS}_1$ est le magasin de la ville de « Paris »… Les données de cet exemple correspondent aux deux premières colonnes de la représentation des données en cube dans l'introduction générale de la thèse (les valeurs nulles ne sont pas représentées).

La hiérarchie *HV* de la dimension *VENTES* est spécifiée comme suit :

$HV = (Param^{HV}, Weak^{HV})$

- $Param^{HV} = \langle Vente, All \rangle$
- $Weak^{HV} = \{Vente \rightarrow \{Montants, Quantites\}\}$

Ainsi, par son unique concept de dimension, le modèle conceptuel en galaxie permet toujours de modéliser un schéma multidimensionnel classique tel qu'un schéma en constellation ou encore un schéma en étoile.

6 Bilan

En conclusion, nous avons proposé un modèle conceptuel multidimensionnel en galaxie pour répondre à nos objectifs. Le modèle en galaxie reprend les avantages des modèles conceptuels multidimensionnels. Il permet de faire abstraction des contraintes logiques et physiques de l'environnement pour obtenir une vision orientée décideur [Golfarelli et al., 2002]. A l'instar du modèle en constellation [Ravat et al., 2007e], le modèle en galaxie dispose aussi d'une représentation d'axes d'analyses à multiples perspectives en fournissant différents niveaux de

granularité pour des analyses poussées. Malgré son concept unique de dimension, le modèle en galaxie est aussi expressif que les modèles classiques reposant sur les concepts de fait et de dimension.

Le modèle en galaxie répond aux objectifs fixés : il permet la représentation de la *structure hiérarchique* de documents au moyen de dimensions documentaires. Les *liens internes ou externes* des documents sont conservés et représentés dans le modèle pour permettre leur utilisation lors de la navigation (décrite dans le chapitre suivant). Les *indicateurs de types documentaires permettent une analyse sur le contenu* des documents. Enfin, l'absence de sujet d'analyse prédéfinis fournit à l'utilisateur une *flexibilité adéquate* pour lui permettre de réorienter l'analyse en cas de non-sens sur une analyse de texte.

Toutefois, le modèle est assez générique pour ne pas être réservé à l'analyse de données issues de documents. Il permet aussi de modéliser une étoile ou une constellation. Ceci est une conséquence de l'unique concept de dimension qui permet une simplification de la modélisation multidimensionnelle. La galaxie est une généralisation des modèles multidimensionnels dotés de deux concepts (fait et dimension) tel que la constellation. En outre, certains auteurs ont insistés sur la nécessité de symétrie entre les attributs et les mesures des modèles multidimensionnels [Agrawal et al., 1997], [Cabibbo & Torlone, 1997] et [Gyssens & Lakshmanan, 1997]. Toutefois, comme cela sera montré dans le chapitre suivant, la notion de mesure ne permet pas une symétrie systématique contrairement à notre modèle et ses dimensions.

Le chapitre suivant présente la spécification d'analyse multidimensionnelle à partir de la représentation en galaxie. Ces opérations, associées à des fonctions d'agrégation adaptées aux données textuelles, permettent la spécification d'analyse mais aussi la manipulation de ces analyses sur des données textuelles issues de documents.

Références

[Agrawal et al., 1997] Rakesh Agrawal, Ashish Gupta, Sunita Sarawagi, "Modeling Multidimensional Databases", *13th Intl. Conf. on Data Engineering* (*ICDE*), IEEE Computer Society, p. 232–243, 1997.

[Annoni, 2007] Estella Annoni, *Eléments méthodologiques pour le développement des systèmes décisionnels dans un contexte de reutilisation*, Thèse de doctorat, Université Paul Sabatier, Toulouse 3 (France), juillet 2007.

[Bondy & Murty, 1976] Adrian J. Bondy, U.S.R. Murty, *Graph Theory with Applications*, Elsevier North-Holland, 1976.

[Boussaid et al., 2006] Omar Boussaid, Riadh Ben Messaoud, Rémy Choquet, Stéphane Anthoard, "X-Warehousing: An XML-Based Approach for Warehousing Complex Data", *10th East European Conf. on Advances in Databases and Information Systems* (*ADBIS*), LNCS 4152, Springer, p. 39–54, 2006.

[Cabibbo & Torlone, 1997] Luca Cabibbo, Riccardo Torlone, "Querying Multidimensional Databases", *6th Intl. Workshop Database Programming Languages* (*DBPL*), LNCS 1369, Springer, p. 319–335, 1997.

[Dublin Core, 2007] The Dublin Core Metadata Initiative de http://dublincore.org/ (Dublin Core Metadata Element Set, version 1.1) en date de mai 2007.

[Ghozzi, 2004] Faiza Ghozzi, *Conception et manipulation de bases de données dimensionnelles à contraintes*, Thèse de doctorat, Université Paul Sabatier, Toulouse 3 (France), novembre 2004.

[Golfarelli et al., 1998] Matteo Golfarelli, Dario Maio, Stefano Rizzi, "The Dimensional Fact Model: A Conceptual Model for Data Warehouses", invited paper, *Intl. Journal of Cooperative Information Systems* (*IJCIS*), vol.7(2-3), World Scientific Publishing, p. 215–247, juin & septembre1998.

[Golfarelli et al., 2002] Matteo Golfarelli, Stefano Rizzi, Ettore Saltarelli, "WAND: A CASE Tool for Workload-Based Design of a Data Mart", *Decimo Convegno Nazionale su Sistemi Evoluti per Basi di Dati* (*SEBD*), p. 422–426, 2002.

[Gray et al., 1996] Jim Gray, Adam Bosworth, Andrew Layman, Hamid Pirahesh, "Data Cube: A Relational Aggregation Operator Generalizing Group-By, Cross-Tab, and Sub-Total", *12th Intl. Conf. on Data Engineering* (*ICDE*), IEEE Computer Society, p. 152–159, 1997.

[Gyssens & Lakshmanan, 1997] Marc Gyssens, Laks V. S. Lakshmanan, "A Foundation for Multi-dimensional Databases", *23rd Intl. Conf. on Very Large Data Bases* (*VLDB*), Morgan Kaufmann, p. 106–115, 1997.

[Horner et al., 2004] John Horner, Il-Yeol Song, Peter P. Chen, "An analysis of additivity in OLAP systems", *7th ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 83–91, 2004.

[Horner et al., 2004] John Horner, Il-Yeol Song, "A Taxonomy of Inaccurate Summaries and Their Management in OLAP Systems", *24th Intl. Conf. on Conceptual Modeling* (*ER*), LNCS 3716, Springer, p. 433–448, 2005.

[Jagadish et al., 1999] H. V. Jagadish, Laks V. S. Lakshmanan, Divesh Srivastava, "What can Hierarchies do for Data Warehouses?", *25th Intl. Conf. on Very Large Data Bases* (*VLDB*), Morgan Kaufmann, p. 530–541, 1999.

[Johnson & Chatziantoniou, 1999] Theodore Johnson, Damianos Chatziantoniou, "Extending Complex Ad-Hoc OLAP", *8th Intl. Conf. on Information and Knowledge Management* (*CIKM*), ACM Press, p. 170–179, 1999.

[Kimball, 1996] Ralph Kimball, *The data warehouse toolkit: Practical Techniques for Building Dimensional Data Warehouses*, John Wiley and Sons, ISBN : 0-471-15337-0, 1996, 2ème ed. : Ralph Kimball, Margaery Ross, *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling*, *2nd Edition*, John Wiley & Sons, 2002.

[Lenz & Shoshani, 1997] Hans-Joachim Lenz, Arie Shoshani, "Summarizability in OLAP and Statistical Data Bases", *9th Intl. Conf. on Scientific and Statistical Database Management* (*SSDBM*), IEEE Computer Society, p. 132–143, 1997.

[Lenz & Thalheim, 2001] Hans-Joachim Lenz, Bernhard Thalheim, "OLAP Databases and Aggregation Functions", *13th Intl. Conf. on Scientific and Statistical Database Management* (*SSDBM*), IEEE Computer Society, p. 91–100, 2001.

[Malinowski & Zimányi, 2006] Elzbieta Malinowski, Esteban Zimányi, "Hierarchies in a multidimensional model: From conceptual modeling to logical representation", *Data & Knowledge Engineering* (*DKE*), vol.59(2), Elsevier, p. 348–377, novembre 2006.

[Mbarki, 2007] Mohamed Mbarki, *Gestion d'hétérogénéité documentaire : le cas d'un entrepôt de documents multimédias*, thèse de doctorat, Université Paul Sabatier Toulouse 3 (France), 2007 (à paraître).

[Messaoud, 2006] Riadh Ben Messaoud, *Couplage de l'analyse en ligne et de la fouille de données pour l'exploration, l'agrégation et l'explication des données complexes*, thèse de doctorat, Université Lumière Lyon 2 (France), novembre 2006.

[Nassis et al., 2004] Vicky Nassis, Rajagopal Rajugan, Tharam S. Dillon, J. Wenny Rahayu, "Conceptual Design of XML Document Warehouses", *6th Intl. Conf. on Data Warehousing and Knowledge Discovery* (*DaWaK*), LNCS 3181, Springer, p. 1–14, 2004.

[Oracle Spatial, 2006] *Oracle Spatial, User's Guide and Reference 10g Release 2* (*10.2*), Oracle, B14255-03, mars 2006.

[Parent et al., 1999] Christine Parent, Stefano Spaccapietra, Esteban Zimányi, "Spatio-Temporal Conceptual Models: Data Structures + Space + Time", *7th Intl. Symposium on Advances in Geographic Information Systems* (*ACM-GIS*), ACM Press, p. 26–33, 1999.

[Park et al., 2005] Byung-Kwon Park, Hyoil Han, Il-Yeol Song, "XML-OLAP: A Multidimensional Analysis Framework for XML Warehouses", *7th Intl. Conf. on Data Warehousing and Knowledge Discovery* (*DaWaK*), LNCS 3589, Springer, p. 32–42, 2005.

[Pourrabas & Rafanelli, 2000] Elaheh Pourabbas, Maurizio Rafanelli, "Hierarchies and Relative Operators in the OLAP Environment", *ACM SIGMOD Record*, vol.29(1), ACM Press, p. 32–37, 2000.

[Pourrabas & Rafanelli, 2003] Elaheh Pourabbas, Maurizio Rafanelli, "Hierarchies", Chapitre IV, *Multidimensional Databases: Problems and Solutions*, Maurizio Rafanelli (Ed.), Idea Publishing Group (IGP), ISBN 1-59140-053-8, p. 91–115, 2003.

[Ravat et al., 2007e] Franck Ravat, Olivier Teste, Ronan Tournier, Gilles Zurfluh, "Algebraic and graphic languages for OLAP manipulations", *Intl. Journal of Data Warehousing and Mining* (*ijDWM*), Idea Group Publishing (IGP), juin 2007 (à paraître).

[Rizzi et al., 2006] Stefano Rizzi, Alberto Abelló, Jens Lechtenbörger, Juan Trujillo, "Research in data warehouse modeling and design: dead or alive?", *9th ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 3–10, 2006.

[Schneider, 2003] Michel Schneider, "Well-formed data warehouse structures", *5th Intl. Workshop on Design and Management of Data Warehouses* (*DMDW*), CEUR Workshop Proceedings vol.77, CEUR-WS.org, p. 2.1–2.13, 2003.

[Schneider, 2007] Michel Schneider, "A general model for the design of data warehouses", *Intl. Journal of Production Economics*, Elsevier, disponible en ligne, 2007 (à paraître).

[Stefanovic et al., 2000] Nebojsa Stefanovic, Jiawei Han, Krzysztof Koperski, "Object-Based Selective Materialization for Efficient Implementation of Spatial Data Cubes", IEEE Transactions on Knowledge and Data Engineering (TKDE), vol.12(6), IEEE Computer Society, p. 938–958, 2000.

[Sullivan, 2001] Dan Sullivan, *Document Warehousing and Text Mining*, Wiley John & Sons, ISBN: 0471399590, 2001.

[Torlone, 2003] Riccardo Torlone, "Conceptual Multidimensional Models", Chapitre III, *Multidimensional Databases: Problems and Solutions*, Maurizio Rafanelli (Ed.), Idea Publishing Group (IGP), ISBN 1-59140-053-8, p. 69–90, 2003.

[Tseng & Chou, 2006] Frank S.C. Tseng, Annie Y.H. Chou, "The concept of document warehousing for multi-dimensional modeling of textual-based business intelligence",

journal of Decision Support Systems (*DSS*), vol.42(2), Elsevier, p. 727–744, novembre 2006.

CHAPITRE IV
XML OLAP : langage de manipulation multidimensionnelle

Résumé

Ce chapitre présente nos propositions pour le niveau de restitution et d'analyse de l'architecture des systèmes d'aide à la prise de décision. Afin de permettre des analyses multidimensionnelles sur des données textuelles, le chapitre commence par la présentation d'une fonction d'agrégation permettant d'agréger des données textuelles. Cette fonction emploie une ontologie de domaine pour permettre l'agrégation de mots-clef entre eux. Le chapitre se poursuit sur la présentation des opérations de manipulation qui permettent la manipulation des concepts représentés par le modèle en galaxie. Ces opérations autorisent premièrement la spécification d'analyses multidimensionnelles et deuxièmement, l'affinage de ces analyses par un noyau d'opérations de manipulations. Ces opérations emploient les liens spécifiés au sein du modèle en galaxie, permettant l'expression d'analyses complexes.

SOMMAIRE

CHAPITRE IV : XML OLAP, langage de manipulation multidimensionnelle

"Tout notre savoir trouve son origine dans nos perceptions."

— Léonard de Vinci.

1 Introduction

Le modèle conceptuel du chapitre précédent offre une représentation destinée au décideur qui lui permet de visualiser les structures multidimensionnelles disponibles pour des analyses.

Ce chapitre présente le langage de manipulation qui permet la manipulation des structures multidimensionnelles par l'intermédiaire d'opérations. Par rapport à l'architecture des systèmes d'aide à la décision, le jeu d'opérations de manipulation se situe au quatrième niveau de notre architecture (cf. Figure 43) ; au niveau restitution et analyse.

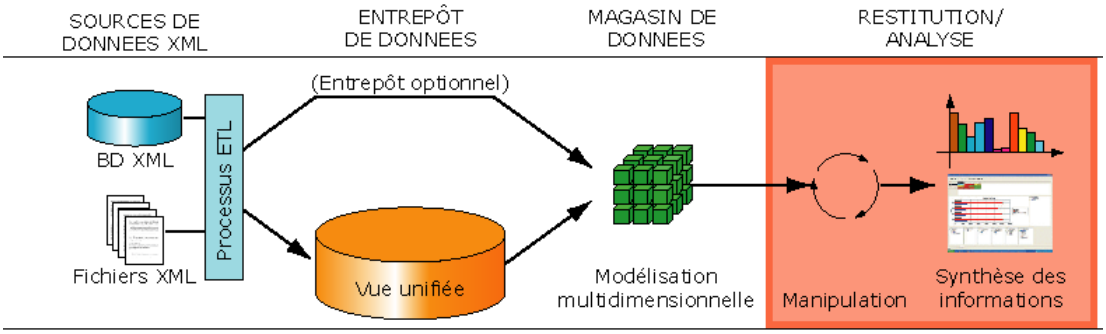


Figure 43 – Positionnement de la manipulation dans l'architecture générale.

À partir de la représentation conceptuelle, le décideur utilise un langage pour manipuler les structures afin de spécifier et de faire évoluer les analyses (cf. Figure 44). Ce langage de manipulation est adapté à la représentation conceptuelle des structures et permet une manipulation simple des concepts pour faciliter l'expression d'analyses et leur affinage.

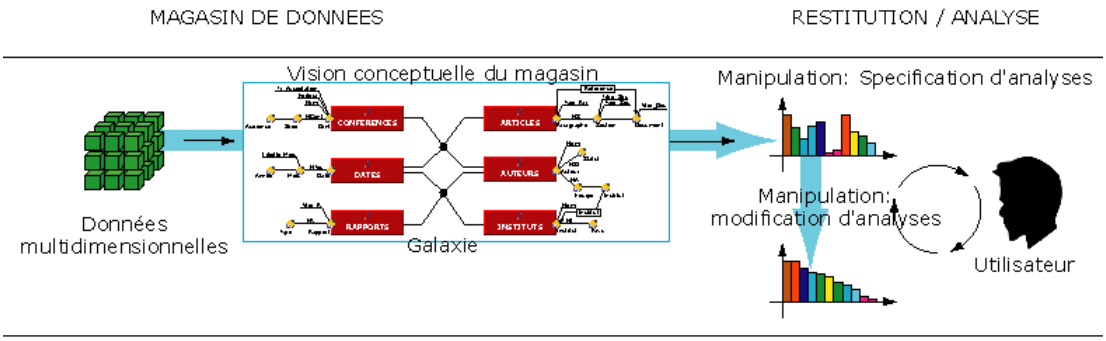


Figure 44 – Spécification et manipulation d'analyses.

La spécification et la manipulation d'analyses multidimensionnelles fait massivement intervenir des fonctions d'agrégation pour synthétiser les informations. Ces fonctions d'agrégation sont très efficaces dans l'environnement classique pour synthétiser des valeurs

numériques. Toutefois l'analyse de données issues de documents XML pose des problèmes car les données à analyser sont majoritairement constituées de texte. C'est ainsi que dans un premier temps nous allons proposer une fonction d'agrégation adaptée aux données de notre environnement.

Le chapitre se décompose en deux parties. La première partie expose l'agrégation de données textuelles au moyen de fonctions. La seconde partie, quant à elle, présente les opérations qui permettent la spécification et la manipulation d'analyses multidimensionnelles spécifiées à partir des concepts modélisés par une galaxie.

2 Agrégation et données textuelles

L'analyse multidimensionnelle repose sur une capacité à résumer et synthétiser des informations très volumineuses. Les données analysées sont agrégées au moyen de fonctions d'agrégation. Mais l'environnement OLAP ne fournit pas de fonction d'agrégation adaptée aux données textuelles. Ainsi pour permettre cette synthétisation d'informations, il est nécessaire de définir des fonctions d'agrégation adaptées.

2.1.1 Principe de l'agrégation

L'analyse multidimensionnelle en ligne (OLAP) expose des données sélectionnées en tant que sujet selon différents niveaux de détails ou niveaux de granularité. Le processus agrège les données du sujet selon le niveau de détails avec des fonctions d'agrégation telles que somme, moyenne… Les opérations de forage, éléments essentiels dans la manipulation de données multidimensionnelles, permettent à l'utilisateur de modifier le niveau de détails des données analysées. Les données analysées sont alors agrégées selon un nouveau niveau de détail.

Dans la Figure 45 un décideur analyse le nombre d'articles par mois, en fonction d'auteurs. Afin d'obtenir une vision plus globale des données, le décideur change le niveau de détail pour avoir les mots-clef par année. On dit qu'il effectue un forage vers le haut (*roll-up*). En conséquence, les valeurs mensuelles sont agrégées en valeurs annuelles.

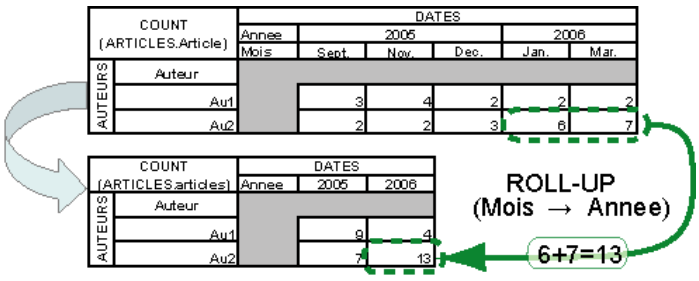


Figure 45 – Analyse multidimensionnelle de nombre de mots-clef par auteurs et par mois, puis par année (seuls les mois avec des données sont affichés).

Pour opérer sur des données non-numériques il est nécessaire de proposer des fonctions d'agrégation adaptées. Nous proposons de permettre l'agrégation de données textuelles. Cette approche a l'avantage de permettre la combinaison d'analyses quantitatives et qualitatives. Dans cette première partie du chapitre sur la manipulation, nous présentons une nouvelle fonction d'agrégation opérant sur des données textuelles et reposant sur un concept novateur, à savoir l'emploi d'une ontologie de domaine pour permettre un processus d'agrégation.

2.1.2 Agrégation et données textuelles

L'agrégation de données textuelles permet de résumer le volume des données à visualiser lors d'une même analyse. En réduisant ainsi le volume des données par des méthodes de synthèse, l'utilisateur peut avoir une vision plus globale du domaine qu'il analyse. Le modèle conceptuel en galaxie présenté dans le chapitre précédent permet de modéliser des données textuelles au sein d'un environnement OLAP. Ainsi pour permettre les anlayses des attributs documentaires présentés précédemment, il est nécessaire de disposer d'une méthode de synthétisation adaptée.

Les fonctions d'agrégation présentées dans la suite sont des fonctions opérant sur des mots-clef. Ces mots-clef peuvent avoir été extraits du texte lors de l'alimentation des documents dans le magasin de données ou bien ils peuvent être extraits à la volée lors de l'analyse par une fonction d'agrégation opérant sur des fractions de texte. Dans [Park et al., 2005], les auteurs présentent plusieurs fonctions d'agrégation sans les définir :

- TOP_KEYWORD est une fonction qui extrait les *n* mots-clef jugés les plus pertinents d'un fragment de texte.
- TOPIC est une fonction qui extrait le sujet d'un fragment de texte.
- SUMMARY est une fonction qui génère un résumé automatique d'un fragment de texte.
- COUNT et LIST : sont deux fonctions d'agrégation génériques opérant sur tout type de données (agrégeables ou non). COUNT compte le nombre d'instances (le nombre de fragments) et LIST effectue la liste complète des valeurs sans les agréger.

Les deux premières fonctions génèrent des mots-clef. Notre idée est de fournir une fonction qui s'inspire de la moyenne pour permettre d'agréger les mots-clef en des mots-clef plus généraux. Par exemple, les mots-clef *Java* et *C++* seraient regroupés en *programmation objet*.

Toutefois, pour permettre une telle agrégation, il est nécessaire de disposer d'un moyen de « calculer » et de trouver les mots-clef moyens. Une ontologie légère est employée à cette fin.

Plan de la partie concernant la fonction d'agrégation. La section 2.2 défini une règle d'agrégation sur laquelle va s'appuyer notre fonction d'agrégation. Il s'agit d'une ontologie accompagnée d'opérations qui seront employées par la suite pour permettre l'agrégation de mots-clef. La section 2.3 expose notre approche concernant les mots-clef. La section 2.4 définit une fonction d'agrégation basée sur l'ontologie qui permet d'agréger plusieurs mots-clef en un (ou plusieurs) mot-clef « moyen » au moyen d'une ontologie. Enfin la section 2.5 présente un exemple de l'emploi de la fonction d'agrégation.

2.2 Règle d'agrégation : ontologie légère et opérations

Afin de permettre l'observation d'indicateurs d'analyse textuels, il est nécessaire de fournir une loi qui permet de calculer une moyenne tout comme la formule dans un environnement numérique. Pour établir cette règle, nous utilisons une représentation hiérarchique de concepts de domaine.

2.2.1 Ontologie légère : définition

Ces concepts sont modélisés au moyen d'une ontologie « légère » ou encore d'une ontologie dotée de liaisons « est-un informelles » ("informal is-a ontology") [Lassila & McGuiness, 2001]. Ce type d'ontologie correspond à une hiérarchie de concepts d'un domaine où chaque

nœud représente un concept (un mot-clef) et chaque lien entre les nœuds modélise une relation plus complexe que la relation « est-un ».

Définition. Étant donnée une ontologie *O*, le *domaine* de *O*, noté *dom*(*O*), est l'ensemble des mots-clef de *O*. *dom*(*O*) est un ensemble de mots-clef uniques (donc sans homographes).

Exemple. Dans l'ontologie des concepts des systèmes d'information *O_IS*, le concept *OLAP* ∈ *dom*(*O_IS*) (cf. Figure 46). Cette ontologie est composée d'un ensemble de concepts pouvant avoir des homonymes.

Définition. La *profondeur* (*depth*) d'une ontologie est le nombre maximum de nœuds entre le nœud racine et les feuilles.

Exemple. Toujours dans le même exemple (cf. Figure 46), la profondeur est : *depth*(*O_IS*) = *8*.

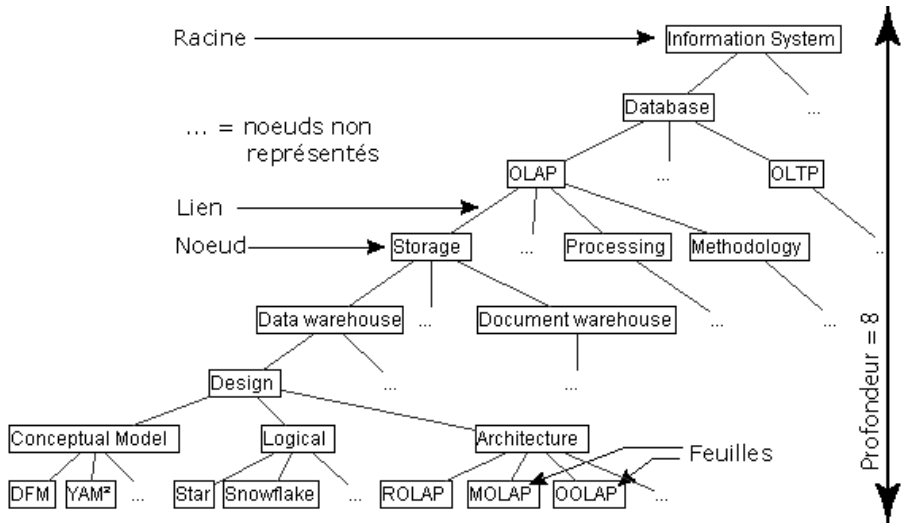


Figure 46 - Exemple d'une ontologie de domaine sur les systèmes d'information: *O_IS*.

2.2.2 Ontologie : opérations

Pour permettre l'emploi de l'ontologie lors du processus d'agrégation, nous définissons deux opérations qui permettent 1) de trouver le plus petit ancêtre commun parmi les nœuds subsumant deux nœuds et 2) de calculer la distance entre ces deux noeuds. Ces opérations prennent en entrée deux nœuds de l'ontologie, c'est-à-dire deux mots-clef.

Définition. Le plus petit ancêtre commun (*Least Common Ancestor*, *lca*), cf. Eq. 3, est une fonction qui retourne le nœud représentant le plus petit commun ancêtre (n_{LCA}) au sein de *O* entre n_1 et n_2.

$lca : \begin{matrix} (dom(O))^2 \\ (n_1, n_2) \end{matrix} \begin{matrix} \longrightarrow \\ \mapsto \end{matrix} \begin{matrix} dom(O) \\ n_{LCA} \end{matrix}$ où $(dom(O))^2 = dom(O) \times dom(O)$	**Eq. 3**

Définition. La *Distance* entre deux nœuds (*d*), cf. Eq. 4, est une fonction qui retourne le nombre d'arcs entre le nœud représentant le plus petit ancêtre commun (n_{LCA}) et le nœud le plus bas dans la hiérarchie entre n_1 et n_2. Concrètement, il s'agit d'une mesure de similarité, à savoir une distance structurelle calculant un nombre d'arcs.

$d : \begin{array}{ccc} (dom(O))^2 & \longrightarrow & N \\ (n_1, n_2) & \mapsto & \max(d(n_1, lca(n_1, n_2)), d(n_2, lca(n_1, n_2))) \end{array}$	**Eq. 4**

Exemple. Par exemple, dans *O_IS* (cf. Figure 46), le plus petit ancêtre commun entre *ROLAP* et *Document Warehouse* est *Storage* :

lca(*ROLAP*, *Document Warehouse*) = *Storage*.

La distance entre ces deux mots-clef est de *4* :

d(*ROLAP*, *Document Warehouse*) =
max(*d*(*ROLAP*, *Storage*), *d*(*Document Warehouse*, *Storage*))
= *max* (*4*, *1*) = *4*

2.3 Exploitation d'un attribut de type mot-clef

Lors de l'agrégation de mots-clef en des mots-clef plus généraux, une perte de précision, en terme de sémantique, est à prévoir. Par exemple, dans l'ontologie présentée précédemment, le terme *Database* est beaucoup plus général que le terme *ROLAP*. Il est donc nécessaire de suivre l'évolution de cette agrégation afin de limiter la perte de sens liée à l'agrégation.

Dans notre environnement, un mot-clef exploitable par notre fonction de moyenne est un doublet composé d'une chaîne de caractères qui représente le mot-clef proprement dit et d'une valeur numérique pour évaluer la perte de sens au fur et a mesure de la recherche de mots-clef de plus en plus généraux.

Définition. Un *mot-clef agrégeable* est un couple $x=(kw, d)$ où *kw* est un mot-clef présent en tant que nœud dans l'ontologie $kw \in O$ et d est une distance, $d=0$ initialement.

L'ensemble des mots-clef agrégeables est représenté par : $X = dom(O) \times \mathbb{N}$.

Afin de pouvoir revenir sur des mots-clef constitués d'une simple chaîne de caractère nous redéfinissons la fonction générique LIST. Cette fonction permet l'obtention des mots-clef sans les distances *d*.

Définition. La fonction générique d'agrégation, *LIST*, génère à partir d'une liste de mots-clef agrégeables, la même liste de mots-clef (sans effectuer d'agrégation) privée des distances associées à chaque mot-clef (cf. Eq. 5).

$LIST : \begin{array}{ccc} X^n & \longrightarrow & (dom(O))^n \\ (x_1, \ldots, x_n) & \mapsto & (kw_1, \ldots, kw_n) \end{array}$ où $X = dom(O) \times \mathbb{N}$	**Eq. 5**

Les mots-clef peuvent être issus d'attributs spécifiques contenant un unique mot-clef ou une liste de mots. Mais il est aussi envisageable de disposer d'une source de mots-clef à la volée. Ainsi, en employant par exemple la fonction TOP_KEYWORDS [Park et al., 2005], il est possible d'extraire d'un bloc de texte les n principaux mots-clef. La sortie de cette fonction, une fois chaque mot-clef associé à une distance (nulle par défaut), est une entrée possible pour notre fonction de moyenne de mots-clef.

2.4 Fonction d'agrégation de mots-clef : *AVG_KW*

La fonction d'agrégation *AVG_KW* est conçue pour synthétiser un ensemble de mots-clef issus d'un vocabulaire contrôlé en un ensemble plus petit de mots-clef plus généraux. La fonction prend en entrée un ensemble de mots-clef, chacun associé à une distance et génère un nouvel ensemble de mots-clef agrégés. Le processus d'agrégation se base sur l'ontologie de domaine définie précédemment. Les mots-clef sont tous issus du vocabulaire contrôlé représenté par l'ontologie et le domaine de l'ontologie est proche du domaine des documents à analyser.

Pour chaque paire de mots-clef, la fonction trouve le plus petit ancêtre commun (lca) correspondant. Mais lors de l'agrégation de mots-clef très éloignés dans l'ontologie, il y a une très forte probabilité de retourner systématiquement le mot-clef représenté par le nœud racine de l'ontologie. Afin d'éviter ce phénomène, une limite dans le processus d'agrégation doit être imposé. En effet, plus les mots-clef sont éloignés les uns des autres, plus l'agrégation se traduit par une perte de sens. Pour surmonter ce problème, la fonction emploie une distance maximale autorisée lors de l'agrégation de mots-clef : D_{MAX}. Pour l'instant, des heuristiques informelles suggèrent une distance comprise entre 3 et 5. Il est à noter qu'avec une ontologie généraliste telle que WordNet[15] D_{MAX} est plus de l'ordre de 3 [Baziz, 2005]. A ce jour et à notre connaissance, dans la recherche concernant les ontologies, ce problème n'a pas encore été résolu.

Pour visualiser les résultats, nous employons une table bidimensionnelle permettant la visualisation d'un sujet d'analyse (une dimension focalisée) et de deux dimensions en tant qu'axes d'analyse [Gyssens & Lakshmanan, 1997] et [Ravat et al., 2007e]. Pour chaque combinaison des valeurs des axes d'analyse, la table contient une cellule correspondant à un ensemble de valeurs. La fonction *AVG_KW* prends en entrée chaque cellule (un ensemble de mots-clef à chaque fois) et produit en sortie un nouvel ensemble de mots-clef. Le nouvel ensemble est composé de mots-clef agrégés ou de mots-clef de la cellule d'origine si le processus d'agrégation échoue suite à une distance excessive entre mots-clef.

2.4.1 Définition formelle

Les paragraphes qui suivent présentent la définition formelle de la fonction d'agrégation.

[15] WordNet : Ontologie lexicale anglaise disponible sur http://wordnet.princeton.edu/

Définition. La fonction d'agrégation moyenne de mots clefs est définie par (Eq. 6) :

$AVG\_KW : \quad X^n \longrightarrow X^m$ $(x_1,\ldots,x_n) \mapsto (y_1,\ldots,y_m)$ avec $m \le n$ et $X = dom(kw) \times \mathbb{N}$	**Eq. 6**

- $(x_1,\ldots,x_n) \in X^n$ est un ensemble ordonné de mots-clef tels que $\forall x_i \in X, x_j \in X \mid i<j, d(x_i, x_{ROOT}) \le d(x_j, x_{ROOT})$. C'est-à-dire les nœuds les plus éloignés de la racine sont en premier et $x_i=(kw_i,d_i)$ avec $kw_i \in dom(O)$ et $d_i \le D_{MAX}$ ($d_i=0$ s'il n'y a eu aucune agrégation auparavant).
- $(y_1,\ldots,y_m) \in X^m$ est un ensemble de mots-clef agrégés. Ces mots-clef sont agrégés par une fonction conditionnelle basée sur le lca (Eq. 7).

$(x_i,x_j) \mapsto \begin{cases} x_{LCA} = (kw_{LCA}, l(x_i,x_j)) \text{ if } l(x_1,x_2) \le D_{MAX} \\ (x_i,x_j) \text{ otherwise} \end{cases}$ Où $l(x_i,x_j) = d(kw_i,kw_j) + d_i + d_j$ et $kw_{LCA} = LCA(kw_i,kw_j)$	**Eq. 7**

Si x_i et x_j sont agrégés en x_{LCA} alors x_i et x_j sont retirés de l'ensemble d'entrée X et x_{LCA} est ajouté à X. Le processus d'agrégation est itéré sur X jusqu'à ce qu'aucune autre agrégation n'ait pu avoir lieue (Eq. 8) :

$\forall (x_i, x_j) \in X^2, \nexists x_{LCA} \mid l(x_i,x_j) \le D_{MAX}$	**Eq. 8**

Remarque. Pour un y_k donné de X, si $d_k=0$, alors le mot-clef correspondant kw_k n'a pas été agrégé durant le processus ($\exists x_i \in X \mid x_i=y_k$). De plus, si $\forall x_i, x_j \in X$, $l(x_i, x_j)>D_{MAX}$, alors il n'y pas d'agrégation possible et $(y_1,\ldots,y_m) = (x_1,\ldots,x_n)$ avec $m=n$.

Basé sur cette définition formelle la sous-section suivante présente l'algorithme de la fonction.

2.4.2 Algorithme

L'algorithme prend en entrée une liste de mots-clef qui doivent être agrégés : `KW_LIST`={kw_1, kw_2,…, kw_n} et une ontologie `O`. Il produit en sortie un ensemble de mots-clef agrégés : `Output_List`.

Une fonction précède l'algorithme et initialise la liste des mots-clef si nécessaire. En effet si la liste en entrée est une liste composée uniquement de mots-clef, cette liste est transformée en une liste de mots-clef agrégeables. La fonction associe à chaque mot-clef un distance valant 0.

L'algorithme emploie les fonctions et procédures suivantes :

- `d(keyword`$_1$`, keyword`$_2$`)` est une fonction qui calcule la distance entre les deux mots-clef `keyword`$_1$ et `keyword`$_2$.
- `Order_List` est une fonction qui ordonne une liste de mots-clef telle que $d(kw_i, kw_{ROOT}) \le d(kw_j, kw_{ROOT})$. C'est-à-dire que les mots-clef sont ordonnés selon le niveau de leur nœud qui les représente dans `O`, en commençant par les niveaux les plus bas (les nœuds les plus éloignés de la racine).

- `LCA` est une fonction qui recherche le plus petit ancêtre commun entre une paire de mots-clef dans un arbre. Nous invitons le lecteur à consulter [Harel & Tarjan, 1984] et plus récemment [Bender & Farach-Colton, 2000] pour une discussion poussée quant à l'implantation de la résolution du problème du LCA.

Algorithme :

```
{KW_List = OrderList(KW_List,O);
 For each KWi of KW_LIST Do
    li = 0;
    For each KWj of KW_List, (j>i) Do
        KWLCA=LCA(KWi,KWj) ;
        lLCA=MAX(d(KWi,KWLCA),d(KWj,KWLCA))+li
        If ( lLCA ≤ DMAX ) Then
            KW_List=KW_List-{KWi, KWj};
            KWi=KWLCA; li=lLCA;
        end_If;
    end_For;
    Add KWi to Output_List;
 end_For;
}
```

L'algorithme génère une liste `Output_List` en sortie.

2.5 Exemple d'analyse

L'emploi d'opérations de forage utilise de manière intensive les fonctions d'agrégations. Ainsi, l'opération de forage vers le haut sera employée pour l'exemple qui suit. Cette fonction sera présentée plus en détails dans la seconde partie de ce chapitre en tant qu'opération de forage.

Le Tableau 7 présente un jeu de données de test qui sera employé au cours de l'exemple. Trois documents ont été sélectionnés et tous trois ont été écrits par le même auteur (*Au1*) à trois dates différentes. Deux mots-clef ont été extraits du contenu de chaque document.

Tableau 7- Exemple de données : trois documents et leurs mots-clef associés.

Documents	Keywords	Date	Author
Doc_1	Document Warehouse Algebra	Nov. 2004	Au_1
Doc_2	Data Warehouse Conceptual Model	Sept. 2004	Au_1
Doc_3	Logical Fact Table	Sept. 2004	Au_1

La Figure 47 montre les positions des différents mots-clef présentés dans le Tableau 7, au sein de l'ontologie qui nous sert d'exemple (*O_IS*). Ces mots-clef sont entourés d'un rectangle.

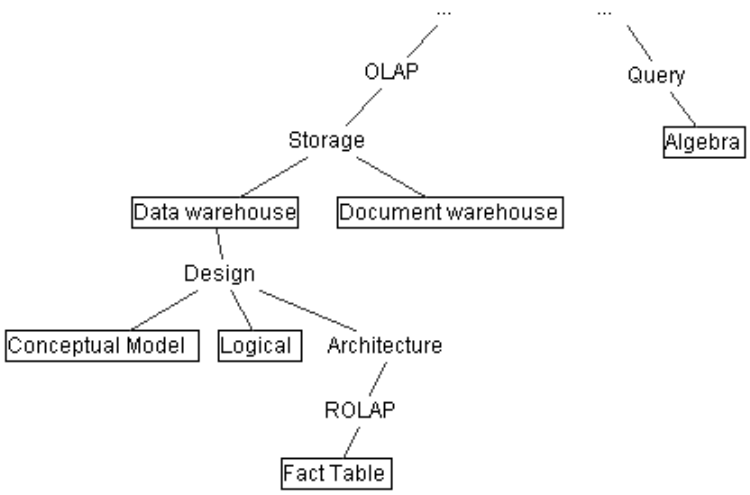


Figure 47 – Le positionnement des différents mots-clef dans l'ontologie de domaine *O_IS* (représenté partiellement).

La Figure 48 montre les agrégations possibles des mots-clef extraits des documents. Les flèches indiquent les agrégations avec les distances que l'agrégation représente. Il s'agit de la distance entre les nœuds de l'ontologie. Pour cet exemple, la distance maximale d'agrégation a été fixée à 3 nœuds : $D_{MAX} = 3$.

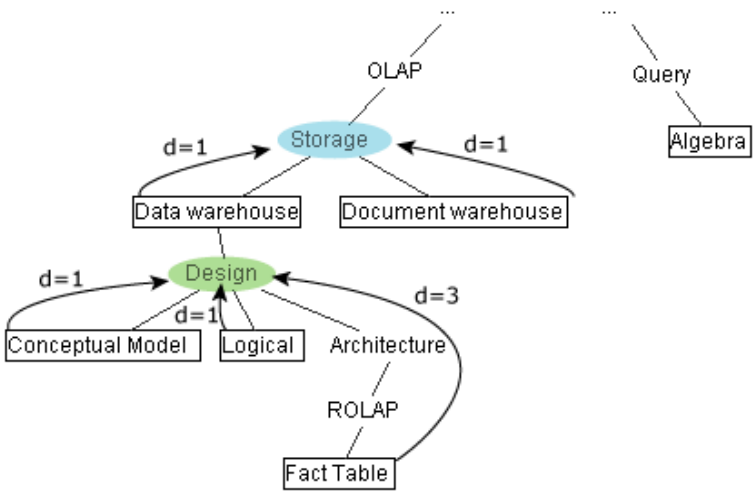


Figure 48 – Agrégations possibles de mots-clef des documents au sein de l'ontologie.

Dans la Figure 49, un décideur analyse les publications de l'auteur *Au1* durant l'année *2004*. Il affiche les résultats de l'analyse par *Mois* (partie supérieure (a) de la Figure 49). Les mots-clef des deux publications du mois de *septembre* sont agrégés : trois d'entre eux (*Fact table*, *Conceptual model* et *Logical*) sont agrégés en un seul (*Design*), représenté en vert (gris clair) sur la figure. Le quatrième par contre est trop éloigné des autres et ne peut être agrégé. Concernant l'autre document (celui du mois de *novembre*) les deux mots-clef qui en ont été extraits sont trop éloignés l'un de l'autre et ne peuvent être agrégés.

Par la suite, le décideur effectue un forage vers le haut afin d'obtenir une vision plus globale. Le forage se fait selon la dimension *DATES* et le niveau de détail des données passe de Mois à Année. Ainsi toutes les publications analysées sont regroupées en un seul groupe

correspondant aux publications de *Au1* durant l'année *2004*. La fonction AVG_KW tente d'agréger l'ensemble des mots-clef.

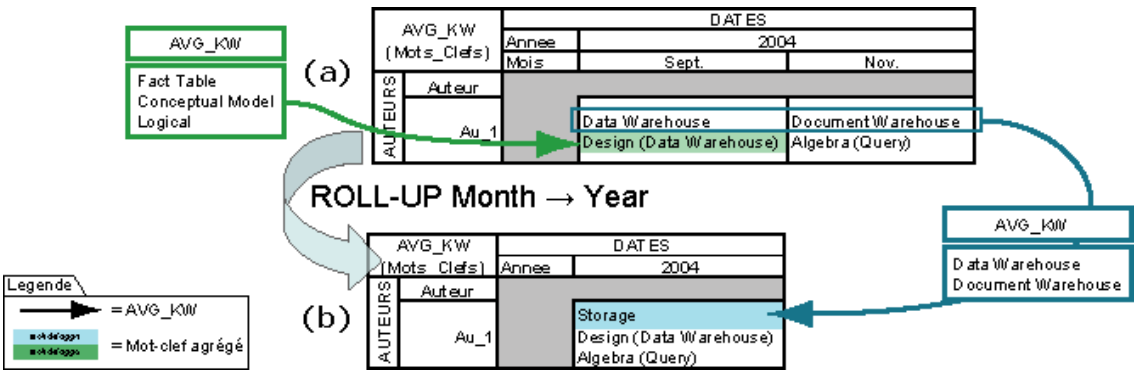


Figure 49 – (a) Analyse des mots-clef par mois, puis (b) par années après un forage vers le haut sur la dimension DATES.

La vision par mois dans une table dimensionnelle génère deux cellules, une correspondant aux publications du mois de septembre et une autre correspondant aux publications du mois de novembre :

c_1={*Doc_2*, *Doc_3*} pour le mois de septembre ;
c_2={*Doc_1*} pour le mois de novembre.

Ces deux ensembles de documents correspondent aux mots-clef agrégeables suivants (mots-clef associés à une distance valant *0* par défaut) :

x_3=(kw_3='Data Warehouse', d_3=0) ; x_4=(kw_4='Conceptuel Model', d_4=0) ;
x_5=(kw_5='Logical', d_5=0) ; x_6=(kw_6='Fact Table', d_6=0) ;
x_1=(kw_1='Document Warehouse', d_1=0) ; x_2=(kw_2='Algebra', d_2=0) ;

Le processus d'agrégation s'opère pour chaque cellule de la table de manière indépendant en commençant par agréger les mots-clef les plus éloignés de la racine :

$AVG\_KW(x_3, x_4, x_5, x_6) = (x_3, x_7)$ avec $x_7 = (kw_7$='Design', $d_7=3)$
$AVG\_KW(x_1, x_2) = (x_1, x_2)$ car $d(x_1, x_2) > D_{MAX}$

Lors du forage, le processus est réitéré :

$AVG\_KW(x_1, x_2, x_3, x_7) = (x_2, x_7, x_8)$ avec $x_8 = (kw_8$='Storage', $d_8=1)$

Ainsi après agrégation l'ensemble des six mots-clef du départ se résume à trois mots-clef plus généraux : 'Algebra', 'Design et 'Storage'.

2.6 Bilan concernant l'agrégation

Dans l'environnement multidimensionnel OLAP, les travaux de Byung-Kwon Park [Park et al., 2005] proposent l'emploi d'une fonction d'agrégation textuelle (TOP_KEYWORD) qui agrège un fragment de texte en une liste de *n* mots-clef. Cette liste peut être très grande et donc la synthèse d'information induite par l'emploi de la fonction d'agrégation n'est guère utile. La fonction d'agrégation AVG_KW, permet de synthétiser des mots-clef d'un vocabulaire contrôlé (mots-clef d'un même domaine). Ainsi contrairement à la fonction proposée par Park [Park et al., 2005], le risque de se retrouver avec une liste trop importante de mots est réduit.

Comme le montre la fonction AVG_KW, il est possible de baser l'agrégation non plus sur des lois mathématiques, telle que la somme, mais sur une autre méthode pour agréger les données. Dans notre cas, nous avons reconstitué une « loi d'agrégation » à partir d'une ontologie. Néanmoins, la fonction décrite est une première étape dans l'analyse de données issues de documents.

Durant les exemples qui ont présenté la fonction d'agrégation, il a été fait mention d'une opération de forage : Roll-Up. Cette opération fait partie des opérations de manipulation qui sont décrites de manière détaillée dans la seconde partie de ce chapitre.

3 Manipulation multidimensionnelle

Les concepts multidimensionnels disponibles pour exprimer des analyses sont présentés à l'utilisateur au moyen des formalismes graphiques du modèle conceptuel. Les éléments de ce modèle sont manipulés avec l'aide d'opérations pour permettre :

- la spécification d'analyses multidimensionnelles ;
- la manipulation d'une analyse en cours.

Cette seconde partie du chapitre vise à définir un ensemble d'opérateurs de manipulation associé à la galaxie permettant la spécification d'analyses.

Les opérations de manipulation ont deux buts : premièrement, elles doivent permettre la *spécification d'analyses multidimensionnelles* à partir du concept de dimension du modèle en galaxie ; deuxièmement, elles doivent permettre la *modification d'analyses en cours* pour affiner les résultats qui permettront au décideur d'avoir la meilleure interprétation des données qu'il observe.

La proposition [Ravat et al., 2007e] offre un support de l'ensemble des catégories d'opérations de manipulation des bases de données multidimensionnelles. Cette proposition est basée sur un modèle en constellation et une structure de visualisation de table multidimensionnelle. Inspiré de ces travaux, ce chapitre présente les principaux opérateurs de manipulations redéfinis pour permettre la manipulation des concepts multidimensionnels modélisés par la galaxie.

Les opérations, qui étaient jusqu'alors spécifiques soit aux structures de dimensions soit aux structures de faits, sont désormais généralisées grâce à l'unique concept de la galaxie.

Plan de la partie 2. La section 3.1 expose la spécification des opérations multidimensionnelles qui seront présentées dans ce chapitre. Il s'agit des notations formelles employées et de la spécification des entrées/sorties des différentes opérations. La section 3.2 présente l'opération de spécification d'une analyse ainsi que l'emploi des liens du modèle pour faciliter la navigation au sein des données. La section 3.3 définit les différentes opérations qui permettent la manipulation d'une analyse une fois celle-ci spécifiée.

3.1 Cadre de spécification des opérateurs

La galaxie, modèle basé sur l'unique concept de dimension est un modèle qui généralise de nombreuses opérations précédemment présentées. L'avantage de cette généralisation est une spécification plus simple des opérations.

3.1.1 Introduction et objectifs

Afin de manipuler les concepts représentés dans le modèle en galaxie, le décideur a besoin d'une opération pour la spécification d'analyses puis d'un ensemble d'opérations pour lui permettre de modifier ces analyses. Les besoins en opérations sont résumés dans la liste suivante :

- Une opération de focalisation (Focus) est nécessaire pour mettre en avant le sujet d'une analyse, projetant les données du sujet sur plusieurs axes d'analyse et permettant la spécification d'une analyse.
- Pour restreindre la portée d'une analyse, une opération est nécessaire pour ne sélectionner qu'un sous-ensemble des données.
- Pour exploiter l'organisation hiérarchique des paramètres, deux opérations de forage sont nécessaires pour permettre de modifier le niveau de détail des données analysées : la première pour zoomer et explorer plus en détails les données ; la seconde pour « dézoomer » et permettre l'inverse, une exploration plus globale.
- Pour changer de critère d'analyse, une opération est nécessaire pour réorienter l'analyse, à savoir, effectuer un changement de sujet ou bien d'axe d'analyse.

Les opérations de manipulation doivent aussi prendre en compte les spécificités du modèle conceptuel en galaxie :

- la navigation au sein des données par l'intermédiaire des liens ;
- l'exploitation de la hiérarchisation des paramètres mis en avant en tant que sujet d'analyse ;

Les auteurs de certains modèles ont souligné la nécessité du traitement symétrique des paramètres et des indicateurs d'analyse (mesures) pour faciliter la définition d'algèbres de requête ou de langage de calcul ainsi que pour introduire plus de flexibilité pour l'utilisateur [Agrawal et al., 1997], [Cabibbo & Torlone, 1997] et [Gyssens & Lakshmanan, 1997]. Pour permettre cette symétrie, les auteurs employèrent un artifice : des opérations de changement de structure (PUSH/PULL, FOLD/UNFOLD). Cependant la grande majorité des opérations telles que les forages ou encore les rotations ne pouvaient opérer de manière symétrique entre tout type d'attributs. En effet en l'absence de hiérarchies au sein des faits, les opérations de forages, effectives sur des paramètres hiérarchisés, étaient incompatibles avec des mesures ne disposant pas cette structure. Le modèle en galaxie, modélisant un sujet au moyen d'une dimension et les indicateurs d'analyse au moyen de paramètres et d'attributs faibles résout définitivement ce problème. Ceci permet :

- une simplification des opérations ;
- une spécification plus simple des opérateurs.

Afin de permettre la spécification des opérations nous introduisons dans le passage suivant les notations formelles qui seront employées.

3.1.2 Notations formelles

Nous introduisons quelques notations concernant la spécification formelle des valeurs des instances qui composent les éléments d'une galaxie.

Notations. $dom(D_i)$ est le domaine de la dimension D_i (tout $i_x \in I^{Di}$), c'est-à-dire, l'ensemble des instances de la dimension D_i. Nous notons $(dom(D_i))^*$ un ensemble fini d'éléments de $dom(D_i)$.

Nous introduisons aussi les quatre expressions qui permettront une simplification dans la formalisation des opérations de manipulation. Les instances d'une galaxie *G*, composée de *n* dimensions, sont représentées par Eq. 9. Ainsi toutes les instances qui composent une galaxie sont représentées par : *dom*(*G*).

$dom(D_1)\times\ldots\times dom(D_n)=\prod_{i=1}^{n} dom(D_i)=dom(G)$	**Eq. 9**

Toutes les instances d'attributs de $a_j \in A^{Di}$ d'une dimension D_i sont représentées par Eq. 10. Il est à noter que $\|A^{Di}\|$ est la cardinalité de l'ensemble A^{Di}, c'est-à-dire le nombre d'attributs de la dimension D_i, soit *r*, car $A^{Di} = \{a^{Di}_1,\ldots, a^{Di}_r\}$.

$dom(D_i.a_1)\times\ldots\times dom(D_i.a_r)=\prod_{j=1}^{\|A^{Di}\|} dom(D_i.a_j)=dom(D_i)$	**Eq. 10**

Nous définissons une fonction d'agrégation f_{AGG} (Eq. 11) où $dom(f_{AGG}(dom(D_i)))$ correspond au domaine des valeurs agrégées du domaine de la dimension D_i. Une fonction d'agrégation prend en entrée un ensemble de valeurs et génère en sortie une valeur agrégée.

$f_{AGG}: \begin{array}{ccc} (dom(D_i.p_j))^* & \longrightarrow & dom(f_{AGG}(dom(D_i.p_,))) \\ (x_1,\ldots,x_m) & \mapsto & f_{AGG}(x_1,\ldots,x_m) \end{array}$	**Eq. 11**

Afin de pouvoir comparer le niveau des paramètres au sein d'une même hiérarchie *H*, nous introduisons la fonction *level*. Étant donné :

$Param^H = <p_1,\ldots,p_{np}>$,
$level^H(p_1)=1$, $level^H(p_2)=2,\ldots$, $level^H(p_{np})=np$ et $\forall\ i<np-1$, $level^H(p_i)< level^H(p_{np})$

Pour permettre le chaînage des opérations, les opérations prennent en entrée et génère en sortie une structure compatible, présentée dans le passage suivant.

3.1.3 Entrée/Sortie des opérations

Toutes les opérations produisent des sorties compatibles permettant leur enchaînement, assurant la **fermeture** des opérations. L'opération de focalisation génère en sortie un sous-ensemble de la galaxie, noté s^G. Ce sous-ensemble est utilisé comme entrée pour toutes les autres opérations qui produisent à leur tour un sous-ensemble en sortie.

De manière formelle, cette structure multiaxe est spécifiée comme suit :

$s^G = (axe_S, axe_x, axe_y, axe_z,\ldots, res)$ avec :

- $axe_S = (DS, HS, Param_S)$ tel que la dimension *DS* est la dimension désignée en tant que sujet d'analyse, *HS* est la hiérarchie courante de *DS* et $Param_S = <f_{AGG}(a^{DS}_i), a_{S\_min},\ldots,a_{S\_max}>$
- $axe_x = (D_x, H_x, Param_x)$ tel que la dimension D_x est la dimension constituant le premier axe d'analyse, H_x est sa hiérarchie courante et $Param_x = <a_{x\_min},\ldots,a_{x\_max}>$
- $axe_y = (D_y, H_y, Param_y)$ tel que la dimension D_y est la dimension constituant le second axe d'analyse, H_y est sa hiérarchie courante et $Param_y = <a_{y\_min},\ldots,a_{y\_max}>$
- axe_z…
- …

- $res = (p_1 \wedge \ldots \wedge p_s)$ est une conjonction normalisée de prédicats restrictifs où chaque prédicat s'applique sur les valeurs d'un attribut d'une des dimension reliées au même nœud central que les dimensions spécifiées dans chaque axe.

Cette structure multiaxe est généraliste. Toutefois, la restitution à l'utilisateur se fera via une table bidimensionnelle. Cette interface permet la restitution de données sur deux axes. Ainsi, la structure de sortie est limitée à trois axes : $s^G = (axe_S, axe_x, axe_y, res)$ où axe_x et axe_y sont les deux axes de l'affichage bidimensionnel de la table (colonne, ligne).

La syntaxe générale des opérations est :

NOM_OPERATION(***entrée***, *paramètres_de_l_opération*) = ***sortie***

A partir des notations formelles et de la formalisation de l'entrée/sortie des opérations, la section suivante présente l'opération permettant la spécification d'une analyse multidimensionnelle.

3.2 Spécification d'analyses

À partir de la représentation conceptuelle en galaxie, le décideur peut spécifier des analyses multidimensionnelles par l'intermédiaire d'une opération de focalisation. L'utilisateur *focalise* l'analyse sur un sujet et *projette* les données du sujet sur plusieurs axes d'analyse. Les données projetées sont *agrégées* par une fonction d'agrégation pour en donner une vision synthétique.

Le décideur peut aussi employer les liens définis dans le modèle pour permettre la spécification d'analyses complexe. Sans l'emploi de ces liens, ces analyses seraient très difficiles voire impossible à exprimer.

Au sein de cette section, les opérations seront suivies d'un exemple applicatif. Ces exemples seront basés sur la galaxie G_1 (cf. Figure 50).

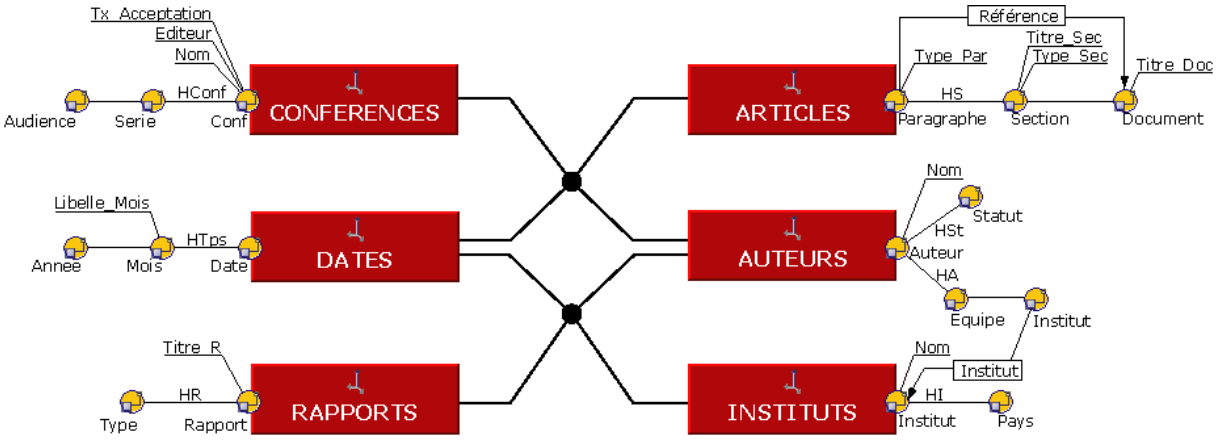


Figure 50 - Représentation graphique de la galaxie G_1.

Dans la suite, nous considérons que l'utilisateur emploie une table bidimensionnelle, extension de la table multidimensionnelle TM définie dans [Ravat et al., 2007e]. Il s'agit d'une structure adaptée aux décideurs [Gyssens & Lakshmanan, 1997]. La sortie compatible avec cette table est un sous-ensemble de la galaxie G_1 composée d'un sujet d'analyse (S), de deux axes d'analyses (x représentant les colonnes et y les lignes) et d'un ensemble de restrictions (res) :

$s^G = (axe_S, axe_x, axe_y, res)$ où :
$axe_S = (DS, HS, Param_S)$, $Param_S = \langle f_{AGG}(a^{DS}{}_i), a_{Sj} \rangle$
$axe_x = (D_x, H_x, Param_x)$, $Param_x = \langle a_{x\_min}, \ldots, a_{x\_max} \rangle$
$axe_y = (D_y, H_y, Param_y)$, $Param_y = \langle a_{y\_min}, \ldots, a_{y\_max} \rangle$
$res = (p_1 \wedge \ldots \wedge p_s)$

Pour plus de simplicité, la dénomination des axes sera simplifiée : axe_S désignant le sujet sera remplacé par S ; axe_x désignant les colonnes sera remplacé par C et axe_y par L (ligne).

Cette représentation impose une limitation sur la sortie : axe_S est défini tel qu'un seul paramètre puisse être associé à celui qui est agrégé via la fonction d'agrégation. Ainsi la liste ordonnée $Param_S$ est réduite à $\langle f_{AGG}(a^{DS}{}_i), a_{Sj} \rangle$.

Particularité de notre structure de sortie, l'emploi de la hiérarchisation des données désignée comme sujet d'analyse se traduit par la génération de « sous-groupes » de cellules dans la table bidimensionnelle de restitution. Ces groupes représentent une catégorisation des valeurs obtenues par $f_{AGG}(a^{DS}{}_i)$ en fonction de a_{Sj}. Ce principe sera vu plus en détail dans la section présentant l'opération de focalisation qui permet la spécification d'une analyse multidimensionnelle.

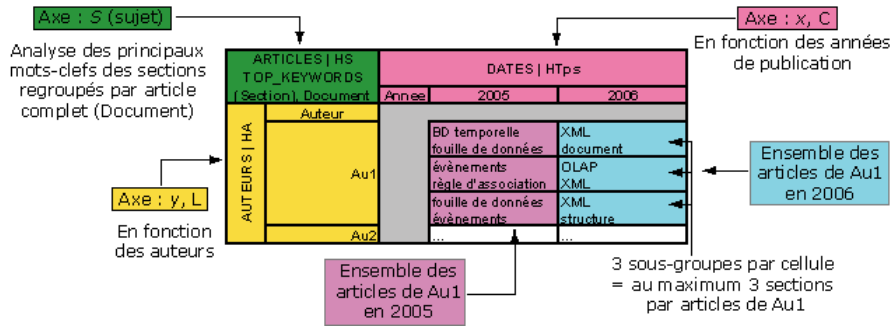


Figure 51 – Description de la table bidimensionnelle employée lors des exemples.

Exemple. La spécification de la table bidimensionnelle présentée en Figure 51 est la suivante :

$s^G = (S, C, L, res)$ avec :
$S = (ARTICLES, HS, \langle \text{TOP\_KEYWORDS}(Section), Document \rangle)$;
$C = (DATES, HTps, \langle Annee \rangle)$;
$L = (AUTEURS, HA, \langle Auteur \rangle)$;
$res = \varnothing$ (aucune restriction).

Cette table analyse le contenu d'articles, à savoir, les *principaux mots-clef par section par document* (S) en fonction de l'*auteur* (L) et des *années* (C). Cet exemple sera plus détaillé dans la section suivante pour présenter les sous-groupes de la table.

Les deux sous-sections suivantes présentent respectivement l'opération permettant la spécification d'analyses et la navigation au sein des données dimensionnelles via les liens.

3.2.1 Opération de focalisation

L'opération de focalisation est utilisée pour spécifier une analyse multidimensionnelle. Elle permet de sélectionner un sujet d'analyse et de projeter les données du sujet sur plusieurs axes d'analyse. Concrètement, cette opération permet la spécification d'un sujet d'analyse (*DS*)

agrégeant les données du sujet au moyen d'une fonction d'agrégation (f_{AGG}) selon le niveau de détail sélectionné dans les axes d'analyse.

Syntaxe : $FOCUS(G, S, P)=s^G$ où :

- G est l'entrée (une galaxie),
- $S=(DS, HS, Param_S)$ est le sujet d'analyse, tel que $Param_S=<f_{AGG}(p_i)[, p_{S\_min},\ldots,p_{S\_max}]>$. Ce sujet étant « focalisé » sur le paramètre p_i de la hiérarchie HS de la dimension DS dont les données sont agrégées par la fonction f_{AGG} (éventuellement en fonction d'une liste d'attributs optionnels : $< p_{S\_min},\ldots,p_{S\_max}>$)
- $P=<(D_x.H_x, Param_x),(D_y.H_y, Param_y),\ldots>$ est l'ensemble (ordonné) des axes de projection où D_x est la dimension sélectionnée en tant que premier axe d'analyse, D_y le second,… H_x est la hiérarchie courante de l'axe représenté par D_x, H_y est la hiérarchie courante de D_y,… $Param_x=<p_{x\_min},\ldots,p_{x\_max}>$ est un ensemble ordonné de paramètres de H_x (cet ensemble pouvant être un singleton), où étant donné $Param^{Hx}=<p_1,\ldots,p_{np}>$, $level^{Hx}(p_{x\_min})\geq level^{Hx}(p_1)$ et $level^{Hx}(p_{x\_max})\leq level^{Hx}(p_{np})$. $Param_x$ représente les paramètres sélectionnés de D_x (il s'agit d'un sous-ensemble de $Param^{Hx}$). Idem pour $Param_y$…

Remarque. Dans la suite de ce mémoire, $Param_S$ et P seront réduits dans les exemples pour l'emploi d'une table bidimensionnelle. Ainsi, $Param_S$ sera réduit à $<f_{AGG}(p_i), p_{Sj}>$ et P sera réduit à deux axes pour permettre la représentation des données dans une table bidimensionnelle : $P = <C, L>$ (colonnes et lignes de la table).

Conditions :

- $\forall D_i \in P$, $D_i \in Star^G(DS)$, c'est-à-dire les dimensions sélectionnées en tant qu'axe d'analyse sont liées à la dimension sélectionnée en tant que sujet (DS).
- La fonction d'agrégation f_{AGG} doit être compatible avec les instances à agréger du paramètre p_i.
- Si p_j est défini dans $Param_S$, alors $level^{HS}(p_i)<level^{HS}(p_j)$

Mathématiquement :

$FOCUS$ (cf. Eq. 14) = $AGGREGATE$ (cf. Eq. 13) $\circ PROJECT$ (cf. Eq. 12)

Où, chaque partie est décrite par les spécifications suivantes :

$\prod_{i=1}^{n} dom(D_i) \xrightarrow{PROJECT} (dom(DS.p_i))^* \times \prod_{j=1}^{\|P\|} \left(\prod_{k=j\_min}^{j\_max} dom(D_j.p_k) \right)$	**Eq. 12**
$(dom(DS.p_i))^* \times \prod_{j=1}^{\|P\|} \left(\prod_{k=j\_min}^{j\_max} dom(D_j.p_k) \right) \xrightarrow{AGGREGATE} dom(f_{AGG}(dom(DS.p_i))) \times \prod_{j=1}^{\|P\|} \left(\prod_{k=j\_min}^{j\_max} dom(D_j.p_k) \right)$	**Eq. 13**
$\prod_{i=1}^{n} dom(D_i) \xrightarrow{FOCUS} dom(f_{AGG}(dom(DS.p_i))) \times \prod_{j=1}^{\|P\|} \left(\prod_{k=j\_min}^{j\_max} dom(D_j.p_k) \right)$	**Eq. 14**

Concrètement, l'opération de focalisation permet de projeter les valeurs analysées sur les axes d'analyse et ensuite d'agréger ces valeurs en fonction des valeurs des axes d'analyse.

Nous définissons aussi une notation simplifiée (cf. Eq. 15), où s^G représente un sous-ensemble de la galaxie avec une dimension désignée en tant que sujet (S_{AGG}) analysée (projetée et agrégée) selon les dimensions de l'ensemble de projection (P).

$dom(G) \xrightarrow{FOCUS} dom(s^G)$ avec $dom(s^G) = dom(S_{AGG}) \times dom(P)$	**Eq. 15**

Exemple. Au sein de la galaxie G_1 représentée en Figure 50, l'analyste peut sélectionner n'importe quelle dimension en tant que sujet d'analyse. Ici, l'analyste focalise l'analyse sur les principaux mots-clef des sections d'articles, les regroupant par auteur et par année. L'objectif est d'observer de manière grossière les recherches des différents auteurs. Dans l'exemple, la fonction d'agrégation TOP_KEYWORDS [Park et al., 2005] retournera seulement les deux principaux mots-clef. L'instruction suivante produit la table représentée en Figure 52 (également présentée en Figure 51). La granularité d'un article complet est désignée par le paramètre *Document*.

FOCUS(
  G_1,
  (*ARTICLES*, *HS*, <TOP_KEYWORDS(*Section*), *Document*>),
  ((*DATES*, *HTps*, <*Annee*>),
   (*AUTEURS.HA*, <*Auteur*>))
) = $s^{G1}{}_1$

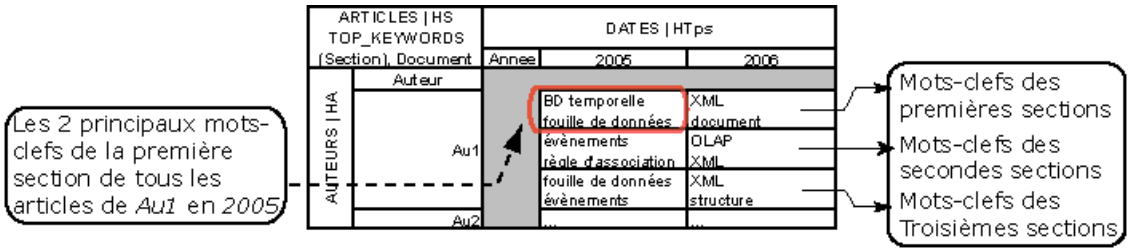


Figure 52 - Exemple de manipulation : opération de focalisation projetant les données d'un sujet d'analyse sur deux axes d'analyse.

La sortie de cette instruction est :

$s^{G1}{}_1 = (S_1, C_1, L_1, res_1)$ avec :
  S_1 = (*ARTICLES*, *HS*, <TOP_KEYWORDS(*Section*), *Document*>) ;
  C_1 = (*DATES*, *HTps*, <*Annee*>) ;
  L_1 = (*AUTEURS*, *HA*, <*Auteur*>) ;
  $res_1 = \varnothing$.

La Figure 53 montre plus en détails la constitution des différentes cellules de la table dimensionnelle en fonction des données sources. Les sources sont constituées de cinq articles, deux datant de *2005* et trois de *2006*. Chaque article est constitué de trois sections (S1, S2 et S3). Le calcul des principaux mots-clef se fait en constituant les sous-groupes des principaux mots-clef par section, par année, par auteur. La fonction d'agrégation TOP_KEYWORDS est abrégée en TOP_KW.

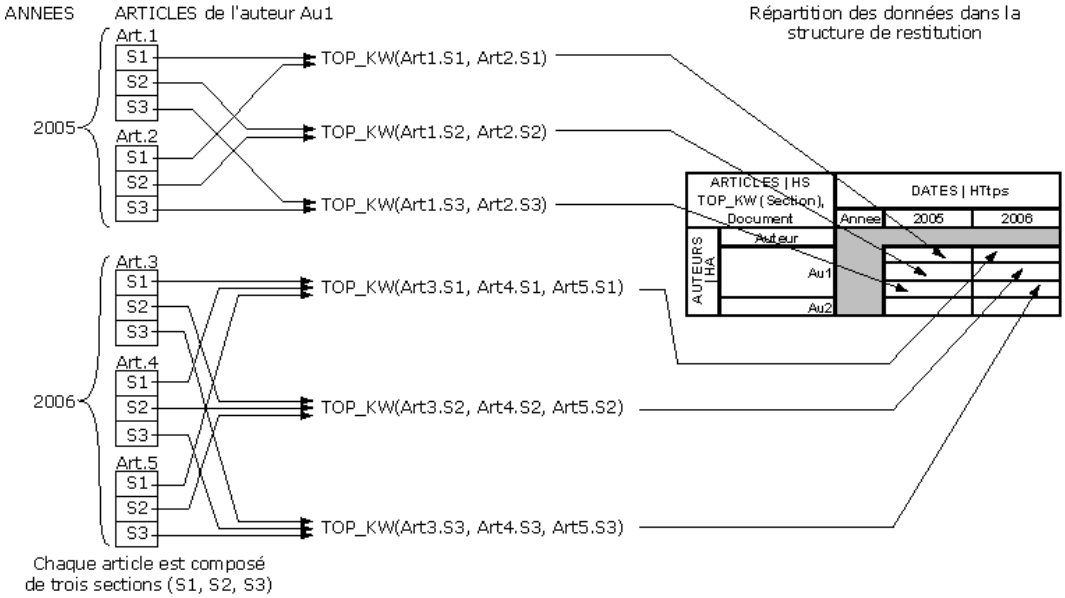


Figure 53 – Détails de l'agrégation (agrégation de sections par article en fonction des auteurs et des années).

Cette fonctionnalité peut être perçue comme l'ajout d'une troisième dimension (celle du sujet d'analyse) au sein du processus d'analyses multidimensionnelles.

Les données documentaires sont liées entre elles souvent par de nombreux liens, le passage suivant expose comment les employer pour permettre une spécification d'analyses plus complètes.

3.2.2 Liens : navigation au sein des données

Les liens récursifs au sein de la galaxie peuvent être employés pour accéder à un ensemble particulier de données. Ceci permet une plus grande flexibilité lors de la désignation de sous éléments de documents et simplifie la spécification de requêtes. L'absence de ses liens se traduirait par des analyses très complexes à exprimer voir même impossibles à formuler à partir de données issues de document.

Par exemple, la séquence d'opérations suivante utilise les liens entre *Reference* et *ARTICLE* (cf. Figure 50). L'opération centre l'analyse sur les principaux mots-clef des articles qui sont cités par l'auteur *Au1*, c'est-à-dire les articles dans les sections « *Références* » des publications de l'auteur *Au1*. Ces principaux mots-clef sont représentés en fonction des années et des auteurs cités par *Au1*. Cette analyse permet d'avoir une vision des domaines que les auteurs font intervenir dans leurs recherches. La spécification de cette analyse emploie une restriction (SELECT) sur les auteurs dont les citations sont analysées : *Au1*, cette instruction sera présentée dans la section suivante (cf. section 3.3.1).

SELECT(
  FOCUS (
    (*ARTICLES*, *HS*, <TOP_KEYWORDS(*ARTICLES.**Reference**.HS.Document*)>),
    ((*TEMPS.HTemps*, <*Annee*>),
    (*ARTICLE.**Reference**.AUTEURS*, *HA*, <*Auteur*>))
  ),
  AUTEURS.Auteur='*Au1*'
)

Où *ARTICLE.**Reference**.AUTEURS* sont les auteurs des articles cités par les publications d'*Au1*. *ARTICLES.**Reference**.HS.Document* représente le contenu complet des articles cités. *ARTICLE.**Reference**.TEMPS.Annee* sont les années de publication des articles cités par *Au1* alors que *TEMPS.Annee* sont les années de publication des articles d'*Au1*.

Ce type d'analyse serait très complexe à exprimer sans l'emploi des liens de la galaxie. Les liens existant dans les sources (les documents), notre modèle permet de les intégrer dans la structure multidimensionnelle et aboutit par leur exploitation à une simplification des analyses.

Autre exemple, pour aller plus loin que la précédente analyse, le décideur observe désormais la portée des travaux de recherches d'un institut (*Inst1*). Il lui suffit d'analyser les principaux mots-clef des différents articles qui citent des articles de cet institut. Cette analyse a été présentée en introduction du chapitre précédent (cf. chapitre 3). Elle est représentée en Tableau 8. Ainsi il est possible de voir quels sont les domaines des articles qui citent les travaux de l'institut. Cette analyse serait simplement impossible à exprimer dans un environnement qui ne permettrait pas l'exploitation des liens entre documents, ou alors elle nécessiterait un magasin entièrement dédié à cette analyse. L'analyse est obtenue à partir de la séquence d'opérations suivantes (en utilisant cette fois une restriction sur l'institut) :

SELECT (
  FOCUS (
    (ARTICLES, HS, <*TOP_KEYWORDS*(*ARTICLES.HS.Document*)>),
    ((*ARTICLES.**Reference**.AUTEURS*, *HA*, <*Auteur, Institut*>),
    (*CONFERENCES*, *HConf*, <*Nom*>))
  ),
  *ARTICLE.**Reference**.AUTEURS.Institut = 'Inst1'*
)

Où *ARTICLES.**Reference**.AUTEURS* sont les auteurs des articles cités dans les conférences *CONFERENCES.Nom* dans les articles dont le contenu est spécifié par *ARTICLES.Article*. Ainsi, *ARTICLE.**Reference**.AUTEURS.Institut = 'Inst1'* représente les articles qui citent des auteurs dont l'institut est « *Inst1* ». Il est à noter que les hiérarchies ne sont spécifiées que dans l'instruction FOCUS, ce qui permet par la suite l'emploi d'opérations de forage sur les hiérarchies en question.

Tableau 8 - Analyse des principaux mots-clef des articles qui citent les auteurs analysés.

ARTICLES TOP_KEYWORDS (Document)		AUTEURS_CITÉS			
		Institut	Inst1		
		Auteur	Au1	Au2	Au3
CONFERENCES	Nom				
	DaWaK		XML, Docments	Fouille de données, Clustering	Fouille de données
	DEXA		XML, BD temporelles	-	-
	CAiSE		Fouille de données	XML, Entrepôts de données	Fouilles de données

Les liens assurent une flexibilité quant à l'expression de requêtes sur des données fortement interconnectées et permettent une exploration plus complète des jeux de données.

À partir d'une analyse multidimensionnelle spécifiée, il se peut que l'utilisateur n'obtienne pas des réponses adéquates. Il est parfois nécessaire de modifier les analyses au moyen d'opérations de manipulation. Ces opérations sont décrites dans la section suivante.

3.3 Spécification des opérations de manipulation

Une fois une analyse multidimensionnelle spécifiée, l'utilisateur peut la modifier afin d'affiner les résultats en vue d'effectuer une prise de décision plus fiable. Cet affinage se fait par l'intermédiaire d'opérations de manipulation qui sont divisées en trois groupes :

- les opérations de manipulation de la portée de l'analyse (restriction / sélection) ;
- les opérations de manipulation du niveau de détail de l'analyse (forage) ;
- les opérations de réorganisation d'analyse (rotation).

L'exemple qui sera employé sera la suite de l'exemple présenté dans la spécification de l'opération de focalisation (cf. section 3.2.1).

3.3.1 Opération de sélection

L'opération de sélection est employée pour réduire la quantité de données à analyser. En spécifiant un prédicat de restriction l'utilisateur peut effectuer une restriction sur un axe d'analyse ou bien sur le sujet d'analyse. Toutes les instances sélectionnées par un prédicat p sont maintenues dans la sélection courante, les autres instances étant retirées.

Remarque. Si cette opération est directement appliquée à la galaxie ceci permet le retrait d'instances avant le processus d'agrégation.

Syntaxe : $SELECT(s^G, p)= s^{G'}$ où s^G est l'entrée et p est un prédicat restrictif sur un attribut a_j d'une dimension.

Conditions : $a_j \in D_i$ et $(D_i \in Star^G(DS) \vee D_i=DS)$

Mathématiquement :

$dom(s^G) \xrightarrow{SELECT(p)} dom(s^G) - dom(\neg p)$	**Eq. 16**

La notation $dom(\neg p)$ représente l'ensemble des instances du domaine ne respectant pas le prédicat p. L'opération inverse $UNSELECT(s^G) = s^{G'}$ supprime tous les prédicats restrictifs.

Exemple. Afin de réduire la portée de l'analyse, l'analyste décide de se restreindre aux seules publications de *Au1* et d'analyser les principaux mots-clef uniquement dans les introductions (la première section). Il s'agit d'appliquer une restriction à la fois sur un axe d'analyse et sur le sujet de l'analyse. En utilisant le sous-ensemble précédemment défini (s^G), l'instruction suivante produit la table définie en Figure 54 :

$SELECT(SELECT(s^{G1}_1, ARTICLE.Titre\_Section=\text{'}Introduction\text{'}), AUTEURS.Auteur=\text{'}Au1\text{'})=s^{G1}_2$

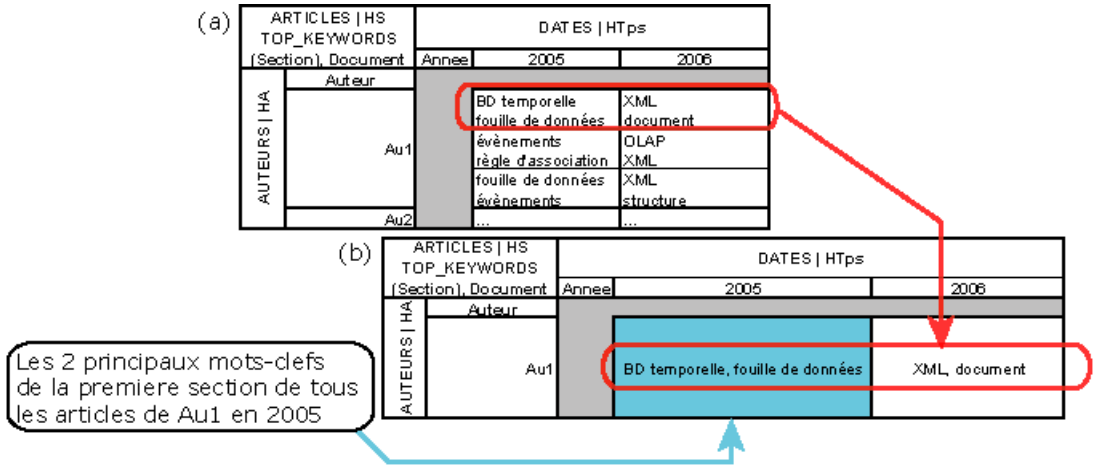


Figure 54 - Exemple de manipulations: application de deux restrictions.

3.3.2 Opérations de forage

Dans une analyse, il est possible de modifier le niveau de détails utilisé pour observer les données. Les opérations de forage permettent de changer ce niveau de détails. Ces opérations s'appuient sur la structure hiérarchique des dimensions. Il s'agit de permettre le changement des paramètres sélectionnés dans un axe d'analyse par d'autres paramètres de la même hiérarchie. Ces opérations de forage sont de deux types :

- le forage vers le bas (drill-down) : consiste à observer les données avec une vision plus détaillée (moins générale).
- le forage vers le haut (roll-up) : opération inverse, elle consiste à observer les données avec une vision moins détaillée (plus générale).

Dans les environnements classiques, la gestion non symétrique des structures multidimensionnelles ne permettait pas l'emploi d'opérations de forage sur les sujets analysés. Contrairement à ces environnements, avec la galaxie qui généralise le concept de dimension, il est désormais possible d'appliquer une opération de forage sur un sujet d'analyse. C'est-à-dire sur une hiérarchie d'une dimension sélectionné en tant que sujet d'analyse.

Forage vers le bas. En employant une opération de forage vers le bas (Drill-Down) l'analyste peut « zoomer » et obtenir des données plus détaillées. Cette opération consiste à ajouter au sein de la liste des paramètres de l'un des axes de projection ($Param_i$), un paramètre p_{new} de la hiérarchie courante dont le niveau est inférieur au niveau du paramètre de plus fine granularité actuellement sélectionné (p_{min}).

Syntaxe : $DRILLDOWN(s^G, D_i, p_{new}) = s^G{}_1$ où s^G est l'entrée, D_i est une dimension de l'ensemble de projection P de s^G, c'est à dire $\exists(D_i.H_i, Param_i) \in P$ tel que $p_{new} \in H_i$.

Condition : Le paramètre doit être d'un niveau inférieur au paramètre de plus fine granularité déjà sélectionné : $level^{Hi}(p_{new}) < level^{Hi}(p_{min})$

Mathématiquement :

$DrillDown: dom(s^G) \xrightarrow{DRILLDOWN} dom(S_{AGG}) \times dom(P) \times dom(D_i.p_{new})$ où $dom(P) = \prod_{\substack{j=1 \\ j \neq i}}^{\|P\|} \left(\prod_{k=j\_\min}^{j\_\max} dom(D_j.p_k) \right) \times \prod_{k'=i\_\min}^{i\_\max} dom(D_i.p_{k'})$	**Eq. 17**

Il est à noter que *dom*(*P*) représente le domaine des paramètres sélectionnés des dimensions ne prenant pas part au forage ($\forall D_j$, $D_j \neq D_i \mid \exists (D_j.H_j, Param_j) \in P$), mais aussi le domaine de la dimension prenant part au forage (D_i).

Nous rappelons aussi que : $Param_j =< p_{j\_\min}, \ldots, p_{j\_\max} > \subseteq Param^{H_j}$

Dans le cadre d'un forage sur la dimension sélectionnée en tant que sujet (*DS*), ce forage transfère la fonction d'agrégation sur le nouvel attribut p_{new}.

Forage vers le haut. L'opération inverse, le forage vers le haut (Roll-Up), est employée pour obtenir une vision plus globale des données analysées. Cette opération est utilisée pour « dézoomer » une vision détaillée des données d'analyse. L'opération consiste à retirer tous les paramètres de la liste des paramètres sélectionnés ($Param_i$) dont les niveaux sont inférieurs à un paramètre spécifié. Ce dernier paramètre est ajouté dans la liste s'il n'y est pas déjà.

Syntaxe : $ROLLUP(s^G, D_i, p_{sup}) = s^G{}_1$ où s^G est l'entrée, D_i est une dimension de l'ensemble de projection *P* de s^G, c'est-à-dire $\exists (D_i.H_i, Param_i) \in P$ tel que $p_{sup} \in H_i$.

Condition : Le paramètre p_{sup} doit être d'un niveau supérieur au paramètre de granularité la plus fine déjà sélectionné : $level^{Hi}(p_{sup}) > level^{Hi}(p_{min})$.

Mathématiquement : dans l'équation suivante (Eq. 18) nous définissons $sup = level^{Hi}(p_{sup})$.

$RollUp : dom(s^G) \xrightarrow{RollUp} dom(S_{AGG}) \times \prod_{\substack{j=1 \\ j \neq i}}^{\|P\|} \left(\prod_{k=j\_\min}^{j\_\max} dom(D_j.p_k) \right) \times \prod_{k'=\sup}^{i\_\max} dom(D_i.p_{k'})$	**Eq. 18**

Ici, $\prod_{k'=\sup}^{i\_\max} dom(D_i.p_{k'})$ est le domaine des paramètres de la dimension prenant part à l'opération de forage (D_i). Le domaine des paramètres dont les niveaux sont inférieurs à p_{sup} sont retirés (ainsi la borne inférieure de *k'* est $level^{Hi}(p_{sup}) = sup$).

Dans le cadre d'un forage sur la dimension sélectionnée en tant que sujet (*DS*), ce forage transfère la fonction d'agrégation sur le nouvel attribut p_{sup}.

Exemple de forage. Comme dans les modèles traditionnels, l'opération de forage peut être utilisée sur une des dimensions employées en tant qu'axe d'analyse. Par exemple, il est possible de forer vers le bas sur la dimension en colonnes (*DATES*) pour représenter les mots-clef de manière plus détaillée et fournir un vision par mois plutôt que par année.

Cependant dans notre modèle, cette opération peut aussi être employée sur la hiérarchie courante de la dimension focalisée. Cette fonctionnalité donne une nouvelle vision du processus d'agrégation en effectuant des agrégations à un nouveau niveau de détails.

A partir de l'exemple précédent, plutôt que d'analyser les principaux mots-clef par section, l'analyste décide de les analyser par paragraphe. Il obtient ainsi une vision plus détaillée du contenu des articles analysés. Pour forer au niveau *Paragraphe*, et effectuer des regroupements par *Document*, il faut éliminer le niveau intermédiaire *Section* qui est actuellement sélectionné en tant que sujet d'analyse. Ainsi il est nécessaire d'effectuer un forage vers le haut au niveau *Document* suivi d'un forage vers le bas au niveau *Paragraphe*. L'instruction suivante produit la table présentée en Figure 55 :

DrillDown (*RollUp* ($s^{G1}{}_2$, *ARTICLES*, *Document*), *ARTICLES*, *Paragraphe*) = $s^{G1}{}_3$

L'utilisateur spécifie un forage vers le haut depuis le paramètre *Section* de la hiérarchie courante (*HS*) de la dimension focalisée *ARTICLES* vers le paramètre de niveau supérieur *Document* pour éliminer le niveau *Section*, suivi d'un forage vers le bas vers niveau inférieur *Paragraphe* (en sautant le niveau *Section*). Au final, l'utilisateur obtient la table dimensionnelle (b).

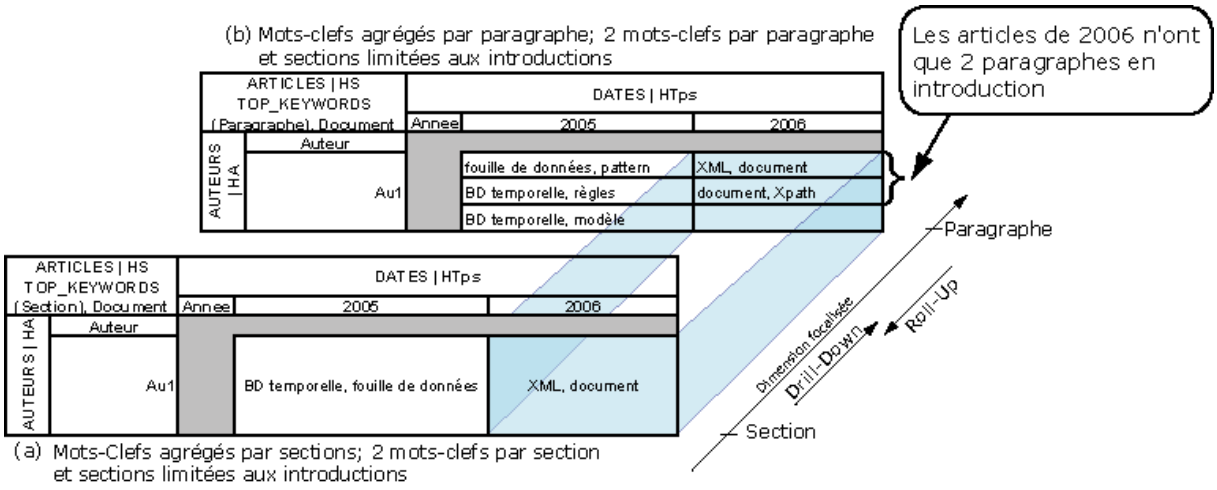


Figure 55 - Exemple de manipulations: forage sur la dimension focalisée.

Il est à noter que les différences dans le nombre de cellules remplies dans la table (b) correspondent à des structures différentes entre les documents. Ainsi il est possible de remarquer qu'en *2005* la première section des articles de l'auteur *Au1* est composée au maximum de trois paragraphes alors qu'en *2006*, la structure est composée au maximum de seulement deux paragraphes (d'où une cellule avec le caractère '-').

Forer sur une hiérarchie courante de la dimension focalisée permet une combinaison puissante de :

- l'utilisation de la hiérarchisation des données fournit par la structure hiérarchique des données dimensionnelles ;
- l'utilisation du processus d'agrégation permettant de résumer des données sélectionnées.

Forer sur la hiérarchie focalisée peut être vue comme ajouter dans la manipulation un troisième axe d'analyse aux deux axes classiques représentés dans une table dimensionnelle (l'axe représenté en colonnes et l'axe représenté en lignes).

Cette opération permet à l'utilisateur d'avoir une vision du processus d'agrégation. Ceci est dû au fait que les fonctions d'agrégations textuelles telles qu'AVG_KW ou celles définies dans [Park et al., 2005], n'opèrent pas de la même manière que les fonctions d'agrégations numériques. En effet, l'extraction des principaux mots-clef d'un article ne correspond pas nécessairement à l'extraction des principaux mots-clef de chaque section qui composent l'article. Ceci est un problème connu :

- Les fonctions holistiques [Gray et al., 1996] ne peuvent être calculés à partir de résultats intermédiaires (par exemple la fonction qui calcule la médiane).
- Le « component ranking » tel que pointé dans [Mass & Mandelbrod, 2004], où dans un environnement de recherche d'information les différents niveaux de granularité des fragments de données retournés à l'utilisateur ont tendance a chambouler les statistiques derrière les modèles de recherche d'information.

Au niveau physique, lors d'un forage sur des données textuelles (ou non) agrégées par une fonction holistique, les agrégats de données sont recalculés systématiquement avec le niveau de granularité nouvellement sélectionné. Ainsi l'analyste peut avoir une meilleure interprétation en disposant de ces différents niveaux d'agrégations.

3.3.3 Opération de réorganisation d'analyse

Comme pour les opérations de forage, les opérations de réorganisation d'analyse n'étaient applicables qu'aux axes d'analyse hormis une opération spécifique qui s'appliquait au sujet d'analyse et uniquement dans le cadre de modélisation en constellation (la rotation de fait). Désormais dans le cadre d'analyse spécifiées à partir d'une modélisation en galaxie, ces opérations sont généralisées et permettent un traitement symétrique des attributs et des dimensions qu'ils soient considérés comme sujet ou axe d'analyse.

Dans certains cas, l'utilisateur pourrait vouloir réorganiser les données analysées. A cette fin, le décideur peut employer l'opération de rotation qui lui permet de remplacer l'une des dimensions employées dans une analyse.

L'opération de rotation remplace l'une des dimensions de s^G par une nouvelle dimension. La dimension remplacée peut être le sujet d'analyse (*DS*) ou bien l'un des axes d'analyse, c'est-à-dire une des dimensions de l'ensemble de projection ($D_i \in P$).

Syntaxe : $ROTATE(s^G, D_{old}, D_{new}.H_{new}, Param_{new})=s^G{}_1$ où :

- s^G est l'entrée ;
- D_{old} est la dimension à remplacer ;
- D_{new} est la nouvelle dimension ;
- H_{new} est sa hiérarchie courante de D_{new} ;
- $Param_{new}$ dépends de D_{old} : Si $D_{old}=DS$ alors $Param_{new}=<f'_{\text{AGG}}(p_{new}), p_{new\_min},\ldots, p_{new\_max}>$, sinon si $D_{old}\neq DS$ alors $Param_{new} = <p_{new\_min},\ldots, p_{new\_max}>$ (un sous-ensemble de $Param^{Hnew}$).

Conditions :

- si $D_{old}=DS$ alors $\forall D_k \in P$, $D_k \in \text{Star}^G(\text{D}_{\text{new}})$ et $p_{new} \in H_{new}$ et $level^{Hnew}(p_{new}) > level^{Hnew}(p_{new\_min}) > \ldots > level^{Hnew}(p_{new\_max})$
- Si $D_{old} \in P$ alors $D_{new} \in Star^G(DS)$ $Param_{new} \subseteq Param^{Hnew}$ et $level^{Hnew}(p_{new\_min}) < \ldots < level^{Hnew}(p_{new\_min})$

Mathématiquement : si $D_{old}=DS$ alors l'opération correspond à (Eq. 19), sinon si $D_{old} \in P$, alors l'opération correspond à (Eq. 20).

$Rotate : dom(s^G) \xrightarrow{ROTATE} dom(f'_{AGG}(dom(D_{new}.p_{new}))) \times dom(P)$	**Eq. 19**
$Rotate : dom(s^G) \xrightarrow{ROTATE} dom(S_{AGG}) \times \prod_{\substack{j=1 \\ j\neq old}}^{\|P\|} \left(\prod_{k=j\_\min}^{j\_\max} dom(D_j.p_k)\right) \times \prod_{k'=new\_\min}^{new\_\max} dom(D_{new}.p_{k'})$	**Eq. 20**

Remarque. Si $D_{old} = D_{new}$, ceci permet le changement d'une des hiérarchies courantes (*HS*, H_x, H_y…) : $ROTATE(s^G, D_{old}, D_{old}, H_{new}, Param_{new})$. La rotation du sujet d'analyse (*DS*) est l'équivalent des opérations *FRotate* [Ravat et al., 2007e] ou *DrillAcross* [Abelló et al., 2003].

Exemple. Ici aussi, l'opération de rotation pourrait être employée pour spécifier le changement de l'un des axes d'analyse. Mais elle peut aussi être employée pour exprimer le changement du sujet d'analyse. En l'occurrence, l'utilisateur va donc observer le taux de

sélectivité moyen des publications de l'auteur analysé. L'expression suivante représente l'opération de rotation du sujet d'analyse.

$Rotate(s^{G1}{}_3, ARTICLES, CONFERENCES, HConf, <\text{AVG}(Tx\_Acceptation)>) = s^{G1}{}_4$

Le Tableau 9 représente le résultat de la nouvelle analyse.

Tableau 9 – Analyse des taux moyens d'acceptation d'articles de Au1.

CONFERENCES | Hconf AVG (CONFERENCES.Tx_Acceptation)		DATES | HTps		
		Annee	2005	2006
AUTEURS | HA	Auteur			
	Au1		22,3	38,1

3.4 Bilan concernant la manipulation

Après avoir défini un modèle conceptuel et des fonctions d'agrégation adaptées aux données textuelles, nous avons défini un ensemble d'opérations de manipulation multidimensionnelles. Ces opérations permettent dans un premier temps la spécification d'analyses multidimensionnelles en précisant un sujet d'analyse. Les données analysées sont observées suivant un certain nombre d'axes d'analyse (deux dans le cas d'un table bidimensionnelle). Dans un deuxième temps, un ensemble d'opérations de manipulation permet d'affiner les analyses spécifiées. Ces opérations permettent une plus grande flexibilité quant aux analyses OLAP de données issues de documents. Contrairement aux précédentes propositions d'opérateurs de manipulation, telle que [Gyssens & Lakshmanan, 1997] ou encore [Datta & Thomas, 1999], les opérations sont complètement symétriques et opèrent aussi bien sur les dimensions désignées en tant que sujet d'analyse que sur les dimensions désignées en tant qu'axe d'analyse.

L'ensemble des opérations présentées est résumé dans les tableaux suivants : Tableau 10 pour l'opération de spécification d'analyse et le Tableau 11 pour les opérations de modification d'analyse.

Tableau 10 – Résumé de l'opérateur de spécification d'analyse.

Entrée	Sortie	Syntaxe
FOCUS (cas général), spécification d'une analyse		
G	s^G	$s^G = FOCUS(G, S, P)$ où : $S = (DS, HS, Param_S)$ et $Param_S = <f_{AGG}(p_i)[, p_{S\_min},\ldots,p_{S\_max}]>$, $P = <(D_x, H_x, Param_x), (D_y, H_y, Param_y),\ldots>$ avec : $Param_x = <p_{x\_min},\ldots,p_{x\_max}>$ et $Param_y = <p_{y\_min},\ldots,p_{y\_max}>,\ldots$
FOCUS (cas particulier), réduit à une table bidimensionnelle (T)		
G	T^G	$T^G = FOCUS(G, S, P)$ où : $S = (DS, HS, Param_S)$ et $Param_S = <f_{AGG}(p_i)[, p_{Sj}]>$, $P = <C, L> = <(D_C, H_C, Param_C), (D_L, H_L, Param_L),\ldots>$ avec : $Param_C = <p_{C\_min},\ldots,p_{C\_max}>$ et $Param_L = <p_{L\_min},\ldots,p_{L\_max}>,\ldots$

Tableau 11 – Résumé des opérateurs de manipulations.

Entrée	Sortie	Syntaxe
SELECT, restriction des données analysées		
s^G	$s^{G'}$	$S^{G'} = \text{SELECT}(s^G, p)$
DRILLDOWN, forage vers le bas		
s^G	s^G	$S^{G'} = \textit{DRILLDOWN}(s^G, D_i, p_{inf})$
ROLLUP, forage vers le haut		
s^G	$s^{G'}$	$S^{G'} = \textit{ROLLUP}(s^G, D_i, p_{sup})$
ROTATE, réorientation de l'analyse		
s^G	$s^{G'}$	$S^{G'} = \textit{ROTATE}(s^G, D_{old}, D_{new}, Param_{new})$ où si $D_{old} = DS$, $Param_{new} = <f_{AGG}(p_{new}), p_{new\_min}, \ldots, p_{new\_max}>$ si $D_{old} \neq DS$, $Param_{new} = <p_{new\_min}, \ldots, p_{new\_max}>$

Il est possible de comparer l'expressivité de ces opérations par rapport aux catégories spécifiées dans [Ravat et al., 2007e] (cf. Tableau 12). L'ensemble des opérations générales est supporté. Toutefois, l'intégralité des catégories d'opérations n'est pas supportée. L'opération d'ordonnancement de type imbrication (NEST) n'est pas considérée dans notre modèle car elle brise la structure hiérarchique d'une dimension. En effet, cette opération permet de réordonner librement les paramètres d'une hiérarchie alors que nous désirons nous appuyer sur cette structure pour maintenir une capacité de forage à tout moment selon les hiérarchies des dimensions. De plus, les opérations binaires ne sont pas considérées car ces dernières sont lié à la fusion de tables multidimensionnelles et sortent du cadre de cette thèse. Néanmoins il faut noter deux nouveautés : la possibilité d'exprimer des forages sur la hiérarchies des attributs sélectionnés en tant que sujet d'analyse par l'opération de focalisation ; et la symétrie parfaite entre tous les attributs d'une galaxie.

Tableau 12 – Puissance d'expression des opérateurs multidimensionnels.

Catégories d'opérations		Opération proposée
Forage	niveau plus détaillé	Drill Down
	niveau moins détaillé	Roll Up
Sélection	Valeurs factuelles	Select
	Valeurs dimensionnelles	
Rotation	Fait	Rotate
	Dimension	
	Hiérarchie	
Modification de fait	Ajout d'une mesure	Focus
	suppression d'une mesure	
Modification de dimension	Réduction de dimension	Focus
	Push	
	Pull	

L'environnement OLAP est bien maîtrisé de nos jours par les décideurs. Afin de bénéficier de cette expertise de manipulation, les opérations définies ont des fonctionnalités similaires aux opérations de manipulations proposées au cours de la décennie de recherche sur la manipulation des bases de données multidimensionnelles.

Toutefois la simplification du modèle multidimensionnel par une modélisation reposant sur un unique concept a généralisé le principe de symétrie entre attributs dimensionnels et factuels (mesures). En effet, dans le modèle en galaxie des opérations qui auparavant ne s'appliquaient qu'à des sujets d'analyse ou bien qu'à des dimensions sont généralisées et opèrent désormais sur l'unique structure qu'elle soit considérée comme axe ou sujet d'analyse. Ceci a pour conséquence, une simplification des opérations et une spécification plus simple de l'expression des opérations.

L'ensemble des opérateurs est fermé. C'est-à-dire que les opérations génèrent toutes une sortie commune permettant un enchaînement des opérations afin de permettre la spécification de manipulations complexes. L'opérateur de focalisation permet l'initialisation d'une analyse en construisant une structure d'analyse. Les autres opérations peuvent s'empiler sur cette structure et permettre la manipulation et l'affinage de l'analyse. Les opérations de manipulation permettent de : 1) restreindre la portée de l'analyse en réduisant le volume des données analysées ; 2) de détailler plus ou moins les données à analyser ; et 3) de réorienter l'analyse selon les besoins de l'utilisateur.

Lors de la spécification d'analyses avec ces opérations il est possible aussi d'effectuer des analyses complexes en utilisant les liens présents dans la structure des documents analysés et entre les données d'analyse. Ainsi des analyses qui se seraient avérées très difficiles voire impossibles à spécifier dans un environnement traditionnel, sont désormais accessibles.

4 Bilan : galaxie et analyses multidimensionnelles

Afin de pouvoir spécifier des analyses sur des données textuelles, nous avons proposé une fonction d'agrégation permettant l'agrégation de données textuelles. Cette fonction est employée avec les opérations de manipulation pour permettre la spécification d'analyses multidimensionnelles à partir des concepts représentés par la galaxie.

Nous avons présenté les opérations permettant l'exploitation d'analyse multidimensionnelles sur des donnés issues de document XML. Ces opérations permettent la spécification d'une analyse puis sa modification afin que le décideur puisse affiner l'analyse et optimiser sa prise de décision.

L'exploitation de documents dans une analyse dimensionnelle est une chose, il en est une autre de disposer d'un magasin de données modélisé par une galaxie pour les analyser. Le chapitre suivant présente des éléments méthodologiques pour l'intégration de documents dans un environnement OLAP.

Références

[Abelló et al., 2003] Alberto Abelló, José Samos, Fèlix Saltor, "Implementing operations to navigate semantic star schemas", *6th ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 56–62, 2003.

[Agrawal et al., 1997] Rakesh Agrawal, Ashish Gupta, Sunita Sarawagi, "Modeling Multidimensional Databases", *13th Intl. Conf. on Data Engineering* (*ICDE*), IEEE Computer Society, p. 232–243, 1997.

[Baziz, 2005] Mustapha Baziz, *Indexation conceptuelle guidée par ontologie pour la recherche d'information*, Thèse de doctorat, Université Paul Sabatier, Toulouse 3 (France), décembre 2005.

[Bender & Farach-Colton, 2000] Michael A. Bender, Martin Farach-Colton, "The LCA Problem Revisited", *4th Latin American Symposium on Theoretical Informatics* (*LATIN*), LNCS 1776, Springer, p. 88–94, 2000.

[Cabibbo & Torlone, 1997] Luca Cabibbo, Riccardo Torlone, "Querying Multidimensional Databases", *6th Intl. Workshop Database Programming Languages* (*DBPL*), LNCS 1369, Springer, p. 319–335, 1997.

[Codd, 1972] E. F. Codd, "Relational Completeness of Data Base Sublanguages", *Database Systems*, R. Rustin (ed.), Prentice Hall & IBM Research Report RJ 987, p. 65–98, 1972.

[Datta & Thomas, 1999] Anindya Datta, Helen Thomas, "The cube data model: a conceptual model and algebra for on-line analytical processing in data warehouses", *Decision Support Systems* (*DSS*), vol.27(3), Elsevier, p. 289–301, décembre 1999.

[Gray et al., 1996] Jim Gray, Adam Bosworth, Andrew Layman, Hamid Pirahesh, "Data Cube: A Relational Aggregation Operator Generalizing Group-By, Cross-Tab, and Sub-Total", *12th Intl. Conf. on Data Engineering* (*ICDE*), IEEE Computer Society, p. 152–159, 1996.

[Gyssens & Lakshmanan, 1997] Marc Gyssens, Laks V. S. Lakshmanan, "A Foundation for Multi-dimensional Databases", *23rd Intl. Conf. on Very Large Data Bases* (*VLDB*), Morgan Kaufmann, p. 106–115, 1997.

[Harel & Tarjan, 1984] Dov Harel, Robert Endre Tarjan, "Fast Algorithms for Finding Nearest Common Ancestors", *SIAM Journal on Computing* (*SICOMP*), vol.13(2), SIAM, p. 338–355, mai 1984.

[Kimball, 1996] Ralph Kimball, *The data warehouse toolkit: Practical Techniques for Building Dimensional Data Warehouses*, John Wiley and Sons, ISBN : 0-471-15337-0, 1996, 2ème ed. : Ralph Kimball, Margaery Ross, *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling*, *2nd Edition*, John Wiley & Sons, 2002.

[Klug, 1982] Anthony C. Klug, "Equivalence of Relational Algebra and Relational Calculus Query Languages Having Aggregate Functions", *Journal of the ACM* (*JACM*), vol.29(3), ACM Press, p. 699–717, juillet 1982.

[Lassila & McGuiness, 2001] Ora Lassila, Deborah L. McGuinness, "The Role of Frame-Based Representation on the Semantic Web", Knowledge Systems Laboratory Report KSL-01-02, Stanford University, 2001 (publié aussi dans *Computer and Information Science*, vol.6(5), Linköping University, 2001).

[Mass & Mandelbrod, 2004] Yosi Mass, Matan Mandelbrod, "Component Ranking and Automatic Query Refinement for XML Retrieval", *3rd Intl. Workshop of the Initiative for the Evaluation of XML Retrieval, Advances in XML Information Retrieval* (*INEX*), LNCS 3493, Springer, p. 73–84, 2004.

[Park et al., 2005] Byung-Kwon Park, Hyoil Han, Il-Yeol Song, "XML-OLAP: A Multidimensional Analysis Framework for XML Warehouses", *7th Intl. Conf. on Data Warehousing and Knowledge Discovery* (*DaWaK*), LNCS 3589, Springer, p. 32–42, 2005.

[Ravat et al., 2006] Franck Ravat, Olivier Teste, Gilles Zurfluh, "Algèbre OLAP et langage graphique", *Actes du XXIVème Congrès INFormatique des ORganisations et Systèmes*

d'Information et de Décision (*INFORSID*), Inforsid (Ed.), ISBN 2-906855-22-7, p. 1039–1054, 2006.

[Ravat et al., 2007e] Franck Ravat, Olivier Teste, Ronan Tournier, Gilles Zurfluh, "Algebraic and graphic languages for OLAP manipulations", *Intl. Journal of Data Warehousing and Mining* (*ijDWM*), Idea Group Publishing (IGP), juin 2007 (à paraître).

[Tseng & Chou, 2006] Frank S.C. Tseng, Annie Y.H. Chou, "The concept of document warehousing for multi-dimensional modeling of textual-based business intelligence", *journal of Decision Support Systems* (*DSS*), vol.42(2), Elsevier, p. 727–744, novembre 2006.

CHAPITRE V
Intégration multidimensionnelle de documents

Résumé du chapitre

Ce chapitre présente une méthode permettant l'intégration de données issues de documents au sein des structures multidimensionnelles d'un magasin de données modélisé par un schéma en galaxie. Cette méthode repose sur une démarche mixte qui effectue simultanément la spécification des besoins des décideurs en termes d'analyse par l'intermédiaire d'un schéma multidimensionnel en galaxie et l'analyse des sources de données. Une étape de confrontation permet de s'assurer de la compatibilité entre les sources et la formalisation des besoins d'analyse. En cas d'incompatibilité, soit les besoins sont revus à la baisse, soit les sources sont enrichies à partir de données auxiliaires. Une fois les sources et la galaxie compatibles, le magasin est alimenté en données.

Sommaire

CHAPITRE V : Intégration multidimensionnelle de documents

"This notion of being able to semantically link various resources [...] is an important one. With this we can begin to move from the current Web of simple hyperlinks to a more expressive semantically rich Web, a Web where we can incrementally add meaning and express a whole new set of relationships [...] This will open new doors for effective information integration, management and automated services."

— Tim Berners Lee & Eric Miller.

1 Introduction

Les deux chapitres précédents exposent les concepts du modèle et les opérations permettant la manipulation de ces concepts. Le but de ce chapitre est de présenter des éléments de démarche pour permettre l'intégration de données issues de documents XML dans un environnement multidimensionnel modélisé par une galaxie. Il s'agit d'effectuer une intégration multidimensionnelle des données issue de documents au sein des structures du magasin de données modélisées par un schéma en galaxie.

La Figure 56 positionne l'intégration de documents par rapport à l'architecture générale d'un système d'aide à la prise de décision. Nous supposons disposer en entrée d'une collection de documents au format XML. Cette collection représente un ensemble de documents avec une structure commune (une DTD commune).

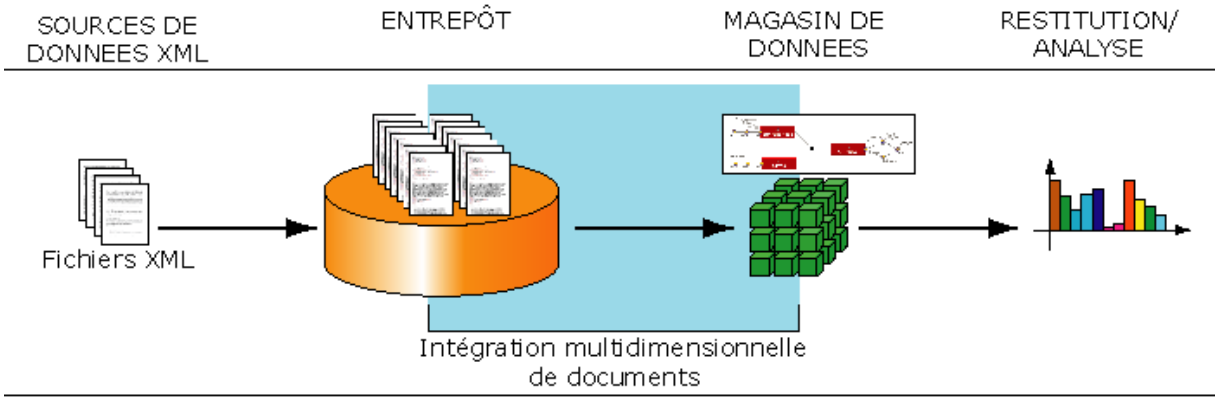


Figure 56- Architecture d'un système d'aide à la prise de décision alimenté par des documents XML.

La Figure 57 présente les différentes étapes de l'intégration de données issues de documents de l'entrepôt dans un magasin de données. Notre approche est décomposée en quatre étapes :

- *Spécification des besoins* d'analyse et *analyse des sources* de données.
- *Confrontation* des schémas des sources et du schéma multidimensionnel ;
- *Enrichissement* des sources (optionnelle) ;
- *Alimentation* du magasin en données.

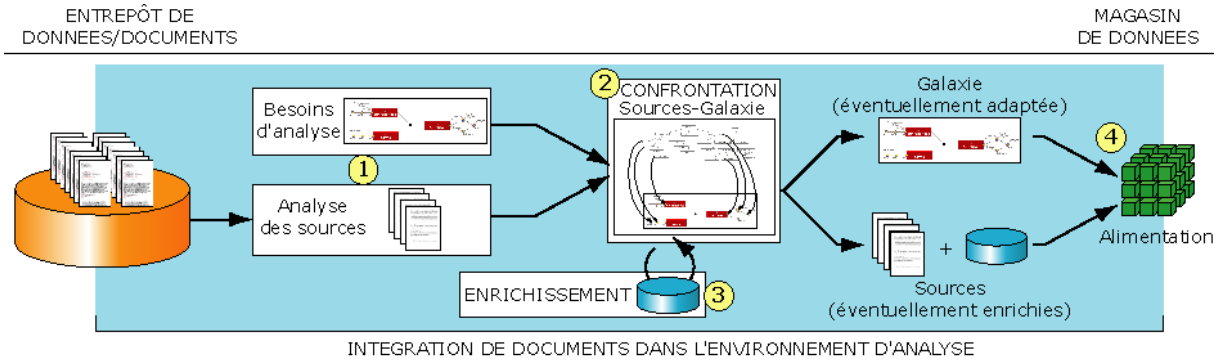


Figure 57 – Détails des étapes de l'intégration de documents.

L'intégration s'effectue selon une démarche mixte. Elle commence par la spécification simultanée des besoins d'analyse et de l'analyse des sources de données (cf. (1) dans la Figure 57). La spécification des besoins se fait via une représentation conceptuelle en galaxie. L'analyse des sources se fait en les représentant par leur schéma logique (une vision arborescente de la structure XML). Une étape de confrontation et d'association permet de s'assurer de la conformité des deux schémas (2) et de lier les sources aux structures multidimensionnelles. En cas d'incompatibilité deux alternatives sont possibles : l'enrichissement des sources (3) ou bien la modification des besoins d'analyse (le schéma multidimensionnel), selon un processus itératif. Enfin, en suivant les liaisons spécifiées durant la confrontation, les structures multidimensionnelles du magasin sont alimentées avec les données sources (4).

Plan du chapitre. La section 2 expose la spécification des besoins et la section 3 se focalise sur l'analyse des sources. Afin de mettre en conformité les besoins d'analyse et les sources, la section 4 présente la confrontation entre le résultat de l'analyse des sources et la formalisation des besoins. En cas d'absence de compatibilité à l'issue de la confrontation, la section 5 présente plus en détails l'une des solutions : l'étape d'enrichissement des sources. Enfin, la section 6 présente l'étape finale d'alimentation des données au sein du magasin.

2 Spécification des besoins d'analyse

But. L'objectif de cette section est de permettre la spécification du schéma conceptuel multidimensionnel en galaxie décrivant un magasin. Ce schéma est spécifié selon les besoins des décideurs. Suite à la spécificité du modèle en galaxie, nous avons été amenés à modifier la méthode descendante proposée dans [Ghozzi, 2004] qui repose, à l'origine, sur la dualité des concepts multidimensionnels de fait et de dimension.

La Figure 58 présente les trois étapes de la modélisation des besoins d'analyse :

- la collecte des besoins,
- la spécification des besoins collectés,
- la formalisation des besoins.

En premier lieu les besoins en terme d'analyse des décideurs sont collectés. Par la suite, ces besoins sont analysés et spécifiés au moyen d'une matrice. Enfin, durant l'étape de formalisation, ils sont traduits en un schéma multidimensionnel en galaxie.

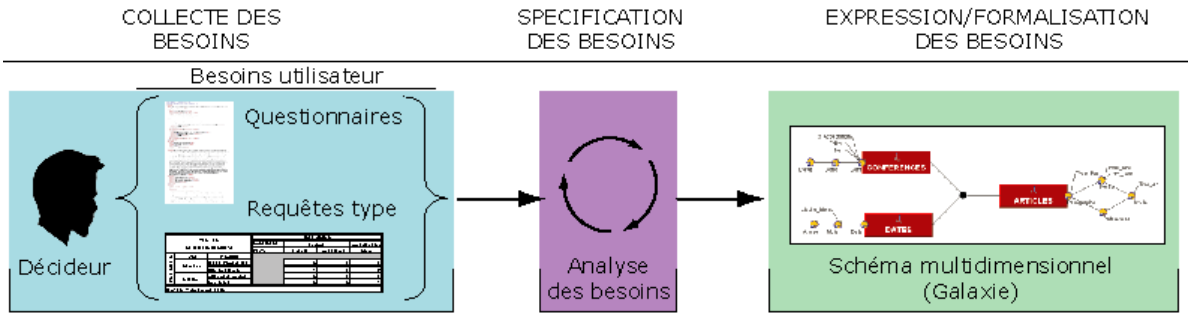


Figure 58 – Démarche de la modélisation des besoins d'analyse.

2.1 Collecte des besoins d'analyse

La première partie de la démarche de modélisation des besoins d'analyse est la collecte des objectifs et des besoins d'analyse. Cette collecte se fait auprès des décideurs et permet l'obtention des informations susceptibles de les intéresser en terme d'analyse.

Cette collecte d'effectue grâce à :

- la collecte de *requêtes-types* ;
- la saisie de *questionnaires* adressés aux décideurs.

Les requêtes-types permettent de spécifier des analyses que voudraient spécifier les décideurs et les questionnaires sont employés pour réunir des informations complémentaires.

2.1.1 Collecte par requêtes-type

Les *requêtes types* des décideurs sont collectées à partir de trois moyens :

- les rapports d'analyse existants ;
- des tables multidimensionnelles (ou tableaux croisés) ;
- l'emploi d'un pseudo-langage.

Les rapports d'analyse existants permettent de connaître les précédentes analyses effectuées par les décideurs. Une interview des décideurs permet de connaître la pertinence de ces rapports dans le contexte actuel et si ces analyses doivent être maintenues ou modifiées. Les tables multidimensionnelles permettent de collecter des besoins courants des décideurs. Les décideurs maîtrisant les tables multidimensionnelles, il est possible de leur demander de spécifier leurs besoins sous la forme de tables car elles contiennent les réponses aux questions posées par l'expression des requêtes-types. Enfin l'emploi d'un pseudo-langage permet une spécification plus concrète d'analyses.

Ce pseudo-langage permet de faciliter l'expression des besoins [Ghozzi, 2004] et repose sur trois clauses :

- clause **Analyser** : cette clause réponds à la question « **Quoi ?** ». Elle définit les données que les décideurs désirent analyser ;
- clause **En Fonction** : cette clause réponds aux questions « **Qui ?** », « **Où ?** » et « **Quand ?** ». Elle indique les paramètres de l'analyse des données décrites par la clause **Analyser** ;
- clause **Pour** : cette clause réponds aux questions « **Pour qui ?** », ou « **Quelles données ?** ». Elle précise les restrictions sur les données qui seront analysées.

2.1.2 Collecte par questionnaires

Le *questionnaire* est un formulaire qui guide le décideur à exprimer ses besoins par l'intermédiaire d'un ensemble de questions. Ces questions sont orientées pour permettre d'obtenir les informations nécessaires à l'expression des besoins d'analyse sous la forme d'un schéma multidimensionnel. Le but des questionnaires est de compléter la description des besoins saisie par les requêtes-types. Le questionnaire contient aussi des questions concernant les documents qui seront employés dans l'analyse. Ces questions permettent d'identifier les caractéristiques des documents telles que l'emploi des documents, leur structuration, leur contenu général…

À l'issue de la collecte des besoins, l'ensemble des sources des requêtes-types (rapports, tables…) est traduit en pseudo-langage. Ceci permet d'obtenir une uniformisation des besoins. Le questionnaire est employé pour éventuellement affiner l'expression de ces requêtes types.

2.1.3 Exemple

En guise de requête-type, il est possible de collecter des tables multidimensionnelles spécifiées par le décideur (cf. Tableau 13). Dans cet exemple, le décideur désire analyser la portée des articles scientifiques de certains auteurs. La table présente le nombre total d'article par conférence citant les travaux des auteurs en colonnes. Par exemple, les articles de l'auteur Au1 ont été cités 3 fois dans des articles de la conférence DaWaK. De manière très simple il est possible de détecter que les références des articles seront analysées, mais qu'il faudra conserver aussi le lien entre les articles cités et les articles les citant.

Tableau 13 – Exemple de requête-type représentée par une table multidimensionnelle.

COUNT (Articles)	Institut	Inst1		
	Auteur	Au1	Au2	Au3
Conference				
DaWaK		3	2	1
DEXA		2	-	-
CAiSE		1	1	2

La requête-type en pseudo-langage associée à cette table multidimensionnelle est la suivante :

> **Analyser** le *nombre de références* **en fonction** du *nom des auteurs* de ces références **et en fonction** du *nom des conférences* des articles qui contiennent ces références, **pour** les articles contenant des références aux articles des auteurs de l'institut *Inst1*.

D'autre requêtes-types ont aussi été récoltées :

> **Analyser** le *contenu d'articles* **en fonction** du *nom de l'auteur* de l'article **et en fonction** de l'*année de publication* de l'article **pour** un contenu d'article limité aux sections de type introduction.
> **Analyser** le *nombre d'article* **en fonction** du *nom de l'auteur* **et en fonction** des *années* **pour** les conférences à audience internationale.
> **Analyser** le *nombre de rapports* **en fonction** du *nom de l'auteur* **et en fonction** de l'*institut* émetteur du rapport.
> **Analyser** le *nombre moyen d'institut* dont dépendent les auteurs de rapports **en fontion** de l'*institut* émetteur du rapport **et en fontion** de l'*année* **pour** les rapports de type rapport scientifique.

Le questionnaire associé à ces requêtes types permet de récolter les informations concernant les documents qui seront analysés (des articles scientifiques) :

> Les articles scientifiques suivent des canevas très similaires. La structuration du texte est constituée de paragraphes regroupés en sections. Les paragraphes sont typés. Par exemple, standard, définition, théorème, référence (…) sont des types de paragraphes. Il en est de même pour les sections (introduction, conclusion, bibliographie…). Les articles scientifiques se citent entre eux et ainsi il existe une liaison entre les différents articles.
> Les rapports sont des documents émis par un institut de recherche et écrit par des auteurs. Ces auteurs ne dépendent pas nécessairement de l'institut qui émet le rapport.

Une fois les besoins collectés, il est nécessaire de les analyser pour pouvoir les interpréter et les convertir ensuite en un schéma multidimensionnel.

2.2 Spécification des besoins

But. La sortie de la collecte des besoins est une liste de requêtes-types formulées à l'aide d'un pseudo-langage. Le but de cette étape est d'obtenir, à partir des informations fournies par les décideurs, les attributs et leurs interactions pour la conception du schéma en galaxie.

2.2.1 Matrice des besoins

Une matrice des besoins permet la spécification de l'interaction des attributs entre eux et permet ainsi de les regrouper au sein des différentes dimensions. Dans un environnement reposant sur l'analyse de données numériques avec des faits et des dimensions, cette étape est critique. En effet, une erreur de spécification entre données factuelles et données dimensionnelles peut rapidement limiter les analyses initialement prévues. Grâce à l'unique concept de la galaxie, cette étape est beaucoup moins critique.

La *matrice des besoins* est une matrice de co-occurrence d'attributs, c'est-à-dire une matrice carrée dont les lignes et les colonnes sont composées de la liste des attributs. La liste des attributs est extraite des requêtes-types et est insérée dans la matrice. Les clauses suivantes du pseudo-langage, qui exprime les requêtes-types, permettent la détection des attributs :

- **analyser** permet de détecter des attributs importants qui seront de manière préférentielle au centre d'analyses. Ils représenteront des sujets d'analyse ;
- **en fonction** et **pour**, permettent de spécifier les autres attributs.

Dans la matrice une case cochée indique qu'un attribut en ligne est analysé en fonction de l'attribut en colonne. Une case de la diagonale cochée signifie que l'attribut intervient uniquement dans une clause *pour*. Il s'en suit une étape de simplification de la matrice.

Règle 1. Chaque ligne ou colonne vide est supprimée de la matrice car une ligne vide signifie que l'attribut en question n'est pas utilisé en tant que sujet d'analyse. Une colonne vide signifie que l'attribut ne décrit pas un sujet d'analyse.

Règle 2. Un attribut présent à la fois en ligne et en colonne doit être supprimé de l'une ou de l'autre, si la diagonale n'est pas cochée. Les connaissances métier sont employées pour conserver soit la ligne soit la colonne.

Au final les attributs sujets d'analyse sont définis par les lignes et les attributs des axes d'analyses sont dans les colonnes. Ceci permet de spécifier la disjonction suivante entre les attributs :

Règle 3. Tout attribut en ligne ne sera pas au sein de la même dimension que les attributs en colonne auxquels il est associé, à l'exception des attributs dont la case diagonale est cochée.

2.2.2 Exemple

En reprenant les requête type précédente il est possible d'extraire des sujets d'analyse, les attributs suivants :

- Références, Contenu.
- Nombre d'articles, Nombre de rapports, nombre moyen d'instituts : qui sont des comptages d'*Articles*, de *Rapports* et d'*Instituts*(*Auteur*).

Et les autres attributs concernés :

- Nom_Auteur, Nom_Conférence, Nom_Institut, Année, Type_Section, Audience, Institut(Rapport), Type_Rapport.

La matrice des besoins correspondante est représentée en Tableau 14. Dans ce tableau, la diagonale est identifiée par des cellules grises. Le Tableau 15 présente la matrice simplifiée, après application de la règle 1 et 2. Les lignes (et colonnes) de couleur orange (en gris) dans la matrice simplifiée représentent les attributs à la fois présents en lignes et en colonne (dont la case diagonale est cochée). Par la suite, ces attributs seront tous supprimés des lignes.

Tableau 14 – Exemple de matrice des besoins.

Nom des attributs	Référence	Contenu	Article	Rapport	Institut (Auteur)	Nom_Auteur	Nom_Conference	Année	Type_Section	Audience	Institut (Rapport)	Type_Rapport
Référence						x	x					
Contenu						x		x	x			
Article						x		x		x		
Rapport						x					x	
Institut(Auteur)								x			x	x
Nom_Auteur												
Nom_Conference												
Année												
Type_Section									x			
Audience										x		
Institut(Rapport)												
Type_Rapport												x

Tableau 15 – Matrice simplifiée.

Nom des attributs	Nom_Auteur	Nom_Conference	Année	Type_Section	Audience	Institut (Rapport)	Type_Rapport
Référence	x	x					
Contenu	x		x	x			
Article	x		x		x		
Rapport	x					x	
Institut(Auteur)			x			x	x
Type_Section				x			
Audience					x		
Type_Rapport							x

A partir de ces éléments, la phase suivante consiste à permettre la spécification d'un schéma en galaxie pour représenter les structures multidimensionnelles de l'analyse.

2.3 Formalisation des besoins

But. La formalisation des besoins traduit les besoins en un schéma en galaxie. Ce schéma représente, de manière conceptuelle, les structures multidimensionnelles du futur magasin de donnée.

La spécification du schéma en galaxie est effectuée par une transformation de la matrice des besoins. Plusieurs étapes se succèdent pour permettre l'obtention du schéma en galaxie :

- identification des ensembles d'interaction,
- regroupement d'attributs en dimensions,
- spécification des hiérarchies au sein des dimensions.

La première étape permet l'identification des attributs qui vont interagir au sein de mêmes analyses. Ceci aura pour conséquence la spécification des groupes de dimensions, les cliques (cf. chapitre 3, section 2), au sein de la galaxie. La seconde étape permet, au sein de ces groupes, d'associer les attributs en dimensions. Enfin, la troisième permet de finaliser la spécification des dimensions en hiérarchisant les attributs qui la composent.

2.3.1 Identification des ensembles d'interaction

L'identification des ensembles d'interaction permet d'isoler les futures dimensions qui vont interagir entre elles. Ceci ce fait automatiquement à partir de la matrice. Il est possible de modifier la matrice non simplifiée en alternant l'ordre des colonnes et des lignes jusqu'à l'obtention de groupes séparés.

Exemple. A partir du Tableau 15, il est possible d'isoler deux groupes : en mauve (gris foncé) et en vert clair (gris clair) présentés en Tableau 16. Dans ce tableau, les attributs ayant une croix dans la diagonale ont été retirés des lignes. Il s'agit des attributs *Type_section*, *Audience* et *Type_Rapport*.

Tableau 16 – matrice simplifiée avec groupes et simplification des lignes.

Nom des attributs	Nom_Conference	Type_Section	Audience	Nom_Auteur	Année	Type_Rapport	Institut (Rapport)
Référence	x			x			
Contenu		x		x	x		
Article			x	x	x		
Rapport				x			x
Institut(Auteur)					x	x	x

Si les groupes ne sont pas évidents à déterminer, une réorganisation de la matrice (selon un tri par blocs diagonaux) permet une vision plus précise. Ce tri consiste à réordonner les lignes et les colonnes afin de concentrer les cellules avec des croix vers la diagonale de la matrice. Ainsi, il est possible de faire apparaître des blocs diagonaux (qui se chevauchent éventuellement). Ces blocs sont identifiés grâce aux blocs de cellules vides qui se constituent de part et d'autre de la diagonale. Par exemple, les attributs en colonnes *Type_Rapport* et

Institut(*Rapport*) n'ont pas d'interaction avec les attributs en ligne *Référence*, *Contenu* et *Articles*. Ceci permet la désignation du premier bloc que l'on retrouve dans la partie supérieure du Tableau 16. La réorganisation permet l'identification de deux groupes distincts en mauve (gris foncé) et vert clair (gris clair).

Les deux groupes identifiés correspondent d'un côté à l'analyse d'articles scientifiques (*Type_Section*, *Nom_Conférence*, *Audience*, *Référence*, *Contenu*, *Article*) et de l'autre à l'analyse de rapports de projets (*Rapport*, *Institut* (*Auteur*), *Institut* (*Rapport*), *Type_Rapport*). Ces deux groupes ont en commun les informations concernant les auteurs et la date (*Nom_Auteur*, *Année*).

Une fois les groupes d'attributs identifiés, il est possible de constituer les dimensions.

2.3.2 Regroupement des attributs en dimensions

Les dimensions sont des regroupements conceptuels d'attributs. Les attributs sont regroupés selon des considérations sémantiques et selon la précédente Règle 3. Les considérations sémantiques sont obtenues à partir des connaissances extraites des questionnaires et des connaissances métiers du concepteur :

- Les attributs spécifiés en colonnes sont regroupés de manière sémantique en dimensions.
- Les attributs spécifiés en ligne sont soit associés aux attributs d'une dimension déjà existante (si l'attribut est sémantiquement lié aux autres attributs de cette dimension), soit à une nouvelle dimension. Cette association se fait par la connaissance du domaine que doit représenter chaque dimension et chaque attribut.
- Les attributs similaires sont fusionnés (par exemple dans le cas de synonymes).
- Les dimensions sont enrichies par des attributs supplémentaires. Cette étape se fait à partir des connaissances métier, mais peut aussi se faire à partir d'une analyse des sources (cf. Section 3).
- Les attributs sont enrichis de compléments sémantiques. Il s'agit d'ajouter des futurs attributs faibles aux attributs. Lors de cette étape, il se peut aussi que pour des raisons de cohérence, des attributs soit convertis en attributs faibles et associés à d'autres attributs en tant que complément sémantique.

Les attributs sont ainsi regroupés en dimensions et le concepteur nomme chaque dimension en fonction de ce qu'elle représente.

Cas particulier. Le repérage des liens se fait en même temps et aussi de manière sémantique avec les connaissances métiers.

Exemple. En reprenant notre exemple, il est possible de regrouper les attributs dans différentes dimensions :

ARTICLES : *Type_Section*, *Référence*, *Document*
CONFERENCES : *Nom_Conférence*, *Audience*
AUTEURS : *Nom_Auteur*, *Institut*
DATES : *Année*
RAPPORTS : *Rapport*, *Type_Rapport*
INSTITUTS : *Institut*

Notez que les attributs, *Contenu* et *Article*, qui représentaient la même chose (le contenu textuel d'articles scientifiques) ont été fusionnés en un attribut *Document*.

Dans notre exemple, il est question d'instituts d'auteurs et d'instituts émetteurs de rapports (qui peuvent être identiques). Il s'agit de deux dimensions liées par un lien. De manière similaire, les références contenues dans un article sont des éléments qui pointent vers des contenus d'articles. Au final, les liens suivants sont obtenus :

AUTEURS : *Institut* → *INSTITUTS* : *Institut*
ARTICLES : *Référence* → *ARTICLES* : *Document*

Les dimensions avec les attributs suivants, sont obtenues après un enrichissement d'attributs supplémentaires :

ARTICLES : *Paragraphe*, *Type_Paragraphe*, *Section*, *Type_Section*, *Titre_Section*, *Document*, *Titre_Document*.
CONFERENCES : *Conférence*, *Taux_Acceptation*, *Editeur*, *Nom*, *Série*, *Audience*.
AUTEURS : *Auteur*, *Nom*, *Statut*, *Equipe*, *Institut*.
DATES : *Date*, *Mois*, *Libellé_Mois*, *Année*.
RAPPORTS : *Rapport*, *Titre_Rapport*, *Type_Rapport*
INSTITUTS : *Institut*, *Nom*, *Pays*.

Lors de l'étape suivante, ces attributs sont hiérarchisés selon une ou plusieurs hiérarchies.

2.3.3 Spécification des hiérarchies

La formalisation des hiérarchies est une étape complexe. Cette formalisation repose sur la confrontation de plusieurs informations :

- les connaissances métier ;
- les dépendances des données dans les sources ;
- l'analyse de valeurs des sources.

Les connaissances métier constituent une base très riche pour créer la hiérarchisation des attributs. Il s'agit de connaissances générales ou spécifiques à un domaine particulier tel que l'organisation des attributs temporels : jours, mois, années. Dans le cadre de l'analyse du contenu de documents, les connaissances métier indiquent la structuration des documents : des paragraphes, regroupés en sous-sections, elles-mêmes regroupées en sections.

Les dépendances fonctionnelles des données sont repérées lors de l'analyse des sources qui a lieue de manière simultanée (cf. section 3). Il s'agit de repérer les liaisons mono-valuées (une contrainte d'intégrité traduite par des cardinalités [1..1]→[1..n]). Dans notre exemple, un auteur dépend d'un seul institut [1..1] et un institut regroupe un certain nombre d'auteurs [1..n]. Ces liaisons sont détectées à partir de la définition de la structure des sources (par exemple une DTD dans le cas de documents XML).

A partir de l'analyse détaillée du contenu des sources, notamment par l'analyse des valeurs de certains éléments, il est possible de déterminer des dépendances hiérarchiques entre les données. Ainsi il est possible d'analyser les sources et de repérer par exemple des regroupements de produits vendus en famille de produits ou encore le regroupement de fabricants de produits par catégories.

A partir de ces trois sources d'informations, les hiérarchies d'attributs sont définies pour chaque dimension :

- Les attributs de chaque dimension sont regroupés de manière hiérarchique avec la définition du niveau le plus fin qui sera l'attribut racine de chaque dimension. Chaque attribut ainsi associé devient alors un paramètre.

- Les attributs qui n'ont pas été associés à des hiérarchies précédemment sont alors convertis en attributs faibles et associés à un paramètre.

Au final, un schéma conceptuel multidimensionnel en galaxie représentant les besoins d'analyse spécifiés par les décideurs est obtenu.

Exemple. Dans notre exemple, la hiérarchie de la dimension *ARTICLES* est spécifiée grâce aux connaissances métier sur la structure « standard » des articles scientifiques. L'agencement suivant des paramètres est obtenu :

$Param^{HS} = <Paragraphe, Section, Document>$

Ces paramètres sont associés à des attributs faibles, qui complètent les informations des paramètres :

$Paragraphe \rightarrow \{Type\_Paragraphe\}$;
$Section \rightarrow \{Type\_Section, Titre\_Section\}$;
$Document \rightarrow \{Titre\_Document\}$;

Au final, la galaxie G_1 est constituée (cf. Figure 59).

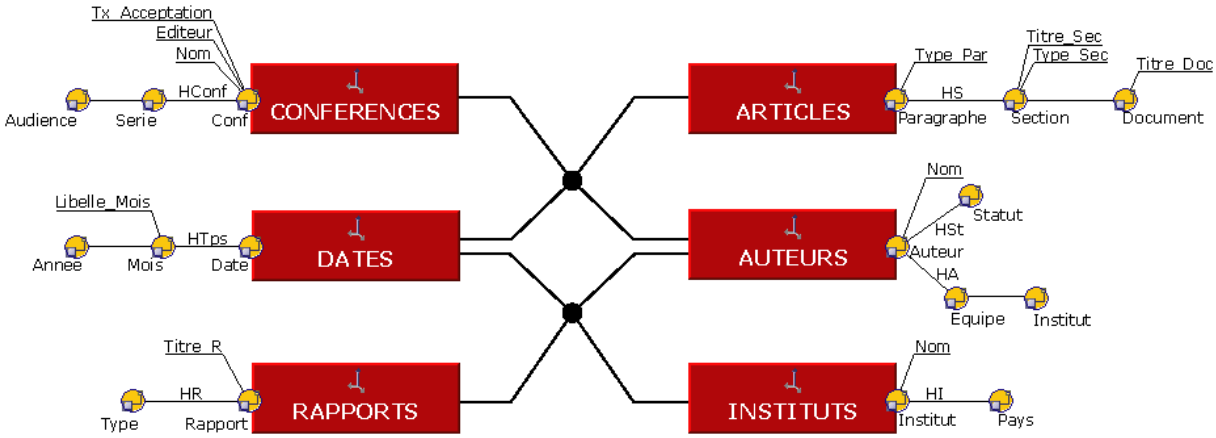


Figure 59 – Schéma multidimensionnel en galaxie obtenu.

2.4 Bilan concernant la spécification des besoins

Cette section a présenté une démarche qui à partir des besoins utilisateur permet la formalisation de ces besoins au travers d'un schéma multidimensionnel en galaxie. Pour y parvenir trois étapes sont exécutées :

- la collecte des besoins ;
- la spécification des besoins au moyen d'une matrice ;
- la traduction des besoins en un schéma conceptuel multidimensionnel.

Toutefois, rien ne garantit que les structures multidimensionnelles définies par le schéma puissent être exploitables par la suite. Pour obtenir cette garantie, il est nécessaire de confronter le schéma théorique aux données des sources. Cette confrontation permettra de vérifier leur concordance autorisant l'alimentation des futures structures du magasin.

Ainsi, parallèlement à la modélisation des besoins, les sources sont analysées pour préparer une confrontation entre leur schéma logique et le schéma conceptuel multidimensionnel.

3 Analyse des sources

But. Le but de l'analyse des sources est de pouvoir identifier les différents éléments qui les composent afin de déterminer ceux nécessaires pour répondre aux besoins d'analyse. Cette analyse permet de préparer la confrontation avec le schéma multidimensionnel pour s'assurer de sa compatibilité avec les sources.

Hypothèses concernant les sources. Dans notre approche nous considérons des sources composées de documents XML essentiellement constitués de données textuelles et conformes vis-à-vis d'une même structure commune. Cette structure est spécifiée selon une grammaire définie au format DTD. Les processus d'extraction, de modification et de chargement (ETL) sortant du cadre de ce mémoire de thèse, les documents sont supposés nettoyés et conforme à leur structure.

Cette section commence par une analyse générale des types de données au sein d'une source XML suivie de règles à suivre pour permettre une identification aisée des éléments constitutifs des sources.

3.1 Différents types de données au sein d'une source XML

Un document XML est constitué d'un contenu agencé selon une structure arborescente. Nous définissons le *contenu global* d'un document XML qui correspond à l'intégralité des données contenues dans un document XML. Plus particulièrement, nous considérons trois types de données distinctes au sein du contenu global de ces documents :

- les *méta-données* : ce sont les données complémentaires au contenu du document. Les méta-données sont généralement disséminées au sein de la structure globale du document XML ;
- le *contenu* : ce sont les données d'intérêt pour l'analyse. Il s'agit du contenu du document XML privé des méta-données et de la structure associée aux méta-données ;
- la *structure* (du contenu) : il s'agit de la structure XML qui structure uniquement le contenu du document. Cela correspond à la structure du document XML privé de la structure associée aux méta-données.

Le contenu d'un document textuel représente la majeure partie d'un document XML. Par exemple, dans le cas d'un article scientifique, le corps de l'article représente le contenu du document. Parfois, ce contenu est profondément enfouis dans la structure arborescente du document XML.

Exemple. La Figure 60 montre une DTD de documents XML représentée sous forme arborescente avec les trois types de données identifiés (le symbole '+' indique une cardinalité [1..*] concernant l'élément en question). Il est possible de retrouver les méta-données en rose (gris foncé), le contenu en bleu ciel (gris clair) et la structure qui lui est associé.

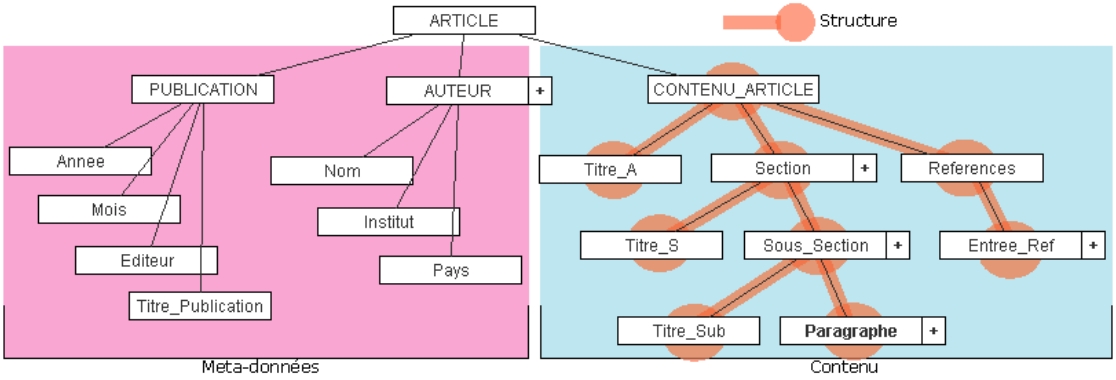


Figure 60 – Exemple d'identification des composants des documents sources (méta-données, contenu et structure du contenu).

Ainsi à partir de la structure complète du document XML on obtient deux structures. La première, qui correspond à la structure des méta-données, sera utile pour l'intégration de ces méta-données dans les différentes dimensions. Les informations associées à cette structure peuvent être aussi employées pour effectuer des associations avec des sources de données externes (cf. section 5). La seconde structure est la structure du contenu du document XML. Cette structure va servir pour constituer une dimension documentaire.

Quelques règles simples permettent l'identification des différents types de données dans les documents XML.

3.2 Règles pour l'analyse des sources

But. Les règles qui suivent permettent d'identifier les types de données au sein des différents éléments des documents XML sources. L'intégration s'en retrouve alors simplifiée car les données sont assignées de manière plus fiable aux éléments conceptuels du modèle.

L'analyse des sources se fait en observant les règles selon une approche semi-automatique où l'utilisateur intervient de temps à autre.

Règle 1. *Localisation du contenu d'un document XML orienté document* : Les données les plus intéressantes (pour notre approche) au sein d'un document XML sont relatives au contenu. A l'inverse de documents XML représentant des transactions ou bien des messages structurés, ce contenu est localisé dans les feuilles au sein de la structure arborescente XML ou dans un niveau très proche des feuilles.

Règle 2. *Localisation de la structure du contenu* : pour obtenir la structure du contenu d'un document orienté document, il suffit de remonter l'arborescence au dessus des éléments qui représentent le contenu.

Règle 3. *Localisation des méta-données* : au sein d'un document XML orienté document de nombreuses méta-données sont associées au contenu. Ces données sont souvent hiérarchisées ce qui permet de construire des dimensions. De plus à l'inverse du contenu, les méta-données sont associées à des éléments proches de la racine du document.

Exemple. Dans notre exemple (cf. Figure 60), le contenu d'un document XML est identifié par les balises représentant les paragraphes de documents (*Paragraphe*). De son côté, la structure est constituée de balises qui s'imbriquent les unes dans les autres : article (*Contenu_Article*), *Section*, *Paragraphe*.

Les auteurs d'un document sont des méta-données. Dans le cadre de publications scientifiques, ces auteurs (balise *Nom*, fille de la balise *Auteur*) sont associés à leur institut de recherche ou entreprise (*Institut*) et un pays (*Pays*).

3.3 Bilan de l'analyse des sources

L'analyse des sources permet l'identification des données présentes et la modélisation des besoins se traduit par la constitution d'un schéma conceptuel multidimensionnel en galaxie. L'intégration de l'analyse des sources et des besoins des décideurs permet de prendre en compte toutes les données pertinentes pour la prise de décision [Trujillo et al., 2003].

Toutefois afin de s'assurer que le schéma conceptuel représentant les éléments multidimensionnels du magasin peuvent être alimentés par les données contenues dans les sources, une étape de confrontation est nécessaire.

4 Étape de confrontation

Lors de l'intégration de documents XML dans un système OLAP, l'utilisateur analyse d'un côté les sources via leur représentation logique, puis il les confronte à la représentation des besoins d'analyse modélisés par un schéma multidimensionnel en galaxie.

But. Le but de l'étape de confrontation est de permettre l'alimentation du magasin avec les données sources. Pour permettre cette alimentation, le schéma en galaxie qui représente les structures multidimensionnelles du magasin doit être compatible avec le schéma logique des sources de données. C'est-à-dire qu'il existe au sein des sources les données nécessaires pour alimenter les différents attributs composant la galaxie.

Durant l'étape de confrontation, l'utilisateur confronte la représentation logique des sources et la représentation conceptuelle des besoins d'analyse représenté par un schéma en galaxie (cf. Figure 61). Les éléments composant les sources sont associées aux éléments des structures multidimensionnelles de la galaxie (le processus est détaillé en section 4.2). En cas d'incompatibilités, l'utilisateur dispose de deux alternatives :

- modifier les sources, en les enrichissant à partir de données complémentaires annexes ;
- modifier le schéma multidimensionnel, par un affinage de la galaxie.

Une fois les modifications effectuées, une nouvelle étape de confrontation a lieu et le processus est itéré jusqu'à ce que le schéma multidimensionnel et les sources soient compatibles.

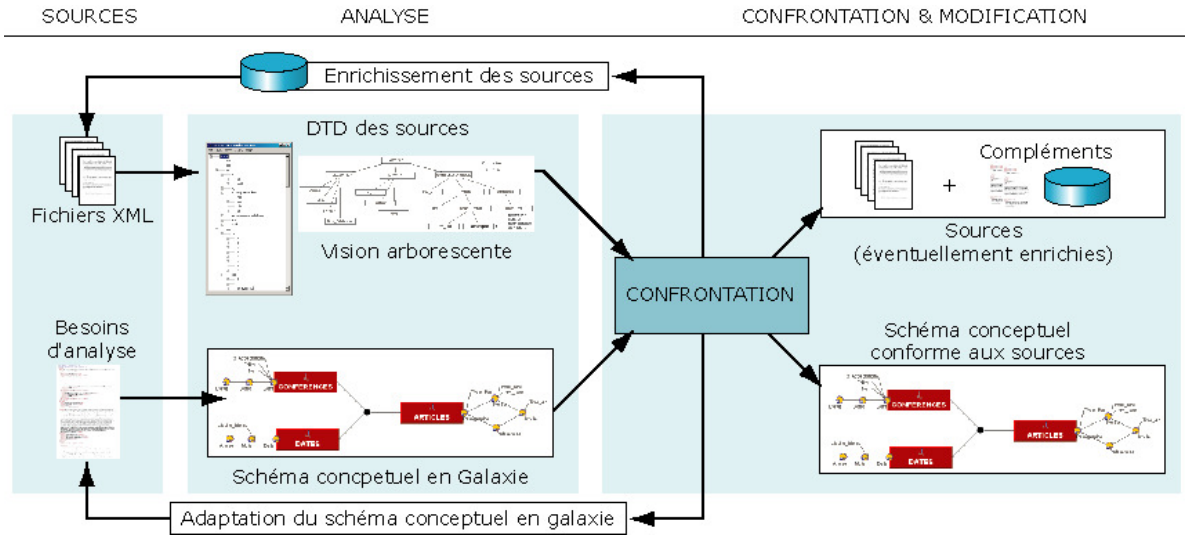


Figure 61 – Etapes de la confrontation des besoins d'analyse et des sources.

4.1 Confrontation et incompatibilités

La confrontation permet de détecter des données nécessaires pour alimenter la galaxie qui seraient absentes des sources (une incompatibilité entre les deux schémas). En effet, si des données spécifiées par le schéma en galaxie s'avèrent absentes des sources, tout un ensemble d'analyses risque de ne pouvoir être effectué.

Problématiques de confrontation. La confrontation peut conclure sur une incompatibilité entre les sources et le schéma multidimensionnel. Ceci est engendré par différents problèmes :

- Les données requises par le schéma multidimensionnel sont présentes dans les sources, mais elles ne disposent pas de la même structuration hiérarchique que celle requise par les hiérarchies des dimensions de la galaxie.
- Les données requises sont carrément absentes des sources.
- Les données requises sont présentes, structurées de manière hiérarchique mais ne respectent pas l'organisation prévue par le modèle conceptuel. Par exemple, des données avec des liaisons [*..*] au lieu de [1..*].
- Il existe aussi des problèmes de sémantique tels que les problèmes de synonymie, des erreurs de saisie… Ces problèmes, relevant plus du domaine du traitement automatique du langage, sortent du cadre de ce mémoire et ne seront pas traités.

Incompatibilités mineures. Si les données ne respectent pas la même structuration hiérarchique, une conversion sera nécessaire. Cette conversion peut être exécutée par un traitement XQuery ou XSLT afin de réorganiser les données XML selon une structuration adéquate. Par exemple, modifier l'emplacement d'un élément au sein de l'arborescence XML. Toutefois il faut noter qu'il se peut que cette solution implique une perte de données. Dans le cas où les données ne respectent pas la même organisation au sein d'une hiérarchie, il faudra convertir la liaison [*..*] en une liaison [1..*] avec, si possible, le minimum de perte d'information. Sinon, dans l'impossibilité de convertir une telle liaison, il sera nécessaire d'envisager la gestion de hiérarchies non-strictes [Malinowski & Zimányi, 2006] mais elle est très problématique. En effet l'emploi de ces hiérarchies ne permet pas une cohérence au sein des processus d'agrégation. Par exemple, si un auteur dépends de un ou plusieurs instituts, alors le nombre total d'auteurs sera inférieur à la somme du nombre total des auteurs de chaque institut.

Incompatibilités majeures. Dans le cas général, il est nécessaire soit d'enrichir les sources, soit de réviser le schéma en galaxie. Une fois ces changements effectués, une nouvelle itération de confrontation est effectuée pour détecter de nouvelles incompatibilités. Ce processus itératif se répètera tant que des incompatibilités entre les sources et les besoins d'analyse (le schéma conceptuel multidimensionnel) existeront. A l'issue de la confrontation, l'utilisateur dispose de sources (éventuellement enrichies) et d'un schéma conceptuel multidimensionnel, compatibles entre eux.

Pour détecter les problèmes lors de la confrontation, deux phases sont effectuées (cf. Figure 62). La première est l'établissement d'associations entre les éléments des sources et ceux du schéma multidimensionnel cible. La seconde phase est une phase d'affinage du schéma multidimensionnel.

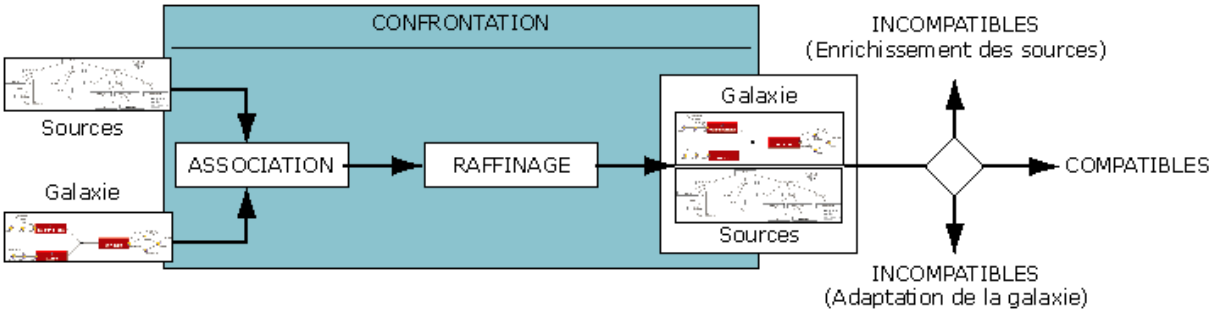


Figure 62 – Les deux phases durant la confrontation.

4.2 Association, détection des incompatibilités

La première phase, l'association, consiste à relier un élément dans le schéma des sources (représenté par une arborescence des éléments XML) à un attribut du schéma multidimensionnel (représenté par un schéma en galaxie). L'établissement de ces liaisons permet d'indiquer les données des sources à employer pour alimenter les structures multidimensionnelles du magasin modélisées par un schéma en galaxie (cf. Figure 63).

Deux nuances sont à prévoir :

- il peut être nécessaire de combiner plusieurs éléments sources entre eux pour obtenir les données nécessaires pour constituer les données d'un attribut du schéma.
- A l'inverse, il peut être nécessaire de subdiviser les données d'un élément afin d'alimenter un ou plusieurs attributs.

Ces deux nuances permettent de gérer le cas où il n'y aurait pas de liaison [1..1] entre les éléments et les attributs.

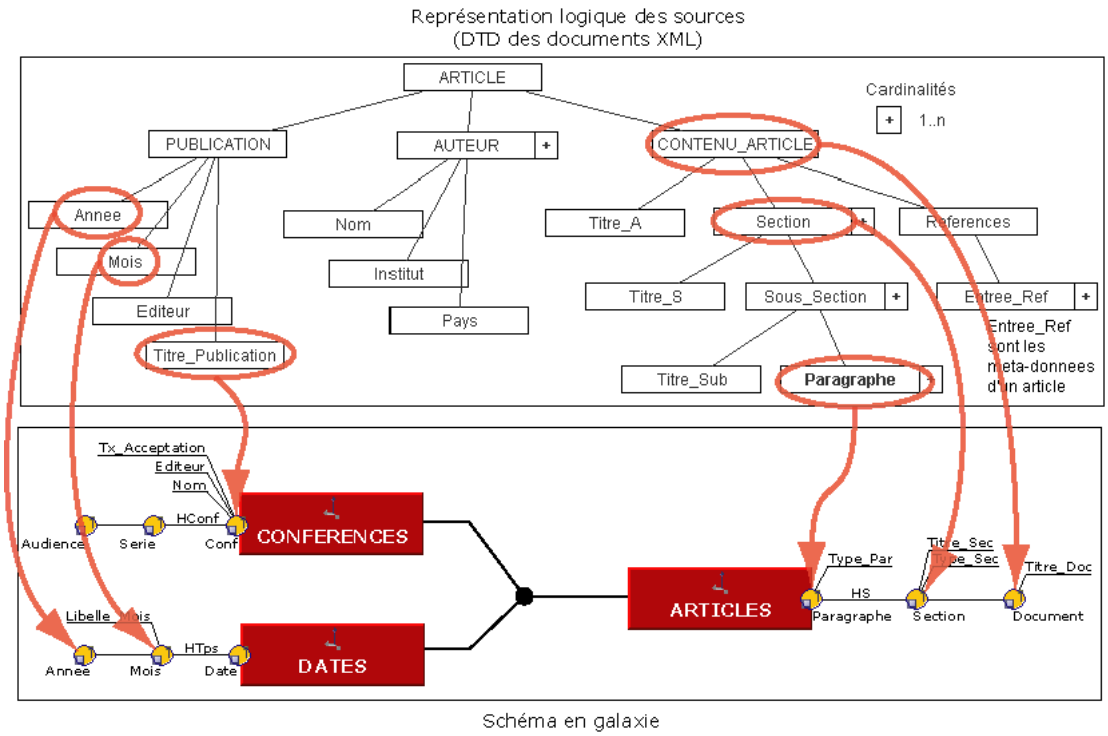


Figure 63 – Exemple de liaisons entre des éléments de la représentation arborescente des sources et les éléments de la galaxie (seules trois dimensions sont représentées).

Quelques règles sont employées pour permettre une assignation entre les éléments. Ces règles permettent d'identifier les types de données qu'il existe au sein des différents éléments des documents XML sources. Dans les documents XML un ensemble d'éléments peuvent être analysés pour définir la structuration des dimensions qui vont constituer le modèle :

Règle 4. *Contraintes structurelles* : Les données hiérarchisées des dimensions (les paramètres) peuvent être trouvées dans les nœuds avec des relations un-vers-plusieurs [1..*] avec ses descendants.

Ces contraintes représentent l'impact de la hiérarchisation sur un modèle multidimensionnel des éléments au sein des documents XML.

Exemple. Une section peut être liée à une ou plusieurs sous-sections qui elles-mêmes peuvent être liées à un ou plusieurs paragraphes.

Règle 5. *Contraintes sémantiques* : Les données hiérarchisées des dimensions (les paramètres) peuvent être trouvées au même niveau dans l'arborescence XML. Une contrainte sémantique identifie le parent du descendant alors que les deux éléments sont au même niveau.

Exemple. Dans une balise *publication*, le mois et l'année (balises *Mois* et *Annee*) sont stockés au même niveau. Toutefois ces deux éléments n'ont pas le même niveau de granularité. Une contrainte sémantique est : les années sont composées de mois.

Il existe plusieurs types de contraintes sémantiques évidentes qui peuvent être traitées de manière semi-automatique. En voici quelques unes :

- les données temporelles (jours, semaine, mois, années),
- les données géographiques (villes, département, régions) pour la France.

Pour les autres catégories, l'utilisateur doit intervenir directement.

Une fois les associations spécifiées, il se peut que des éléments du schéma multidimensionnel en galaxie ne soient pas liés. Il s'agit d'autant de cas d'incompatibilité qui doivent être résolus. Certaines incompatibilités mineures peuvent être résolues simplement par un léger affinage du schéma en galaxie.

4.3 Affinage, résolution des incompatibilités mineures

L'affinage d'une galaxie représente des règles de nettoyage qui permettent de commencer à mettre en phase la structure des données des sources et la structuration des éléments multidimensionnels de la galaxie.

Règle 6. *Ajustement des attributs faibles* : tout attribut faible qui n'est pas relié à un nœud de la structure des sources est retiré de la représentation conceptuelle.

Règle 7. *Ajustement des paramètres* : tout paramètre qui n'est pas relié à un nœud de la structure des sources est retiré de la représentation conceptuelle sauf si celui-ci est lié à un attribut faible qui lui-même est lié à un nœud de la structure des sources. Le paramètre sera alors constitué de données d'identification (telles que des valeurs clés).

Une fois ces règles suivies, si d'autres incompatibilités sont présentes, il est possible soit de modifier en profondeur le schéma en galaxie par la modification des besoins d'analyse (cf. section 2), soit d'ajouter aux sources les données complémentaires manquantes (cf section 5).

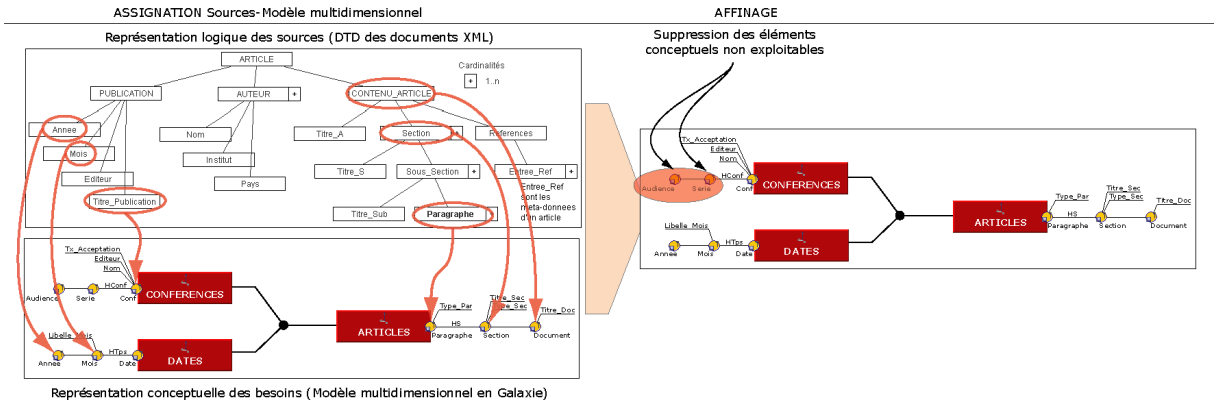


Figure 64 – Association des élements des sources aux éléments conceptuels du modèle (seules les flèches sur les paramètres sont indiquées).

Il faut noter toutefois que l'affinage ne sera pas nécessairement complètement appliqué. En effet, dans le cas d'un enrichissement des sources a posteriori (cf. section 5), certain attributs ne seront alimentés que tardivement. Une nouvelle étape d'association sera alors effectuée entre les structures du schéma conceptuel multidimensionnel restées vides et les futures sources complémentaires.

4.4 Bilan et résumé de la confrontation

L'étape de confrontation est la fin d'un processus décomposé en trois parties. Premièrement les sources et les besoins sont analysés et traduits en schéma logique pour les sources et schéma conceptuel multidimensionnel pour les besoins d'analyse. Deuxièmement ces deux schémas sont confrontés et un processus itératif permet de les rendre compatibles.

En résumé, les étapes de l'analyse des sources sont les suivantes :

- Identification des différents éléments des sources : méta-données, contenu et la structure du contenu. Car les éléments représentant le contenu et sa structure vont constituer une dimension alors que les méta-données seront réparties en autant de dimensions que nécessaire.
- Confrontation des besoins en méta-données : identifier méta-données manquantes. Ce cas arrive avec des éléments XML optionnels au sein de la structure du document. Ainsi il se peut que certaines méta-données ne soient pas complètement spécifiées.

Parallèlement la conversion des besoins d'analyse en un schéma conceptuel suit les étapes suivantes :

- Identification des différentes données à analyser : données documentaires, données numériques factuelles classiques, données dimensionnelles…
- Spécification de la structure des données documentaires (organisation du contenus de documents orientés documents qui seront à analyser).
- Hiérarchisation du contenu documentaire (génération des dimensions documentaires).
- Spécification des indicateurs d'analyse classiques (génération de dimensions simples adaptées).
- Spécification des méta-données associées aux données documentaires.
- Hiérarchisation des méta-données (génération des dimensions de type méta-données).
- Regroupement des dimensions en fonction des objectifs d'analyse (spécification des nœuds centraux de la galaxie).

La confrontation des deux schémas permet de les rendre compatibles via le processus itératif qui modifie soit le contenu des sources, soit la spécification du schéma multidimensionnel, soit les deux.

Dans certains cas, la confrontation entre les sources et le schéma multidimensionnel en galaxie se conclut par la nécessité d'enrichir les sources pour permettre l'alimentation des structures multidimensionnelles, cette étape est décrite plus en détails dans la section suivante.

5 Enrichissement des sources

Le contenu des documents peut ne pas satisfaire les besoins d'analyse or, dans certains cas, il ne peut être envisagé de revoir à la baisse les objectifs d'analyse. Par conséquent la seule solution est d'ajouter les données manquantes à partir de sources auxiliaires complémentaires.

But. Le but de l'enrichissement des sources est de disposer d'informations complémentaires par rapport aux méta-données contenues dans les documents sources ou bien de combler des lacunes de ces sources, par exemple, dans le cadre de méta-données partielles ou incomplètes.

L'étape de confrontation fait ressortir les données manquantes. Deux alternatives sont alors envisageables :

- l'enrichissement des sources a priori : à savoir enrichir les sources avant l'alimentation des données dans le magasin de données ;
- l'enrichissement a posteriori : à savoir, enrichir les données chargées dans le magasin de données (après le processus d'alimentation).

Les deux sous-sections suivantes exposent chaque alternative.

5.1 Enrichissement des sources a priori

L'enrichissement des sources a priori est une des étapes (optionnelles) pour la résolution de conflits dans le cadre de la confrontation des schémas sources et d'analyse en galaxie.

La Figure 65 présente les étapes de cet enrichissement. Les données des sources (documents XML orientés documents) sont liées aux données des sources complémentaires par des index de liaisons. Les données des sources complémentaires sont adaptées pour être compatible (en terme de structure et de format) avec les documents sources. Des structures intermédiaires sont employées pour permettre la fusion entre les différentes sources, les sources sont alors définies selon une présentation uniforme (schéma et formats compatibles). C'est-à-dire que la structure de l'une est adaptée pour être compatible avec l'autre, soit avec une troisième structure spécialement conçue pour la fusion. Les sources fusionnées sont ensuite chargées dans les structures multidimensionnelles du magasin.

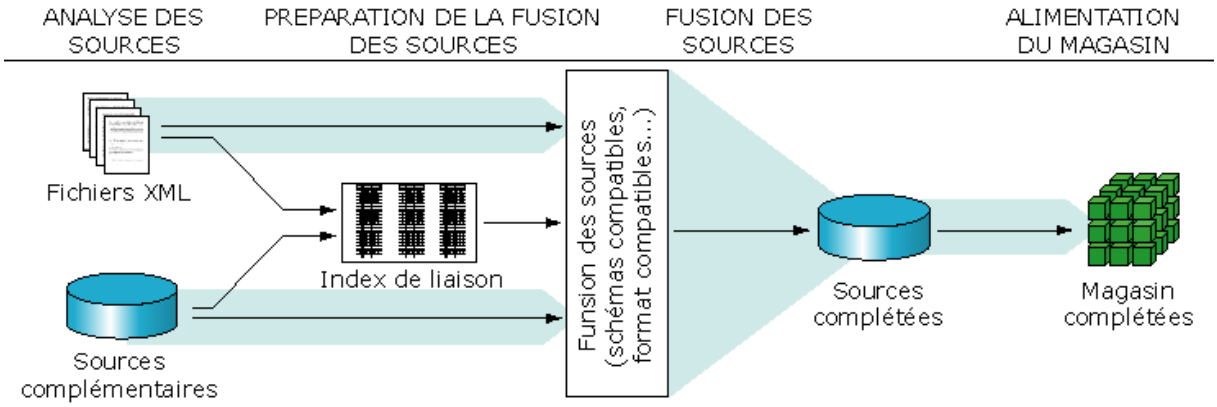


Figure 65 – Enrichissement des sources a priori.

Avantages/Inconvénients. Dans le cadre de l'enrichissement a priori, la fusion des sources principales et auxiliaires en une seule source a l'avantage de fournir une unique source uniformisée. L'alimentation du magasin se fait en une seule fois et un schéma global des sources est disponible. Toutefois ceci implique un fort volume de données intermédiaires pour le traitement des données en vue de leur fusion. Le format d'origine des sources est perdu à moins de maintenir une version d'origine dupliquée.

5.2 Enrichissement des sources a posteriori

Dans le cadre de l'enrichissement a posteriori, les sources complémentaires ne sont intégrées que lors de l'alimentation du magasin (cf. Figure 66). Concrètement, l'alimentation du magasin se fait en deux phases : la première consiste à alimenter les structures en données issues des sources principales (documents XML) ; la seconde consiste a fusionner les données issues de sources auxiliaires dans un second temps dans les structures multidimensionnelles du magasin déjà définies mais seulement partiellement alimentées.

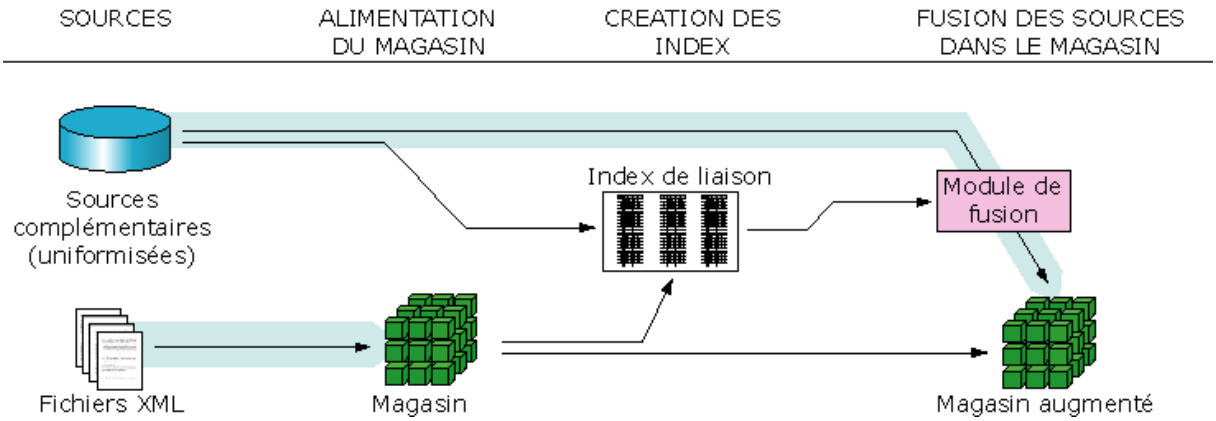


Figure 66 - Enrichissement des sources a posteriori.

Avantages/Inconvénients. Dans le cadre du processus a posteriori, de multiples sources sont à gérer durant le processus d'enrichissement (une source principale et les sources auxiliaires). Ceci limite les redondances en terme de données ainsi que les traitements associés. Bien que le schéma global soit disponible, il n'est que virtuel car les données ne sont pas nécessairement uniformisées. Enfin l'alimentation du magasin de données se fait en plusieurs étapes, mais cela permet un chargement progressif. Le chargement progressif nécessite une adaptation des traitements du magasin afin de pouvoir cohabiter avec des données partielles et incomplètes.

5.3 Exemple d'enrichissement

Dans notre exemple d'analyse d'articles scientifiques, il existe deux sources que nous désirons employer :

- l'ensemble des publications des journaux de l'IEEE jusqu'en 2004 au format XML[16].
- une base de données des publications scientifiques principalement en systèmes d'information et en base de données : DBLP[17].

Dans la collection d'articles de l'IEEE, les auteurs sont tous repérés dans un élément `<au>`. Cet élément contient deux sous éléments qui contiennent le nom `<snm>` et le prénom `<fnm>` de l'auteur. De son côté la base de DBLP (disponible au format XML) identifie un auteur par l'élément `<author>`, contenant à la fois le prénom et le nom.

Il est possible de constituer un index entre ces deux sources de données par une comparaison des chaînes de caractères du nom de l'auteur reconstitué par la fusion de l'élément prénom et de l'élément nom pour la base de l'IEEE avec le nom d'auteur contenu dans la base DBLP. Avec ces données, il est évident que les synonymes ne peuvent être gérés.

Dans cet exemple, l'index doit permettre la liaison entre des éléments provenant de différents fichiers XML. Ainsi il sera constitué (par exemple) d'une relation avec deux attributs. Chacun étant un chemin XPath permettant l'identification d'un élément dans chaque source.

Cette liaison permettra de joindre des données correspondant à des articles complets et des listes de publications (sans contenu). Dans notre exemple, la collection de documents étant

[16] collection de test employée jusqu'en 2005 par INEX : http://inex.is.informatik.uni-duisburg.de/
[17] http://www.informatik.uni-trier.de/~ley/db/

limitée à l'IEEE, l'ajout de ces données annexes permet l'adjonction de publications d'autres éditeurs (sans toutefois ajouter leur contenus).

Si les données sources employées pour créer l'index ont été extraites de l'entrepôt, il sera possible alors d'effectuer un enrichissement a priori. Si les données sources pour la constitution de l'index ont directement été prises du magasin, il s'agira alors d'un enrichissement a posteriori.

5.4 Bilan et choix du type d'enrichissement

Afin de combler des lacunes en méta-données dans les sources principales, nous suggérons la solution d'enrichir les sources par des données complémentaires. Cet enrichissement peut se faire avant ou après l'intégration des données dans le magasin.

Le choix de l'alternative repose sur des questions de disponibilité :

- disponibilité en terme d'espace de stockage ;
- disponibilité des sources auxiliaires ;
- disponibilité a posteriori des sources principales.

Contrairement à l'intégration a posteriori, la première solution, l'intégration a priori, nécessite un grand volume de stockage par rapport aux volumes des sources. La seconde solution permet de pallier des problèmes d'accès aux données auxiliaires et permet leur intégration progressive dans le magasin. Enfin si l'utilisateur requiert un accès aux données sources principales d'origine, il doit, soit disposer d'un volume de stockage de données adaptées dans le cadre de la solution a priori, sinon il devra envisager la seconde solution pour ne pas altérer les fichiers des sources.

Remarque. Dans certains cas, il peut être nécessaire de combiner les deux méthodes d'enrichissement des données. Par exemple, dans le cas ou certaines données complémentaires ne sont pas disponibles lors de l'étape de fusion des sources.

À l'issue de l'étape d'enrichissement, une nouvelle itération du processus de confrontation est effectuée.

6 Étape d'alimentation du magasin

A partir des liaisons établies précédemment, les données sont alimentées dans le magasin défini par le schéma en galaxie.

But. Le but de cette étape est de fournir les données qui vont être contenues dans les structures multidimensionnelles du magasin définies par le schéma conceptuel en galaxie.

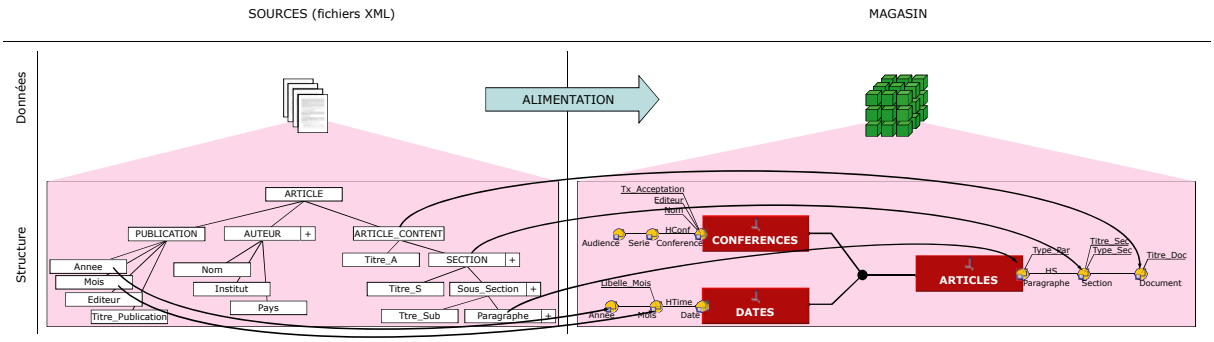


Figure 67 – Alimentation du magasin avec les données issues de documents sources.

Le processus d'alimentation suit simplement les liaisons établies entre les nœuds des documents XML sources et les attributs du schéma multidimensionnel en galaxie. Ceci permet de constituer des requêtes XQuery pour l'intégration des données dans le magasin.

Alimentation à partir de multiples sources. Bien que ce cas ne soit pas considéré dans les présents travaux, il est envisageable d'intégrer des données issues de plusieurs sources. Par exemple, deux collections de documents, décrites par des DTD différentes. Dans ce cas, il est nécessaire que le schéma multidimensionnel en galaxie soit compatible avec les deux. Le processus de confrontation et d'association est effectué plusieurs fois avec chacune des sources.

Sources à DTD multiples. Dans le cadre de collection de documents, décrites par de multiples DTD complémentaires, un schéma global est considéré. Dans ce cas une DTD fusionnée est présentée permettant l'intégration des données. Si les DTD ne peuvent être fusionnées, l'utilisateur se retrouve en réalité en face de multiples sources à DTD différentes (cf. le cas décrit ci-dessus).

Filtrage des sources. Dans certains cas, des limites de l'environnement d'analyse peuvent imposer un filtrage des données sources. Par exemple, dans notre cas, l'emploi de la fonction d'agrégation AVG_KW est conditionné à un vocabulaire contrôlé. En effet cette fonction (cf. le chapitre précédent) utilise une ontologie de domaine qui hiérarchise les mots-clefs et les thèmes d'un domaine précis. En l'occurrence dans l'exemple présenté il s'agissait des systèmes d'informations. Ainsi analyser des documents issus d'autres domaines, avec ce type de fonction, produirait peu de résultats pertinents. Ainsi il est parfois nécessaire d'envisager un filtrage des sources en fonction de critères tels que cette ontologie de domaine.

7 Bilan

L'objectif de ce chapitre est de proposer des éléments méthodologiques pour permettre l'intégration de données issues de documents XML dans un environnement OLAP permettant l'analyse de leur contenu.

L'intégration de documents en fait via quatre étapes :

- La représentation des sources et la représentation des besoins au moyen de schémas suivi de la confrontation de ces schémas pour la résolution des conflits.
- L'enrichissement éventuel des sources ou la modification du schéma multidimensionnel pour permettre l'obtention d'une compatibilité lors de la phase de confrontation.

- L'établissement de liaisons entre les documents orientés documents et le schéma conceptuel multidimensionnel.
- L'alimentation des structures multidimensionnelles, définies dans le magasin par le schéma conceptuel multidimensionnel en galaxie, par les données des sources.

Ces quatre étapes permettent l'obtention d'un magasin de données permettant une analyse du contenu de documents.

Le chapitre suivant présente un prototype que nous avons développé afin de valider nos propositions.

8 Références

[Agrawal et al., 1997] Rakesh Agrawal, Ashish Gupta, Sunita Sarawagi, "Modeling Multidimensional Databases", *13th Intl. Conf. on Data Engineering* (*ICDE*), IEEE Computer Society, p. 232–243, 1997.

[Ghozzi, 2004] Faiza Ghozzi *Conception et manipulation de bases de données dimensionnelles à contraintes*, Thèse de doctorat, Université Paul Sabatier, Toulouse 3 (France), novembre 2004.

[Golfarelli et al., 1998] Matteo Golfarelli, Dario Maio, Stefano Rizzi, "The Dimensional Fact Model: A Conceptual Model for Data Warehouses", invited paper, *Intl. Journal of Cooperative Information Systems* (*IJCIS*), vol.7(2-3), World Scientific Publishing, p. 215–247, juin & septembre1998.

[Inmon, 1996] W. H. Immon, *Building the Data Warehouse*, John Wiley and sons, New York, NY, ISBN : 0764599445, 1996 2ème ed., 4ème ed. 2005.

[Kimball, 1996] Ralph Kimball, *The data warehouse toolkit: Practical Techniques for Building Dimensional Data Warehouses*, John Wiley and Sons, ISBN : 0-471-15337-0, 1996, 2ème ed. : Ralph Kimball, Margaery Ross, *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling*, *2nd Edition*, John Wiley & Sons, 2002.

[Malinowski & Zimányi, 2006] Elzbieta Malinowski, Esteban Zimányi, "Hierarchies in a multidimensional model: From conceptual modeling to logical representation", *Data & Knowledge Engineering* (*DKE*), vol.59(2), Elsevier, p. 348–377, novembre 2006.

[Nassis et al., 2006] Vicky Nassis, Tharam S. Dillon, Rajugan Rajagopalapillai, Wenny Rahayu, "An XML Document Warehouse Model", 12*th Intl. Conf. on Database Systems for Advanced Applications* (*DASFAA*), LNCS 3882, Springer, p. 513–529, 2006.

[Trujillo et al., 2003] Juan Trujillo, Sergio Luján-Mora, Il-Yeol Song, "Applying UML For Designing Multidimensional Databases And OLAP Applications", *Advanced Topics in Database Research*, Keng Siau (Ed.), vol.2, Idea Group Publishing (IGP), p. 13–36, 2003.

CHAPITRE VI
Implantation et validation

Résumé du chapitre

Ce chapitre présente l'implantation des propositions de ce mémoire de thèse au sein d'un prototype : GraphicOLAPXML. Ce prototype est constitué d'une application java qui emploie du SGBD relationnel Oracle 10g2 étendu pour la gestion du XML. A partir d'un entrepôt de données et de documents XML, un magasin modélisé par un schéma en galaxie est constitué. Des analyses sont exprimées à partir de la manipulation des concepts multidimensionnels représentés via l'interface graphique de l'outil.

Sommaire

CHAPITRE VI : Implantation et validation

« La théorie, c'est quand on sait tout et que rien ne fonctionne. La pratique, c'est quand tout fonctionne et que personne ne sait pourquoi. […] »

— Albert Einstein.

1 Introduction et architecture

Le but de notre implantation est de fournir un environnement d'analyse en ligne de données issue de documents XML. Cet environnement représente un magasin de données reposant sur un entrepôt de documents XML. L'environnement permet à l'utilisateur d'analyser les documents relativement aux méta-données auxquelles les documents sont associés, mais aussi le contenu des documents.

Notre prototype repose principalement sur un magasin de données construit sur un système de gestion de bases de données (SGBD). Le magasin est consulté par une application cliente qui permet la spécification des analyses et la restitution auprès de l'utilisateur.

L'architecture générale du prototype (GraphicOLAPXML) repose sur un système à la fois relationnel et XML (cf. Figure 68). Les documents XML sont chargés au sein d'un SGBD relationnel étendu pour gérer du XML : Oracle 10*g*2. Les données de ces documents (contenu, et méta-données) sont extraites et insérées dans les structures multidimensionnelles correspondantes du magasin de données inspirées d'un précédent prototype, GraphicOLAPSQL [Tournier, 2004], [Ravat et al., 2007d] et [Ravat et al., 2007e]. Enfin les analyses sont spécifiées à partir des données stockées dans les structures du magasin. Les opérations de liaison entre les différents éléments de l'architecture sont effectuées par une application java indépendante du SGBD.

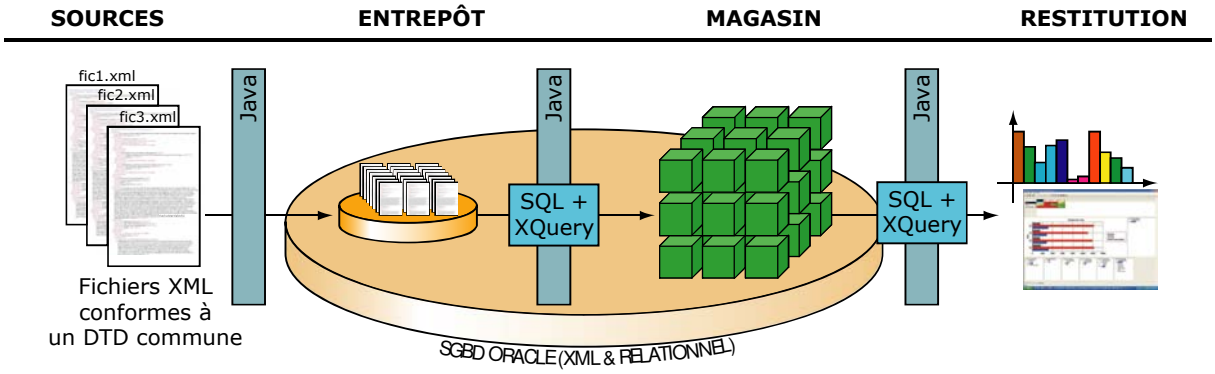


Figure 68 – Architecture générale.

L'entrepôt est l'espace de stockage centralisé qui fédère les documents XML ainsi que les données annexes soit au format XML soit dans un format relationnel. Le magasin de données est bâtit selon une architecture R-OLAP (OLAP relationnelle). Quant au système de restitution, il repose sur une application java. Cette application dispose aussi de modules qui permettent la gestion des interfaces entre les différents éléments de l'architecture.

Plan du chapitre. Ce chapitre s'articule comme suit : la section suivante présente l'architecture employée pour l'implantation du prototype. Les sections qui suivent exposent les niveaux de l'architecture implantés : la section 2 expose l'entrepôt de données ; la section 3, l'implantation du magasin et la section 4 termine sur la restitution.

2 Entrepôt de données et de documents

But. Le but de cet entrepôt de données est de présenter de manière unifiée les documents afin que leur contenu et leurs méta-données puissent être extraits pour alimenter les structures multidimensionnelles du magasin. L'entrepôt présente aussi les données complémentaires soit sous la forme de documents XML, soit sous la forme de données relationnelles.

2.1 Approche

Trois approches étaient envisageables pour stocker les documents XML source :

- le stockage des données XML directement dans le système de fichiers ;
- le stockage des données XML dans un SGBD relationnel ;
- le stockage des données XML dans un SGBD XML.

Aucune des trois approche ne satisfaisait notre problématique : le système de gestion de fichiers impose que l'application gère l'accès au support de stockage (le disque) ; l'emploi d'un SGBD relationnel permet de bénéficier d'une architecture ROLAP bien maîtrisée, mais de coûteux processus ETL doivent être employés pour convertir les données XML dans un format relationnel ; enfin, les SGBD XML bien que fournissant une aisance dans la gestion des données XML, imposent que l'intégralité des données de l'entrepôt soient au format XML. Cette solution peut s'avérer limitative dans le cadre de l'emploi de données complémentaires relationnelles.

Nous avons finalement retenu une combinaison entre la seconde approche et la troisième approche. L'implantation repose sur un SGBD relationnel étendu pour la gestion du XML (cf. Figure 69), Oracle Database 10*g*2. Cette solution combine l'avantage d'un SGBD relationnel (manipulation des données avec le langage SQL) avec celui d'un SGBD XML (manipulation des données XML avec XQuery).

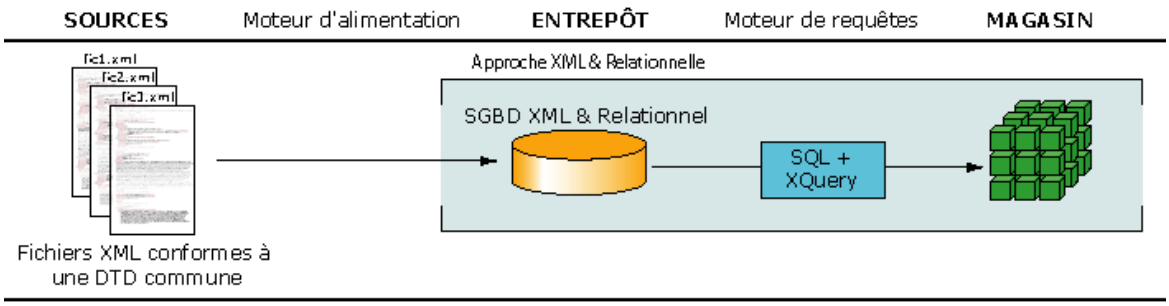


Figure 69 – Approche retenue pour implanter l'entrepôt.

2.2 Avantages de l'approche retenue

Le principal avantage de cette approche est de bénéficier d'une architecture ROLAP bien maîtrisée et déjà implantée dans notre équipe dans le prototype GraphicOLAPSQL [Tournier, 2004], [Ravat et al., 2007d] et [Ravat et al., 2007e]. Grâce à cette architecture il est possible de maintenir une certaine uniformité entre l'entrepôt et le magasin.

L'utilisation du langage XQuery associé au langage d'expression de chemins XPath permet une grande flexibilité pour la manipulation des données textuelles des documents. Elle permet aussi d'éviter les lourds processus de transformation ETL[18] nécessaires pour convertir les données XML en des tables dans le cadre d'une implantation relationnelle pure.

L'utilisation d'un SGBD adapté pour le format XML permet de bénéficier de la flexibilité de la structure XML des données ainsi que de l'emploi d'un langage de requête adapté tel que XQuery.

En outre, cette approche a permis la réutilisation d'une partie des concepts employés dans un précédent prototype de gestion de magasin de données dédié à l'analyse de données multidimensionnelle classique [Tournier, 2004].

2.3 Implantation de l'entrepôt

Source de données. Les sources de données sont des documents XML conformes à une DTD commune. Les documents sont supposés bien formés. Les données complémentaires sont soit des documents XML (éventuellement avec une autre DTD), soit des données relationnelles.

Alimentation. Les documents sont chargés un à un par une application java dans une structure simple qui facilitera l'exécution des requêtes pour en extraire le contenu et les méta-données. Les données complémentaires sont elles aussi chargées dans des structures très simples et sont liées à chaque document ou si possible à un sous-élément de chaque document.

Stockage. Les documents XML sont stockés dans une table relationnelle à raison d'un n-uplet par document. Cette table est composée d'un identifiant et d'un champ contenant l'intégralité du document sous la forme d'une chaîne de caractères comprise entre des balises XML (cf. Figure 70). Par conséquent, un ensemble de traitements XML peut être appliqué pour identifier et sélectionner les éléments qui composent les méta-données et le contenu du document. Les données complémentaires sont stockées au moyen de relations classiques si elles ne sont pas au format XML. Des index permettent les associations entre les documents et les données complémentaires. Si une donnée complémentaire est associée à seulement une sous-partie d'un document, un chemin au format XPath permet l'association.

[18] ETL : Acronyme de « Extraction, Transform, Loading » (Extraction, transformation et chargement).

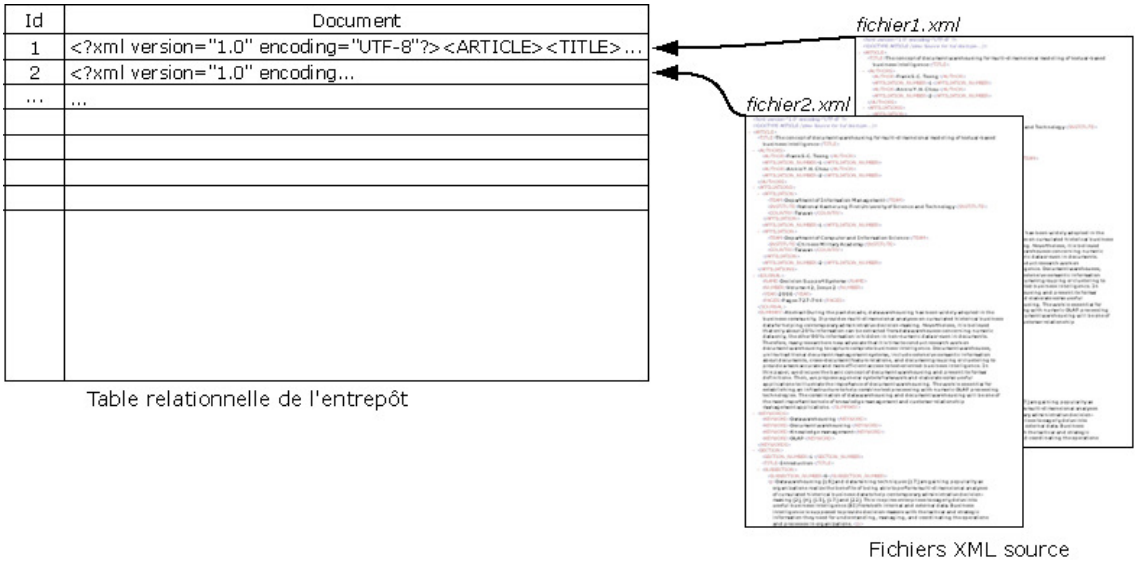


Figure 70 – Gestion des documents XML dans l'entrepôt.

Au final l'entrepôt peut être vu comme une combinaison d'un entrepôt de données relationnel et d'un entrepôt de documents XML. Toutefois, il faut noter que la priorité de cet entrepôt est d'être orienté vers des traitements d'analyse en ligne OLAP.

A partir de l'entrepôt une combinaison d'instructions adaptées du langage XQuery et SQL permet l'alimentation des structures multidimensionnelles du magasin.

3 Magasin de données

Le magasin de données est stocké dans le SGBD et repose sur une architecture R-OLAP. La Figure 71 présente l'architecture du magasin. Les données du magasin sont stockées dans des structures multidimensionnelles décrites par une méta-base qui traduit le schéma multidimensionnel en galaxie sous la forme de tables relationnelles.

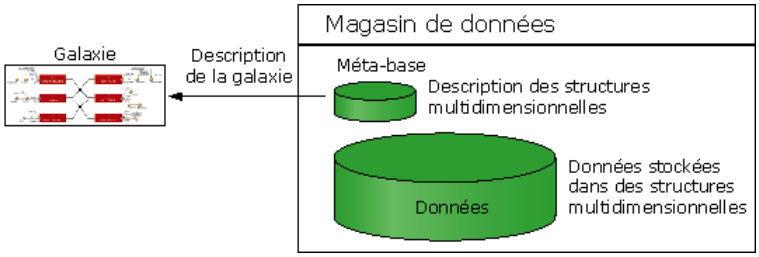


Figure 71 – Implantation du magasin.

Techniquement, les données analysées sont stockées avec les types standard du SGBD. Toutefois, il faut noter que l'interface de restitution du SGBD gérant les données textuelles est limitée à des champs de 4000 caractères. Ceci impose l'emploi de types XML spécifiques (XMLType) et de C-LOB (Character-Large OBjects) pour contourner ce problème.

3.1 Méta-base

But. Le but du magasin est de présenter les données de l'entrepôt pour en permettre l'analyse. Les données de l'entrepôt y sont structurées de manière multidimensionnelle et respectent le schéma conceptuel en galaxie qui les modélise.

Sources de données. Les structures multidimensionnelles du magasin de données sont alimentées à partir de l'entrepôt. Si nécessaire, les données XML sont converties dans d'autres formats grâce aux opérations de conversions du SGBD permettant l'interopérabilité entre structures relationnelles et types de données au format XML.

Alimentation. L'utilisateur spécifie un schéma en galaxie en accord avec les sources de données et les besoins d'analyse. Ce schéma définit les structures multidimensionnelles disponibles dans le magasin. Les données de l'entrepôt sont ensuite utilisées pour alimenter les structures.

Stockage. Le modèle en galaxie qui représente les structures multidimensionnelles du magasin est stocké sous la forme d'une méta-base qui repose sur une architecture R-OLAP. Cette implantation est inspirée de précédents travaux [Tournier, 2004].

3.1.1 Description

La méta-base permet la description des structures multidimensionnelles disponibles (cf. le schéma simplifié représenté en Figure 72). Concrètement, la galaxie est composée de cinq éléments (galaxie, dimension, attribut, hiérarchie et lien). Les attributs sont de deux types (paramètre et attribut faible). Dans la Figure 72, une classe d'association permet la description du type de liaison hiérarchique du paramètre.

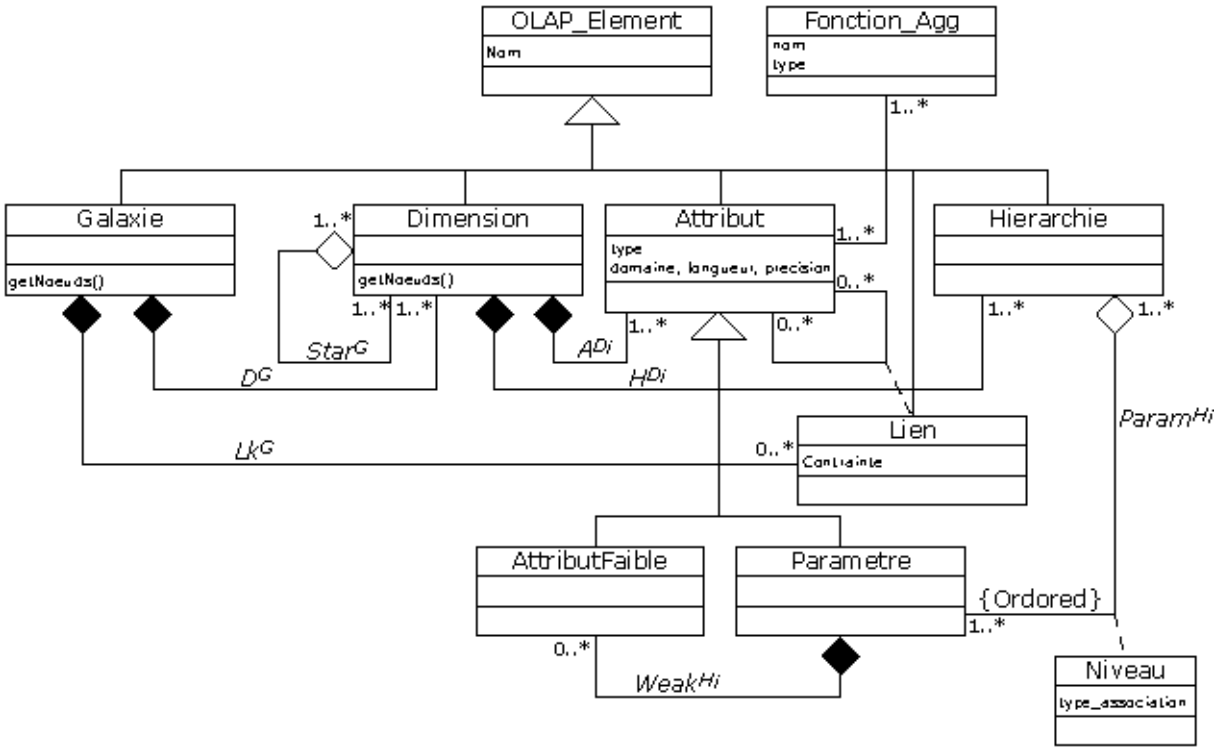


Figure 72 – Schéma conceptuel (simplifié) de la méta-base (représenté au format UML).

Les concepts représentés dans le diagramme des classes sont implantés au sein du SGBD sous la forme de relations. Dans la description qui suit, chaque relation est présentée avec les concept du diagramme des classe de la Figure 72 qu'elle traduit. Dans l'implantation relationnelle, les relations *meta_galaxy*, *meta_dimension*, *meta_attribut* et *meta_hierarchy* définissent les structures multidimensionnelles de chaque galaxie (les classes *Galaxie*, *Dimension*, *Attribut* et *Hierarchie* dans le diagramme de la Figure 72). *meta_star* représente les liaisons entre les dimensions (la liaison $Star^G$) ; *meta_parameter* permet l'identification des paramètres au sein des attributs (la classe *Parametre*) ; *meta_level* permet le positionnement des paramètres et des attributs faibles au sein des hiérarchies (la classe *Niveau* avec la notion d'ordre traduit par l'attribut *position* dans le schéma relationnel) ; *meta_link* spécifie les liens entre attributs (classe *Lien*) ; *meta_agg* permet l'association entre chaque attribut et les fonctions d'agrégation qui peuvent être employées (la liaison entre la classe *Attribut* et *Fonction_Agg*). Les fonctions sont décrites dans une table *agg_function* non représentée).

```
meta_galaxy (idg, name);
meta_dimension (idd, name, idg#);
meta_attribut (ida, name, type, domain, lgth, prec);
meta_hierarchy (idh, name, idd#);
meta_star (idd1#, idd2#);
meta_parameter (idd#, ida#, assoctyp);
meta_level (idd#, idh#, ida#, position, type);
meta_link (idl, name, ida1#, ida2#, constr);
meta_agg (ida#, idagg#);
```

3.1.2 Exemple d'instanciation

Les tables suivantes présentent un exemple de contenu de méta-schéma basé sur l'exemple d'analyse de publications scientifiques. Ces données sont issues de l'exemple d'analyse de publications scientifiques présenté en Figure 73 : la galaxie G_2, correspondant à la galaxie G_1 présentée dans les chapitres précédents et réduite à sa partie supérieure.

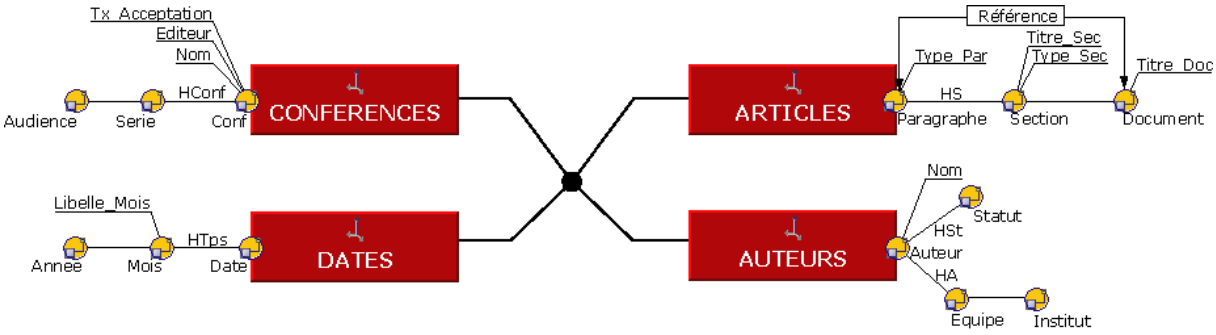


Figure 73 – Galaxie G_2.

Les tables suivantes (cf. Tableau 17) représentent les éléments structurels de G_2, à savoir, les dimensions, les hiérarchies et les attributs. Dans la table *meta_attribut*, la colonne *TYP* correspond au type de données que représente l'attribut correspondant : textuel (*Txt*) ou documentaire (*Doc*). La colonne *DOM* correspond au type physique employé pour stocker les données (*XMLType* pour du XML et *VARCHAR2* pour les chaînes de caractères).

Tableau 17 – Elements de la galaxie G_2.

META_GALAXY

IDG	Name
1	G2

META DIMENSION

IDD	Name	IDG
1	AUTEURS	1
2	ARTICLES	1
3	CONFERENCES	1
4	DATES	1

META_HIERARCHY

IDH	Name	IDD
1	HS	2
2	HA	1
3	HTps	4
4	Hconf	3

META_ATTRIBUT

IDA	Name	TYP	DOM	LON	PRE
1	Paragraphe	Doc	XMLType	NULL	NULL
2	Section	Doc	XMLType	NULL	NULL
3	Document	Doc	XMLType	NULL	NULL
4	Type_Par	Txt	VARCHAR2	10	NULL
5	Titre_Sec	Txt	VARCHAR2	30	NULL
6	Type_Sec	Txt	VARCHAR2	10	NULL
7	Titre_Doc	Txt	VARCHAR2	30	NULL
8	Auteur	Txt	VARCHAR2	10	NULL
9	Nom	Txt	VARCHAR2	20	NULL
10	Statut	Txt	VARCHAR2	5	NULL
11	Equipe	Txt	VARCHAR2	15	NULL
12	Institut	Txt	VARCHAR2	15	NULL
…					

Les tables suivantes (cf. Tableau 18) représentent les liaisons entre les différents éléments de la galaxie, avec : les liaisons entre dimensions (à gauche) ; les attributs qui sont des paramètres avec le type d'association dans la (ou les) hiérarchies dont ils dépendent (au centre) ; et la disposition des paramètres et des attributs faibles (les autres attributs) au sein des hiérarchies (à droite). La colonne « *assoctyp* » de *meta_parameter* permet de représenter le type d'association entre les différents paramètres. Les deux types : continu (C) et discontinu (D) sont décrits dans la section 3.3.

Tableau 18 – Elements de liaisons de la galaxie G_2.

META_STAR

IDD1	IDD2
1	2
1	3
1	4
2	1
2	3
2	4
…	

META_PARAMETER

IDD	IDP	ASSOCTYP
2	1	C
2	2	C
2	3	C
1	8	D
1	9	D
1	10	D
1	11	D
…		

META_LEVEL

IDD	IDH	IDP	POS	TYP
2	1	1	1	P
2	1	2	2	P
2	1	3	3	P
2	1	4	1	W
2	1	5	2	W
2	1	6	2	W
2	1	7	3	W
1	2	8	1	P
1	2	9	1	W
…				

Les tables suivantes (cf. Tableau 19) représentent les fonctions d'agrégations compatibles avec chaque attribut et la description des liens. Remarquez, que si le lien est une restriction (cf. chapitre 3), il est décrit par un prédicat stipulé dans la colonne *constr*. Ici, ce lien représente tous les *paragraphes* (attribut d'identifiant 1) dont le type (*Type_Par*) vaut la valeur « *ref* ».

Tableau 19 – associations entre fonctions d'agrégation et attributs et spécification des liens de G_2.

META_AGG

IDA	IDAGG
1	1
1	2
1	3
…	

META_LINK

IDL	Name	IDA1	IDA2	IDG	CONSTR
1	Reference	1	3	1	"Type_Par='ref'"

La relation présentant les fonctions d'agrégation n'est pas représentée. Les fonctions contenues dans la liste sont les fonctions classiques ainsi que les fonctions adaptées à l'agrégation de données textuelles. Les fonctions génériques (COUNT et LIST) ne sont pas référencées car, dans notre environnement, tout attribut est agrégeable par le biais de ces deux fonctions.

3.2 Galaxie ROLAP

Les dimensions d'une galaxie sont modélisées par des relations dénormalisées associées aux relations des autres dimensions correspondantes (cf. Figure 74). Dans le cadre d'une dimension partagée, les références vers les autres dimensions incluent aussi celles des autres nœuds auxquels participe la dimension.

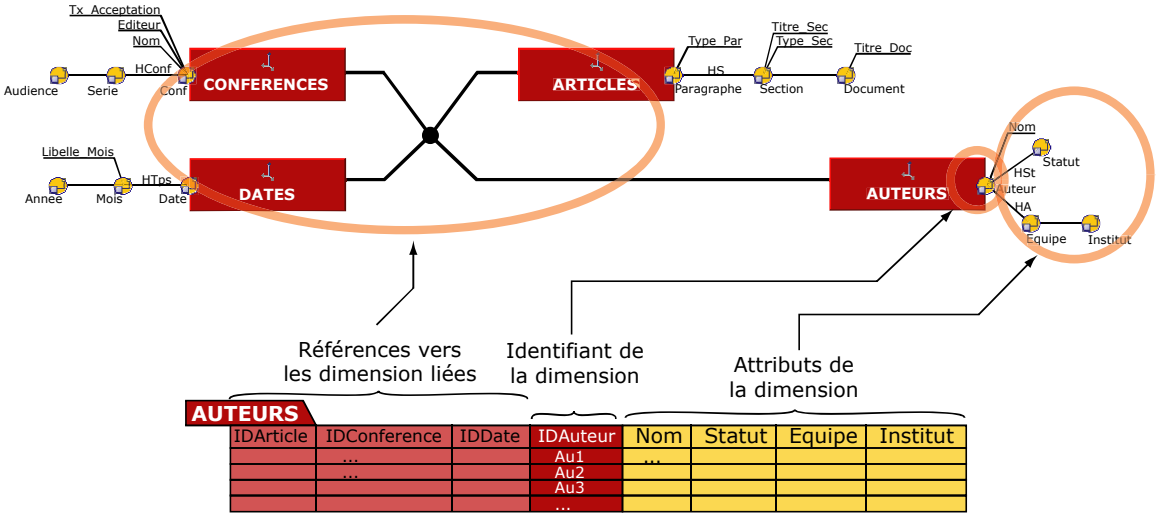


Figure 74 – Implantation logique d'une dimension.

La relation correspondant à la dimension *AUTEURS* présentée dans la Figure 74 est représentée ci-dessous :

```
auteurs (idarticle#, idconference#, iddate#, idauteur, nom, statut,
                                                  equipe, institut);
```

Cette vision dénormalisée prend de la place dans le cadre de dimensions volumineuses (nombreuses instances). Une approche normalisée (avec un schéma logique en flocon) est alors envisageable. Toutefois cela se fera au détriment du temps de calcul. Dans les cas extrêmes, à cause du volume, l'intégration des références vers les autres dimensions au sein de chaque dimension n'est plus envisageable. En conséquence, un unique index central est employé ; similaire à une table de fait dans le cas d'un schéma en étoile ou en flocons tels que ceux de [Kimball, 1996]. Il faut noter que cette solution se fait aussi au détriment du temps de calcul.

Exemple d'index central sur la galaxie G_2 :

```
index (idarticle#, idconference#, iddate#, idauteur#) ;
auteurs (idauteur, nom, statut, equipe, institut);
articles (idarticles, paragraphe,...);
conferences (idconferences, conf,...) ;
dates (iddates, date,...) ;
```

Le système dispose d'autant d'index centraux que la galaxie contient de cliques. Et chaque index est dédié à la liaison des instances correspondant uniquement à une clique.

De leur côté, les liens sont représentés par une relation composée de deux champs permettant l'association des valeurs des attributs liés. Le type de chaque champ de la relation de liaison est identique au type de chaque attribut lié (entier, varchar, XPath,…).

3.3 Implantation des données textuelles

Au niveau de l'implantation, deux types de dimensions sont distingués car ils ont un impact sur l'implantation. Au sein d'une dimension, il existe une disjonction entre les valeurs des attributs [Gyssens & Lakshmanan, 1997]. Par exemple, dans la dimension *DATES*, le domaine de l'attribut *Mois* est disjoint de l'attribut *Annee*. En effet, bien que les mois soient « contenus » dans des années, les mois sont représentés par des valeurs différentes de celles qui représentent les années. On parlera de discontinuité au sein des hiérarchies de la dimension.

$$dom(Mois) \cap dom(Année) = \varnothing$$

Toutefois une hiérarchie, qui reprend la structure de documents (par exemple *HS* dans dimension *ARTICLES*, composée de *Paragraphe*, *Section* et *Document*) la disjonction entre les domaines des attributs n'est plus valable. Comme dans le cadre d'une hiérarchie classique, il est possible de dire que les paragraphes sont « contenus » dans des sections. Mais en plus, les paragraphes étant représentés par des chaînes de caractères, les sections correspondent exactement à la concaténation des chaînes de caractères des paragraphes qu'elles contiennent. Ainsi il est possible de dire que le domaine des paragraphes est contenu (au sens propre) dans le domaine des sections. Il en est de même entre les attributs *Section* et *Document*. On parlera alors de continuité au sein des hiérarchies d'une dimension.

$$dom(Paragraphe) \subseteq dom(Section)$$

Ainsi les deux types suivants de dimensions sont considérés :

- les dimensions à *domaines discontinus* : il s'agit des dimensions dont toutes les hiérarchies sont discontinues. C'est-à-dire dont la disjonction entre les domaines des attributs tient pour l'ensemble des attributs de la dimension.
- les dimensions à *domaines continus* : une telle dimension contient au moins une hiérarchie avec une continuité entre ses attributs. C'est-à-dire que la disjonction entre les domaines des attributs n'est plus valable entre au moins deux attributs d'une même hiérarchie de la dimension.

En terme d'implantation, les dimensions à domaines discontinus sont implantées comme présenté ci-dessus. Dans le cadre de dimensions à domaines continus, l'implantation dénormalisée ou normalisée, pour les paramètres concernés, serait très volumineuse (surtout dans le cadre de données textuelles). Ainsi les données sont stockées une unique fois au format XML et des liens XPath sont employés.

La section suivante présente comme exemple de dimension à domaine continu, la dimension *ARTICLES*.

3.4 Exemple d'implantation de dimension à domaine continu

Exemple. La Figure 75 représente la décomposition d'un article. Le contenu de l'article peut être décomposé en paragraphes. Ces paragraphes sont regroupés en sections elles-mêmes regroupées en documents.

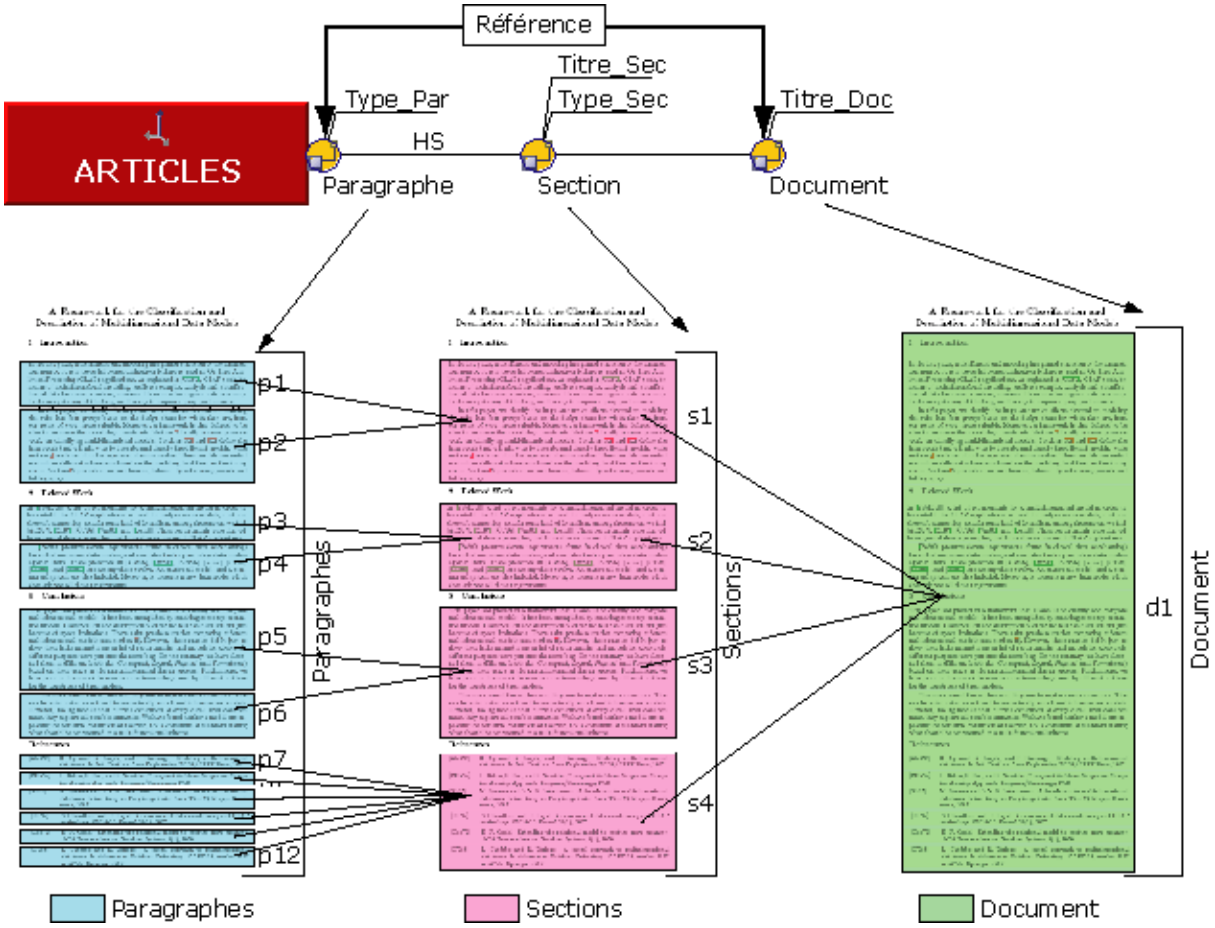


Figure 75 – Décomposition d'un article.

Tous les éléments constitutifs d'un article sont référencés par des identifiants. Les données de la dimension correspondant à cet article sont représentées dans les tableaux suivants. Pour des raisons de place et de présentation, les données ont été subdivisées en trois niveaux : paragraphe (cf. Tableau 20) ; section (cf. Tableau 21) ; et document (cf. Tableau 22). La colonne « Id » permet de faire le lien entre les trois tableaux.

Tableau 20 – Représentation des données du niveau paragraphe de la dimension *ARTICLES*.

	IdPA	Paragraphe	Type_Par
	1	/doc[@id=d1]/sec[@id=s1]/p[@id=p1]/	std
	2	/doc[@id=d1]/sec[@id=s1]/p[@id=p2]/	std
	3	/doc[@id=d1]/sec[@id=s2]/p[@id=p3]/	std
	4	/doc[@id=d1]/sec[@id=s2]/p[@id=p4]/	definition
	5	/doc[@id=d1]/sec[@id=s3]/p[@id=p5]/	theoreme
	6	/doc[@id=d1]/sec[@id=s3]/p[@id=p6]/	std
doc1	7	/doc[@id=d1]/sec[@id=s4]/p[@id=p7]/	ref
	8	/doc[@id=d1]/sec[@id=s4]/p[@id=p8]/	ref
	9	/doc[@id=d1]/sec[@id=s4]/p[@id=p9]/	ref
	10	/doc[@id=d1]/sec[@id=s4]/p[@id=p10]/	ref
	11	/doc[@id=d1]/sec[@id=s4]/p[@id=p11]/	ref
	12	/doc[@id=d1]/sec[@id=s4]/p[@id=p12]/	ref
doc2	13	/doc[@id=d2]/sec[@id=s1]/p[@id=p1]/	std
…	…		

Les liaisons des paramètres continus sont représentées par des liaisons XPath permettant le stockage de l'information textuelle de manière unique dans la base. Ainsi les données du

paramètre paragraphe sont spécifiées par un chemin XPath pointant directement sur une chaîne de caractère XML contenue dans l'entrepôt.

Par exemple, dans le Tableau 20, le premier paragraphe de l'article dont l'identifiant est d1 est représenté par le chemin XPath suivant :

/doc[*@id=d1*]/*sec*[*@id=s1*]/*p*[*@id=p1*]/

Il s'agit du paragraphe repéré par la balise <*p*> et dont l'identifiant est *p1*, ce paragraphe est contenu dans la section identifiée par la balise <*sec*> et dont l'identifiant est *s1*, cette section étant au sein du document *d1*.

Au passage, cette solution permet aussi de faire de même avec des éléments discontinus mais dont les données sont contenues dans les documents. Par exemple, les titres de sections ou encore de document (cf. la colonne *Titre_Sec* dans le Tableau 21). Bien entendu, matérialiser ces éléments est une méthode pour gagner des performances au détriment de place.

Tableau 21 - Représentation des données du niveau section de la dimension *ARTICLES*.

	IdPA	Section	Type_Sec	Titre_Sec
	1	/doc[@id=d1]/Sec[@id=s1]/	introduction	/doc[@id=d1]/Sec[@id=s1]/Titre
	2			
	3	/doc[@id=d1]/Sec[@id=s2]/	proposition	/doc[@id=d1]/Sec[@id=s2]/titre
	4			
	5	/doc[@id=d1]/Sec[@id=s3]/	proposition	/doc[@id=d1]/Sec[@id=s3]/titre
doc1	6			
	7	/doc[@id=d1]/Sec[@id=s4]/	references	/doc[@id=d1]/Sec[@id=s4]/titre
	8			
	9			
	10			
	11			
	12			
doc2	13	/doc[@id=d2]/Sec[@id=s1]/	references	/doc[@id=d2]/Sec[@id=s1]/titre
…	…			

Tableau 22 - Représentation des données du niveau document de la dimension *ARTICLES*.

	IdPA	Document	Titre_Doc
	1		
	2		
	3		
	4		
	5		
doc1	6	/doc[@id=d1]/	/doc[@id=d1]/Titre
	7		
	8		
	9		
	10		
	11		
	12		
doc2	13	/doc[@id=d2]/	/doc[@id=d2]/Titre
…	…		

4 Restitution et analyse

But. Le but de l'interface de restitution et d'analyse est dans un premier temps de permettre la représentation graphique des structures multidimensionnelles du magasin pour visualiser les données et naviguer en leur sein afin d'aider le décideur à spécifier ses analyses. Dans un second temps, elle permet la restitution des données analysées.

La représentation des structures multidimensionnelles disponibles se fait par une représentation graphique de la galaxie conforme au formalisme du modèle (cf. chapitre 3). Les schémas de galaxies présentés tout au long de la thèse sont extraits de cette interface (cf. Figure 76).

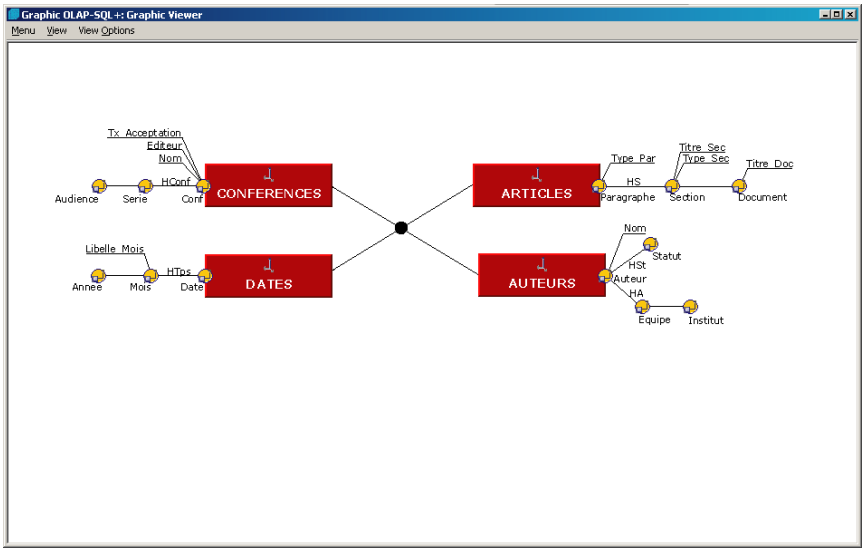


Figure 76 – Représentation graphique d'une galaxie.

La Figure 77 présente l'architecture de l'interface de restitution et d'analyse. La représentation graphique du schéma en galaxie permet au décideur d'observer les structures du magasin en faisant abstraction des contraintes d'implantation. Le décideur exprime ensuite une requête multidimensionnelle. Une fois cette requête exécutée, les résultats de la requête sont restitués à l'utilisateur.

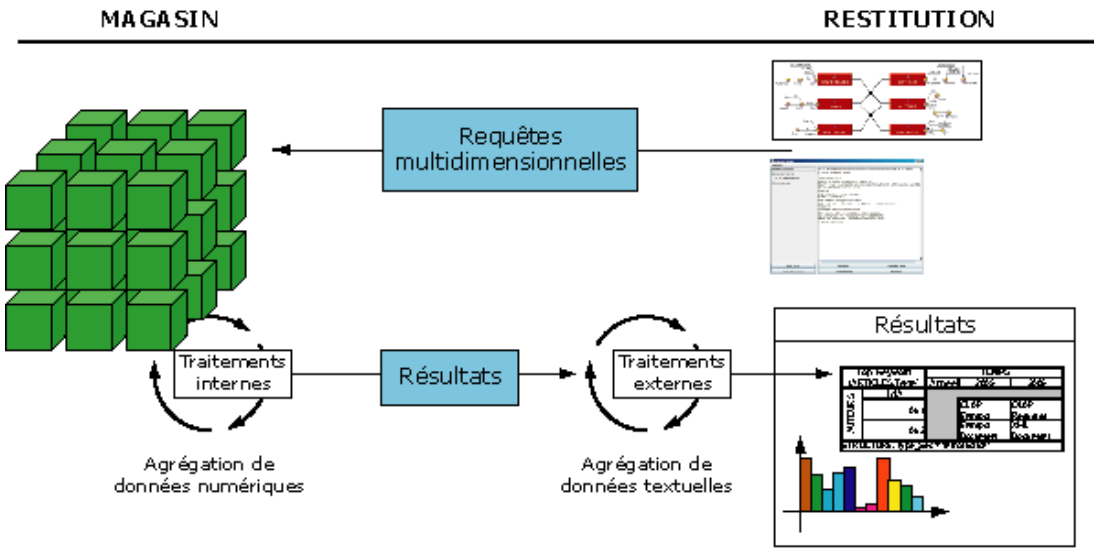


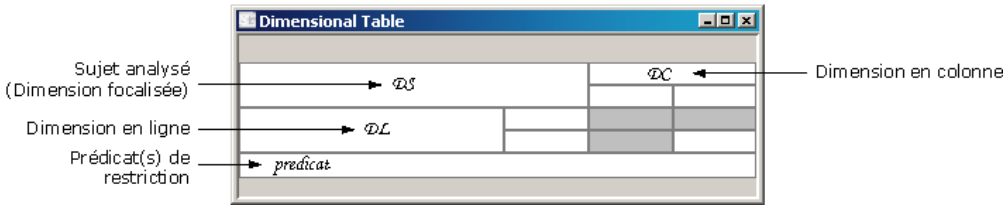
Figure 77 – Traitements au niveau du magasin.

4.1 Langage de manipulation

Inspiré du prototype GraphicOLAPSQL [Tournier, 2004], l'utilisateur spécifie ses requêtes multidimensionnelles par un langage graphique.

Pour visualiser les données analysées, une table multidimensionnelle est employée. Cette interface est adaptée à la restitution de données multidimensionnelles pour un décideur [Gyssens & Lakshmanan, 1997] et [Ravat et al., 2007e]. Il s'agit d'une table bidimensionnelle permettant la représentation de données textuelles au sein de ses cellules centrales (cf. Tableau 23).

Tableau 23 – Table multidimensionnelle.



Le langage permet la spécification d'une table multidimensionnelle par la manipulation des éléments graphiques de la représentation graphique de la galaxie (cf. Figure 78). Les cellules graphiques qui représentent les attributs (paramètres et attributs faibles) peuvent être déposées dans l'une des quatre zones de la table multidimensionnelle (cf. Tableau 23).

Exemple. Dans la Figure 78, le décideur décide d'analyser les principaux mots-clefs issus des introductions d'articles et de visualiser ces mots-clefs en fonction des auteurs et des années (Le résultat est présenté en Figure 79). La spécification de cette analyse s'effectue en trois étapes :

(1) L'utilisateur spécifie le sujet en déposant l'attribut qui l'intéresse dans la zone réservée au sujet analysé (*DS*). Il s'agit de l'attribut *Document* de la dimension *ARTICLES*. Le décideur devra ensuite préciser la fonction d'agrégation qu'il souhaite utiliser : TOP_KEYWORD.

(2) Ensuite, l'utilisateur spécifie les informations en lignes en déposant un attribut dans la zone permettant la spécification de la dimension en lignes (*DL*). Il s'agit du paramètre *Auteur* de la dimension *AUTEURS*. L'utilisateur devra ensuite préciser quelle hiérarchie il désire utiliser car ce paramètre appartient à deux hiérarchies et le système ne peut déterminer quelle hiérarchie sera employée automatiquement : *HA*.

(3) l'utilisateur répète le même processus avec la zone des colonnes (*DC*). Il dépose le paramètre *Année* de la dimension *DATES*.

(4) Enfin, l'utilisateur spécifie une restriction sur les données des articles à analyser, en limitant leur contenu à l'introduction. Ceci est fait en déposant l'attribut faible *Type_Sec* de la dimension *ARTICLES* qui décrit le type de section dans la zone de prédicats de restriction. Le système demande alors la spécification du prédicat à associer à cet attribut faible : « *Type_Sec* = *Introduction* ».

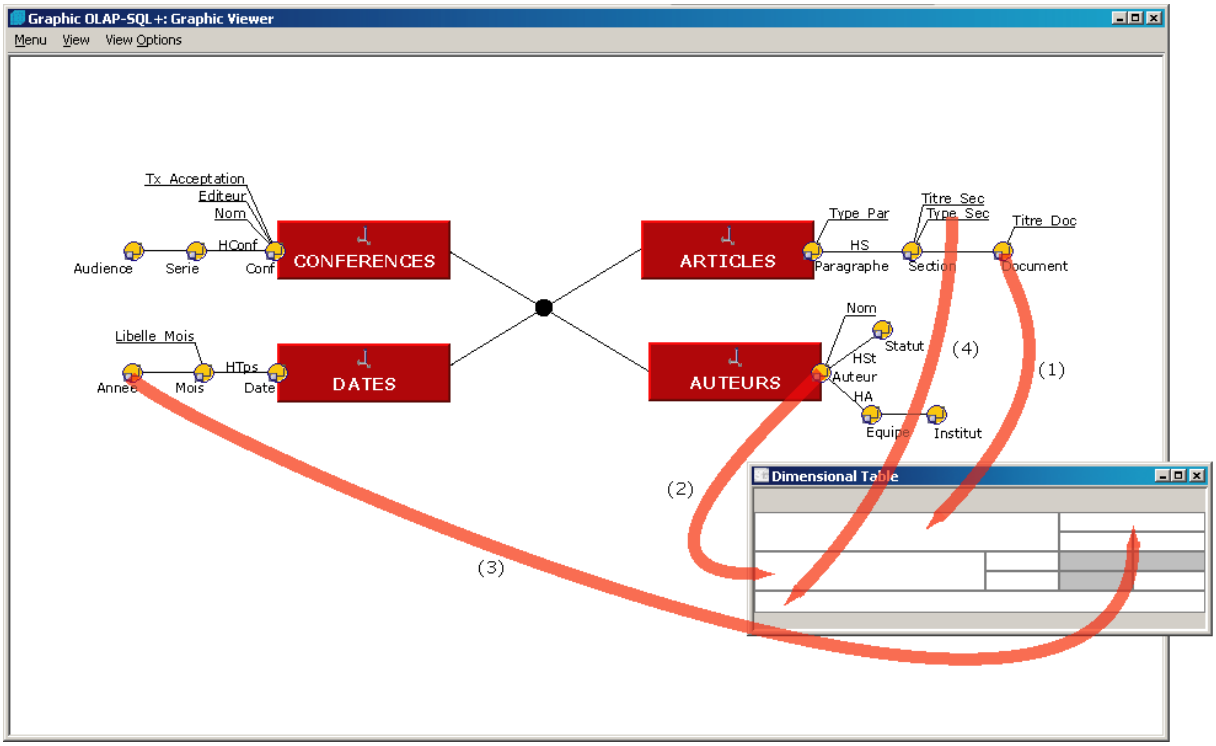


Figure 78 – Principe de la manipulation graphique permettant la spécification d'une analyse multidimensionnelle.

A partir de ce langage graphique, il est possible d'exprimer l'ensemble des analyses qui peuvent être spécifiées par les opérations de manipulation présentées dans le chapitre 4. Le langage est incrémental, c'est-à-dire qu'une fois une table dimensionnelle spécifiée, elle peut être modifiée en ajoutant de nouveaux éléments issus de la représentation multidimensionnelle conceptuelle ou bien en enlevant des éléments déjà présents dans la table.

Les opérations de forage sont spécifiées en ajoutant un attribut d'un niveau de détail plus fin pour le forage vers le bas. L'opération inverse, le forage vers le haut, est effectuée en ajoutant un attribut d'un niveau de détails moindre que l'actuel niveau le plus fin sélectionné.

Les opérations de rotation sont effectuées en ajoutant un élément provenant d'une autre dimension. Ceci a pour conséquence le remplacement des éléments sélectionnés par le nouvel élément.

A tout moment, l'utilisateur peut employer les liens présents dans la galaxie. Il précise alors à partir d'un attribut déposé dans la table multidimensionnelle, le chemin comportant le lien qu'il désire employer (cf. chapitre 4).

Exemple. Par exemple, soit un article *a*.

ARTICLES.Section représente les sections de l'article *a*.
ARTICLES.Référence.Section représente les sections des articles cités par l'article *a*, à savoir les articles cités dans la section référence de l'article *a*.
ARTICLES.Référence.AUTEURS.Nom représente les noms des auteurs des articles cités par l'article *a*.

Par exemple, la spécification de l'analyse présentée dans l'introduction du modèle en galaxie (cf. chapitre 3 et Tableau 24), l'utilisateur effectue les opérations suivantes :

déposer le paramètre *Document* de la dimension *ARTICLES* dans la zone de sujet ;

déposer le paramètre *Auteur* dans l'en-tête des colonnes, puis spécifier le chemin exact pour atteindre les auteurs : *ARTICLES.Référence.AUTEURS* ;
déposer le paramètre *Série* de la dimension *CONFERENCES* dans l'en-tête des lignes ;
déposer le paramètre *Institut* de la dimension *AUTEURS* dans la zone de prédicats de restriction et spécifier premièrement le chemin exact pour atteindre les instituts : *ARTICLES.Référence.AUTEURS* et deuxièmement spécifier le prédicat restrictif à appliquer sur le paramètre *Institut* : « = 'Inst1' ».

Tableau 24 – Exemple d'analyse de citations.

ARTICLES TOP_KEYWORDS (Document)		AUTEURS		
	Institut	Inst1		
	Auteur	Au1	Au2	Au3
CONFERENCES	Nom			
	DaWaK	XML, Docments	Fouille de données, Clustering	Fouille de données
	DEXA	XML, BD temporelles	-	-
	CAiSE	Fouille de données	XML, Entrepôts de données	Fouilles de données
Institut = 'Inst1'				

La restitution des données analysées se fait par l'intermédiaire d'une table multidimensionnelle. Toutefois, cette table a été modifiée afin de permettre une représentation plus adaptée à des données textuelles issues de documents XML.

4.2 Restitution des analyses

Pour donner plus de précision au décideur, les données textuelles présentées sont associées via des liens hypertextes aux données issues de documents d'origine (cf. Figure 80). Ces liens ne sont bien entendus établis que lorsque les données analysées s'y prêtent. Lorsque le système emploie des chemins XPath pour générer les résultats, à partir de ces chemins, il est possible de générer des liens vers les documents à l'origine des résultats au sein d'une page Web. Néanmoins, contrairement à [Tseng & Chou, 2006], seuls les fragments de documents concernés sont restitués.

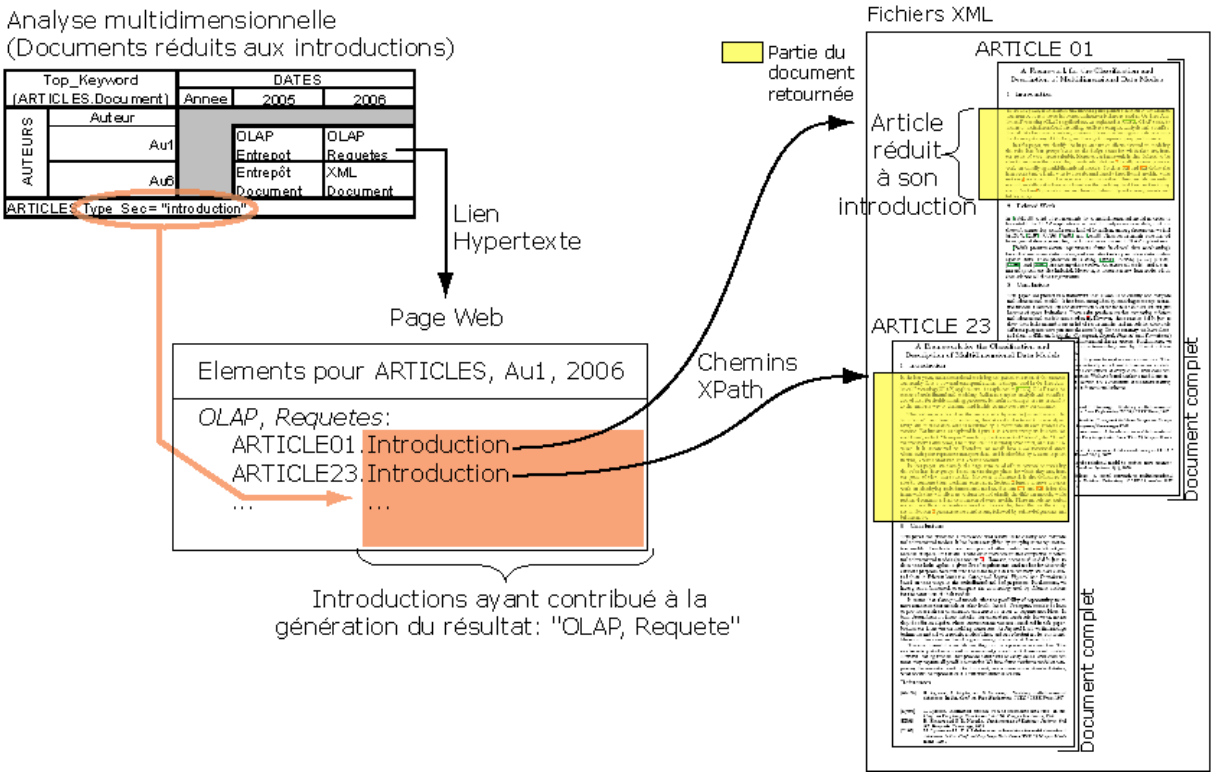


Figure 79 – Principe de consultation des sources à partir de la table multidimensionnelle.

Concrètement, le système utilise les index disponibles au sein du magasin et regroupe les chemins XPath en fonction de la requête multidimensionnelle (cf. Figure 80). Ces chemins sont alors restitués sous la forme de liens hypertextes. Ces derniers accèdent alors aux données sources des documents. Il faut noter que lorsque l'analyse ne porte pas sur des données issues de document, le système n'emploie pas ces expression de chemin au format XPath, ainsi les pages Web ne sont pas générées faute de sources documentaires à restituer.

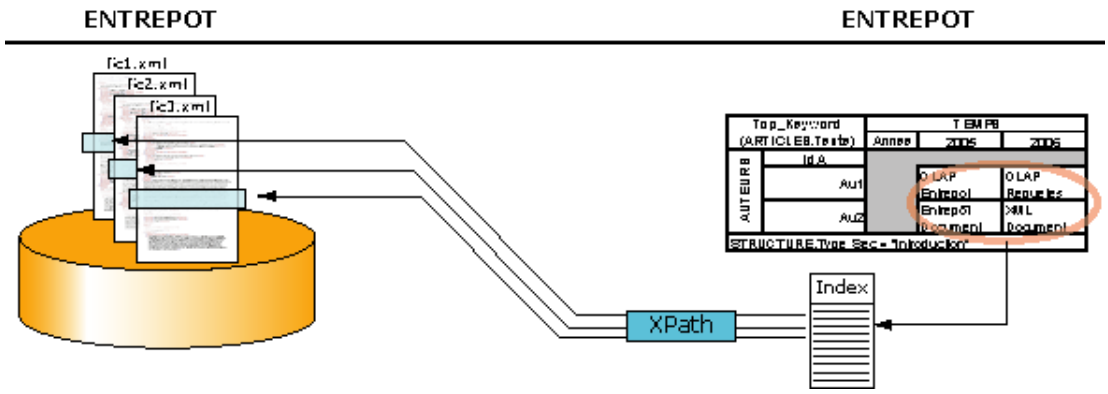


Figure 80 – Liens permettant la consultation des fragments sources correspondant aux données analysées.

Au sein de l'architecture, le type des données et des dimensions permet la gestion de leurs contraintes d'implantation. La section suivante présente ces types.

5 Validation

Le prototype, GraphicOLAPXML, présenté dans ce chapitre, implante le modèle en galaxie et les méthodes d'analyse adaptées aux données issues de documents XML. Le prototype repose sur une architecture à trois niveaux :

- un entrepôt de données et de documents ;
- un magasin de données ;
- une interface de restitution et d'analyse.

L'entrepôt nous permet de présenter des données relationnelles et des documents XML à analyser selon une vision uniforme au sein d'une structure simple. Cette structure facilite l'intégration de ces données dans les structures multidimensionnelles qui composent le magasin de données.

L'environnement multidimensionnel que constitue le magasin de données est constitué des structures définies par un schéma conceptuel en galaxie. Les données issues des documents XML stockés dans l'entrepôt sont intégrées dans des structures multidimensionnelles modélisées par la galaxie.

La galaxie modélise sous la forme de dimension les différents éléments conceptuels que le décideur peut manipuler afin de spécifier des analyses multidimensionnelles. Via ces dimensions le décideur peut analyser des données issues de documents XML principalement constitués de texte.

Les concepts modélisés par la galaxie sont manipulés par les décideurs, leur permettant la spécification d'analyses multidimensionnelles. Une interface de restitution adaptée aux données textuelles permet une analyse des données issues des documents XML de l'entrepôt.

Notre prototype permet une validation de la modélisation en galaxie pour représenter les concepts multidimensionnels disponibles pour effectuer des analyses. Il permet également de valider notre approche de manipulation pour effectuer la spécification de ces analyses. Il s'agit d'un premier pas vers l'intégration de 100% des données qui transitent au sein des systèmes d'informations.

Références

[Gyssens & Lakshmanan, 1997] Marc Gyssens, Laks V. S. Lakshmanan, "A Foundation for Multi-dimensional Databases", *23rd Intl. Conf. on Very Large Data Bases* (*VLDB*), Morgan Kaufmann, p. 106–115, 1997.

[Kimball, 1996] Ralph Kimball, *The data warehouse toolkit: Practical Techniques for Building Dimensional Data Warehouses*, John Wiley and Sons, ISBN : 0-471-15337-0, 1996, 2ème ed. : Ralph Kimball, Margaery Ross, *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling*, *2nd Edition*, John Wiley & Sons, 2002.

[Khrouf, 2004] Kaïs Khrouf, *Entrepôts de documents : De l'alimentation à l'exploitation*, Thèse de doctorat, Université Paul Sabatier, Toulouse 3 (France), juillet 2004.

[Malinowski & Zimányi, 2006] Elzbieta Malinowski, Esteban Zimányi, "Hierarchies in a multidimensional model: From conceptual modeling to logical representation", *Data & Knowledge Engineering* (*DKE*), vol.59(2), Elsevier, p. 348–377, novembre 2006.

[Ravat et al., 2007d] Franck Ravat, Olivier Teste, Ronan Tournier, Gilles Zurfluh, "Querying Multidimensional Databases", *11th East-European Conference on Advances in Databases and Information Systems* (*ADBIS 2007*), LNCS 4690, Springer, p. 298–313, 2007.

[Ravat et al., 2007e] Franck Ravat, Olivier Teste, Ronan Tournier, Gilles Zurfluh, "Algebraic and graphic languages for OLAP manipulations", *Intl. Journal of Data Warehousing and Mining* (*ijDWM*), Idea Group Publishing (IGP), juin 2007 (à paraître).

[Tournier, 2004] Ronan Tournier, *Bases de données multidimensionnelles : étude et implantation d'un langage graphique*, rapport de master de recherche, IRIT, Université Paul Sabatier, Toulouse 3 (France), juin 2004.

[Tseng & Chou, 2006] Frank S.C. Tseng, Annie Y.H. Chou, "The concept of document warehousing for multi-dimensional modeling of textual-based business intelligence", *journal of Decision Support Systems* (*DSS*), vol.42(2), Elsevier, p. 727–744, novembre 2006.

CHAPITRE VII
Conclusion et perspectives

CHAPITRE VII : Conclusion et perspectives

« Prediction is very difficult, especially about the future. »

— Niels Bohr.

1 Bilan général

Les travaux de recherche présentés dans ce mémoire de thèse s'inscrivent dans le cadre des systèmes d'aide à la prise de décision. Ces systèmes se basent sur un processus d'analyse en ligne (OLAP) et structurent les données de manière multidimensionnelle. Nous avons proposé une nouvelle approche qui consiste à intégrer les données issues de documents XML au cœur du processus d'analyse. Jusqu'à présent, ces documents sont restés exclus des systèmes d'aide à la prise de décision, faute de méthode et d'outils adaptés.

Selon [Tseng & Chou, 2006], les système OLAP n'emploient que 20% des données qui transitent au sein des systèmes d'information des entreprises. Les 80% restants, des documents restent hors de portée des systèmes d'aide à la prise de décision. Dans [Sullivan, 2001], l'auteur argumente en faveur de l'intégration des données issues de documents et de l'emploi de la fouille de texte pour en permettre l'analyse. De leur côté, les auteurs de [Fankhauser & Klement, 2003] affirment que la technologie XML est assez mature pour permettre l'implantation d'outils de fouilles de texte performant. Nous avons proposé d'aller au delà de ces propos en fournissant un environnement permettant l'intégration de documents XML au sein d'un système OLAP adapté pour l'analyse de données textuelles contenues dans ces documents. Il s'agit d'une première approche en vue de l'intégration de 100% des données issues des systèmes d'information des entreprises.

Pour permettre l'intégration de documents au sein de l'environnement d'aide à la prise de décision, nous avons proposé un **modèle** multidimensionnel associé à un ensemble d'**opérations** pour permettre la manipulation des concepts du modèle. Nous avons proposé une **démarche** pour intégrer les documents au sein de l'environnement. La **validation** de nos propositions a été effectuée par le développement d'un prototype permettant l'analyse multidimensionnelle de données issues de documents.

Modèle en galaxie. Afin de modéliser les concepts multidimensionnels que le décideur va manipuler pour spécifier ses analyses, nous avons proposé un modèle conceptuel adapté aux données issues de documents XML [Ravat et al., 2007b] et [Ravat et al., 2007f]. Ce modèle permet au décideur de faire abstraction des contraintes logiques et d'implantation. A l'instar du modèle en constellation [Ravat et al., 2007d] et [Ravat et al., 2007e], duquel il est inspiré, le modèle en galaxie dispose d'une représentation d'axes d'analyses à perspectives multiples en fournissant différents niveaux de granularité pour des analyses poussées. Le modèle permet également une représentation simple et symétrique de la notion de sujet et d'axe d'analyse par l'intermédiaire d'un unique concept de dimension. Ce modèle résume ainsi l'ensemble des éléments multidimensionnels à des axes d'analyse. Il permet de prendre en compte les caractéristiques de documents avec, la représentation de leur structure logique et la gestion des données textuelles en vue d'analyses au moyen de dimensions documentaires [Ravat et al., 2007g]. Il permet également de conserver les liens intra et inter-documents. Enfin, l'absence de sujet prédéfini fournit une plus grande flexibilité au décideur quant à l'expression d'analyses multidimensionnelles. En conclusion, le modèle en galaxie est une généralisation des modèles multidimensionnels classiques tout en permettant la prise en compte des spécificités inhérentes aux documents.

Fonction d'agrégation textuelle et langage de manipulation. La spécification d'analyses multidimensionnelles sur des données issues de documents pose le problème de l'agrégation de données textuelles. Nous avons proposé une fonction permettant d'obtenir une vision synthétique des données textuelles analysées. Cette fonction, AVG_KW [Ravat et al., 2007a] permet de résumer un ensemble de mots-clefs en un ensemble réduit et plus synthétique. Nous avons également proposé un ensemble d'opérations de manipulation [Ravat et al., 2007b] et [Ravat et al., 2007f] des concepts représentés par le modèle en galaxie. Ces opérations facilitent l'exploitation d'un schéma en galaxie par un décideur afin qu'il puisse spécifier et affiner des analyses multidimensionnelles. Ces opérations permettent : la spécification d'analyses, la réduction du volume des données analysées, le changement du niveau de détail de l'analyse et la réorientation de l'analyse. La spécification et l'exploitation des opérations sont simplifiées par l'unique concept du modèle. Cet ensemble constitue un noyau minimum fermé d'opérateurs, ce qui permet leur combinaison pour exprimer des analyses complexes.

Démarche. Nous avons proposé une démarche pour permettre l'intégration de données issues de documents XML au sein de notre environnement. Nous avons spécifié une démarche mixte qui permet la constitution d'un schéma en galaxie pour représenter les besoins en terme d'analyse de la part des décideurs tout en s'assurant de la compatibilité avec les sources de données et de documents qui seront exploitées. Les besoins d'analyse représentés par un schéma en galaxie sont confrontés aux sources de données disponibles [Ravat et al., 2007c]. En cas de conflit entre les sources et la galaxie, un processus de modification des besoins et d'enrichissement des sources permet de mettre en adéquation les deux. Les données sont alimentées ensuite au sein des structures multidimensionnelles du magasin de données.

Validation par implantation. En guise de validation, notre proposition a été implantée au sein d'un prototype. Cet outil permet la conception de schéma conceptuel en galaxie. Il permet également la réalisation de magasins reposant sur une galaxie. Enfin, il permet l'alimentation des structures multidimensionnelles du magasin à partir de données de documents XML stockées dans un entrepôt.

L'intégration des documents au sein du processus d'analyse fournit une nouvelle source de données aux systèmes d'aide à la prise de décision. Ces travaux représentent une première étape et de nombreuses perspectives peuvent être envisagées.

2 Perspectives

Le domaine de l'analyse de données issues de documents XML en est encore à ses débuts. De nombreuses perspectives sont envisageables dont voici quelques unes que nous envisageons :

Intégration de données semi-structurées. Disposer de sources XML est un avantage concernant la clarté des informations et des données. Toutefois, la structure de documents XML peut être très flexible, ainsi il faut noter qu'il est possible de rencontrer des données incomplètes (un document semi-structuré peut contenir des « trous » et ainsi avoir des éléments manquants) ou encore des structures alternatives. Les données incomplètes sont détectables par des éléments structurels optionnels dans la spécification de la structure des documents (DTD). Pour pouvoir gérer ces absences de données, il est nécessaire de créer des données artificielles comme dans la gestion de hiérarchies irrégulières [Mansmann & Scholl, 2006] et [Malinowski & Zimányi, 2006]. De leur côté, les structures alternatives sont détectables par des hiérarchisations alternatives ou optionnelles d'éléments dans la spécification des DTD. Il s'agit d'un important problème avec des données de type semi-

structurées et il est envisageable d'employer des méthodes de relaxation de schéma [Amer-Yahia et al., 2002].

Gestion des versions. Tout au long de ce mémoire de thèse les documents employés furent des articles scientifiques. Ces documents, une fois écrits sont figés dans le temps. Toutefois, ce n'est pas nécessairement le cas de tous les documents. Des documents textuels tels que des pages Web ont tendance à évoluer constamment. Des travaux au sein de notre équipe proposent de gérer les versions au sein des magasins de données [Ravat et al., 2006] et [Ravat & Teste, 2006] ou encore de gérer les versions de documents au sein des entrepôts de documents [Khrouf et al., 2007]. Il est naturel d'envisager la gestion des versions de documents au sein des magasins de données. Cela permet de maintenir une consistance au niveau des données et des structures au fur et à mesure que ces données, que le magasin permet d'analyser, évoluent. Ainsi, nous envisageons d'intégrer une gestion des versions permettant de suivre l'évolution des documents à la fois en terme de structure logique mais aussi en terme de contenu. Nous envisageons également la gestion des versions de dimensions qui permettront de faire face aux évolutions générales des besoins d'analyse des décideurs.

Problèmes de performances. Le prototype développé repose sur un SGBD XML et relationnel. Des suites du manque de maturité de la technologie XML en terme de performance, les traitements des données XML ralentissent globalement les performances du système relationnel (Oracle 10*g*2 en l'occurrence). Toutefois pour pallier à ces problèmes de performances, deux perspectives, parmi d'autres, sont envisagées :

- La matérialisation d'une partie des vues. Ces vues représenteraient le résultat de requêtes supposées probables, évitant au système de coûteux temps de calculs. Ces requêtes probables peuvent être détectées lors de la spécification de la galaxie. En effet, dans le processus d'assignation des attributs aux dimensions, les attributs en lignes de la matrice des besoins représentent des sujets d'analyse. Ainsi, avec ces informations et une évaluation du coût des autres vues à matérialiser (en comparant l'espace occupé et l'économie en terme de temps de calcul), il serait possible de sélectionner de manière optimale les vues à matérialiser.
- L'optimisation de l'espace et du traitement des données analysées. Le format XML stocke les données en tant que chaînes de caractères. Dans les documents XML principalement constitués de données textuelles, les séquences de caractères représentent des phrases constituées de mots. Le dictionnaire complet des mots utilisés dans les articles de recherche en informatique est loin d'être infini. Ainsi, il serait envisageable de remplacer les mots par des entiers qui feraient références à une entrée du dictionnaire via un index. Ceci aurait pour conséquence : 1) un gain en terme d'espace car un entier est plus court à coder informatiquement qu'une chaîne de caractères en UTF-8 ou UTF-16 (le format de caractères employé par XML) ; 2) une accélération en terme de traitement des chaînes de caractères. Notamment, ceci aurait pour conséquence une accélération du traitement de l'agrégation de mots-clef avec l'ontologie. La comparaison de données numériques étant plus rapide que la comparaison de chaînes de caractères, l'emploi de données numériques dans l'ontologie accélèrerait la recherche du LCA (le plus petit ancêtre commun). Il est à noter que les méthodes d'optimisation proposées dans [Bender & Farach-Colton, 2000] reposent elle aussi sur un traitement numérique.

Visualisation adaptée aux données textuelles. Lors d'une analyse de données textuelles, le décideur visualise les résultats au moyen d'une table multidimensionnelle adaptée pour la représentation de données issues de documents XML. Mais cette visualisation devient très vite surchargée lors de la visualisation de données textuelles volumineuse. Aussi nous

envisageons de poursuivre nos travaux sur la visualisation de données textuelles. L'emploi de cartes auto-adaptatrices (SOM – Self-Organising-Maps), aussi appelées cartes de Kohonen, pourraient être envisagées [Kohonen, 1998].

Environnement d'intégration de fonctions d'agrégation. Nous avons proposé une fonction d'agrégation adaptée aux données textuelles associée à l'emploi de fonctions d'agrégations proposées par [Park et al., 2005]. D'autres fonctions pourraient être employées et il serait envisageable de proposer un environnement d'intégration pour de nouvelles fonctions. Ainsi un concepteur pourrait ajouter des fonctions, adaptées à des besoins spécifiques d'analyse, par l'intermédiaire d'un environnement uniformisé permettant cette spécification.

Aller plus loin. L'une des perspectives les plus importante est l'inversion du principe d'un moteur de recherche sur le Web. Prenons un moteur de recherche (Exalead, Yahoo, Google…) ce dernier pose à l'utilisateur une question : « Que recherchez-vous ? Décrivez-moi ce que vous recherchez. » Il est évident de constater que « se promener » sur Internet, est impossible sans savoir ce que l'on cherche. Conclusion, au niveau de l'utilisateur ce qu'il ne connaît pas sur Internet (ou ce qu'il ne sait décrire par des mots-clefs) n'existe pas. En anglais l'expression « one browses the Web for information », est en réalité impossible. Il n'y a pas de parallèle pour le fait d'entrer au hasard dans une rue et de découvrir ses magasins et ses maisons… Ainsi, l'idée serait d'appliquer un environnement d'analyse multidimensionnel aux données indexées et issues de pages Internet, pour qu'un utilisateur puisse retourner une question au moteur de recherche : « Qu'as-tu à me proposer ? De quoi est constitué Internet aujourd'hui ? ». Ainsi, via une vision synthétique, il serait possible de parcourir l'ensemble d'Internet, d'employer les opérations de forage pour analyser les thèmes et les concepts plus en détails, de synthétiser des pages (voire des sites entiers) pour obtenir une vision orientée décideur à partir du contenu d'Internet : une vision synthétique du Web…

Références

[Amer-Yahia et al., 2002] Sihem Amer-Yahia, SungRan Cho, Divesh Srivastava, "Tree Pattern Relaxation", *Advances in Database Technology, 8th Intl. Conf. on Extending Database Technology* (*EDBT*), LNCS 2287, Springer, p. 496–513, 2002.

[Bender & Farach-Colton, 2000] Michael A. Bender, Martin Farach-Colton, "The LCA Problem Revisited", *4th Latin American Symposium on Theoretical Informatics* (*LATIN*), LNCS 1776, Springer, p. 88–94, 2000.

[Fankhauser & Klement, 2003] Peter Fankhauser, Thomas Klement, "XML for Data Warehousing Chances and Challenges" (Extended Abstract), *5th Intl. Conf. on Data Warehousing and Knowledge Discovery* (*DaWaK*), LNCS 2737, Springer, p.1-3, 2003.

[Khrouf et al., 2007] Kaïs Khrouf, Mohamed Mbarki, Franck Ravat, Chantal Soulé-Dupuy, Nathalie Vallès-Parlangeau, "Les entrepôts de documents : gestion de versions", *5ème colloque Veille Stratégique Scientifique & Technologique* (*VSST*), octobre 2007.

[Kohonen, 1998] Teuvo Kohonen, "The self-organizing map", Neurocomputing, vol.21(1-3), Elsevier, p. 1–6, 1998.

[Malinowski & Zimányi, 2006] Elzbieta Malinowski, Esteban Zimányi, "Hierarchies in a multidimensional model: From conceptual modeling to logical representation", *Data & Knowledge Engineering* (*DKE*), vol.59(2), Elsevier, p. 348–377, novembre 2006.

[Mansmann & Scholl, 2006] Svetlana Mansmann, Marc H. Scholl, "Extending Visual OLAP for Handling Irregular Dimensional Hierarchies", *8th Intl. Conf. on Data Warehousing and Knowledge Discovery* (*DaWaK*), LNCS 4081, Springer, p. 95–105, 2006.

[Park et al., 2005] Byung-Kwon Park, Hyoil Han, Il-Yeol Song, "XML-OLAP: A Multidimensional Analysis Framework for XML Warehouses", *7th Intl. Conf. on Data Warehousing and Knowledge Discovery* (*DaWaK*), LNCS 3589, Springer, p. 32–42, 2005.

[Ravat & Teste, 2006] Franck Ravat, Olivier Teste, "Supporting Data Changes in Multidimensional Data Warehouses", *Intl. Review on Computers and Software*, vol.1(3) ,Praize Worthy Prize, Wantag - USA, p. 251–259, novembre 2006.

[Ravat et al., 2006] Franck Ravat, Olivier Teste, Gilles Zurfluh, "A Multiversion-based Multidimensional Model", *8th Intl. Conf. on Data Warehousing and Knowledge Discovery* (*DAWAK*), LNCS 4081, Springer, p. 75–84, 2006.

[Ravat et al., 2007a]* **Franck Ravat, Olivier Teste, Ronan Tournier, "OLAP Aggregation Function for Textual Data Warehouse", *International Conference on Enterprise Information Systems* (*ICEIS 2007*), Funchal, Madeira - Portugal, 12-17 juin 2007, Vol. DISI, INSTICC Press, p. 151–156, juin 2007.**

[Ravat et al., 2007b]* **Franck Ravat, Olivier Teste, Ronan Tournier, Gilles Zurfluh, "Modèle conceptuel pour l'analyse multidimensionnelle de documents", *3ème journées francophones sur les Entrepôts de Données et Analyse en ligne* (*EDA*), Revue des Nouvelles Technologies de l'Information (RNTI), numéro spécial, vol.RNTI-B-3, Cépaduès Editions, p. 161–175, 2007.**

[Ravat et al., 2007c]* **Franck Ravat, Olivier Teste, Ronan Tournier, Gilles Zurfluh, "Integrating Complex Data into a Data Warehouse", *Intl. Conf. on Software Engineering and Knowledge Engineering* (*SEKE*), Knowledge Systems Institute (KSI), p. 483–486, 2007.**

[Ravat et al., 2007d]* **Franck Ravat, Olivier Teste, Ronan Tournier, Gilles Zurfluh, "Querying Multidimensional Databases", *11th East-European Conference on Advances in Databases and Information Systems* (*ADBIS*), LNCS 4690, Springer, p. 298–313, 2007.**

[Ravat et al., 2007e]* **Franck Ravat, Olivier Teste, Ronan Tournier, Gilles Zurfluh, "Algebraic and graphic languages for OLAP manipulations", *Intl. Journal of Data Warehousing and Mining* (*ijDWM*), Idea Group Publishing (IGP), juin 2007 (à paraître).**

[Ravat et al., 2007f]* **Franck Ravat, Olivier Teste, Ronan Tournier, Gilles Zurfluh, "A Conceptual Model for Multidimensional Analysis of Documents", *26th Intl. Conf. on Conceptual Modeling* (*ER*), LNCS, Springer, 2007 (à paraître).**

[Ravat et al., 2007g]* **Franck Ravat, Olivier Teste, Ronan Tournier, "Analyse multidimensionnelle de documents via des dimensions OLAP", *Document numérique, n° spécial : Entreposage de documents et données semi-structurées*, vol.9, Hermès, 2007 (à paraître).**

[Sullivan, 2001] Dan Sullivan, *Document Warehousing and Text Mining*, Wiley John & Sons, ISBN: 0471399590, 2001.

[Tseng & Chou, 2006] Frank S.C. Tseng, Annie Y.H. Chou, “The concept of document warehousing for multi-dimensional modeling of textual-based business intelligence”, *journal of Decision Support Systems* (*DSS*), vol.42(2), Elsevier, p. 727–744, novembre 2006.

---

* La liste des auteurs de ces articles est ordonnée par ordre alphabétique.

Bibliographie Générale

Bibliographie Générale

A

[Abelló et al., 2001a] Alberto Abelló, José Samos, Fèlix Saltor, "Understanding Facts in a Multidimensional Object-Oriented Model", *4th ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 32–39, 2001.

[Abelló et al., 2001b] Alberto Abelló, José Samos, Fèlix Saltor, "Understanding Analysis Dimensions in a Multidimensional Object-Oriented Model", *3rd Intl. Workshop on Design and Management of Data Warehouses* (*DMDW*), CEUR Workshop proceedings vol.39, CEUR-WS.org, p. 4.1–4.9, 2001.

[Abelló, 2002] Alberto Abelló, *YAM²: a multidimensional conceptual model*, Thèse de doctorat, Université Polytechnique de Catalogne (Espagne), avril 2002.

[Abelló et al., 2003] Alberto Abelló, José Samos, Fèlix Saltor, "Implementing operations to navigate semantic star schemas", *6th ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 56–62, 2003.

[Abelló et al., 2006] Alberto Abelló, José Samos, Fèlix Saltor, "YAM²: a multidimensional conceptual model extending UML", *Information Systems* (*IS*), vol.31(6), Elsevier, p. 541–567, septembre 2006.

[Abiteboul et al., 2002] Serge Abiteboul, Sophie Cluet, Guy Ferran, Marie-Christine Rousset, "The Xyleme project", *Computer Networks*, vol.39(3), Elsevier, p. 225–238, 2002.

[Abiteboul, 2003] Serge Abiteboul, "Managing an XML Warehouse in a P2P Context", *15th Intl. Conf. on Advanced Information Systems Engineering* (*CAiSE*), LNCS 2681, Springer, p. 4–13, 2003.

[Abiteboul, 2006] Serge Abiteboul, "Entrepôts de contenu autour de XML et des services Web", *2ème journées francophones sur les Entrepôts de Données et Analyse en ligne* (*EDA*), Revue des Nouvelles Technologies de l'Information (RNTI), numéro spécial, vol.RNTI-B-2, Cépaduès Editions, conference invite, p. 1, 2006.

[Agrawal et al., 1995] Rakesh Agrawal, Ashish Gupta, Sunita Sarawagi, *Modeling Multidimensional Databases*, IBM Research Report, http://rakesh.agrawal-family.com/papers/icde97olap_rj.pdf, 1995.

[Agrawal et al., 1997] Rakesh Agrawal, Ashish Gupta, Sunita Sarawagi, "Modeling Multidimensional Databases", *13th Intl. Conf. on Data Engineering* (*ICDE*), IEEE Computer Society, p. 232–243, 1997.

[Agrawal et al., 2000] Rakesh Agrawal, Roberto J. Bayardo Jr., Ramakrishnan Srikant, "Athena: Mining-Based Interactive Management of Text Database", *7th Intl. Conf. on Extending Database Technology* (*EDBT*), LNCS 1777, Springer, p. 365–379, 2000.

[Amer-Yahia et al., 2002] Sihem Amer-Yahia, SungRan Cho, Divesh Srivastava, "Tree Pattern Relaxation", *Advances in Database Technology, 8th Intl. Conf. on Extending Database Technology* (*EDBT*), LNCS 2287, Springer, p. 496–513, 2002.

[Annoni, 2007] Estella Annoni, *Eléments méthodologiques pour le développement des systèmes décisionnels dans un contexte de reutilisation*, Thèse de doctorat, Université Paul Sabatier Toulouse 3 (France), juillet 2007.

B

[Baziz, 2005] Mustapha Baziz, *Indexation conceptuelle guidée par ontologie pour la recherche d'information*, Thèse de doctorat, Université Paul Sabatier Toulouse 3 (France), décembre 2005.

[Bender & Farach-Colton, 2000] Michael A. Bender, Martin Farach-Colton, "The LCA Problem Revisited", *4th Latin American Symposium on Theoretical Informatics* (*LATIN*), LNCS 1776, Springer, p. 88–94, 2000.

[Beyer et al., 2005] Kevin S. Beyer, Donald D. Chamberlin, Latha S. Colby, Fatma Özcan, Hamid Pirahesh, Yu Xu, "Extending XQuery for Analytics", *ACM SIGMOD Intl. Conf. on Management of Data* (*SIGMOD*), ACM Press, p. 503–514, 2005.

[Bondy & Murty, 1976] Adrian J. Bondy, U.S.R. Murty, *Graph Theory with Applications*, Elsevier North-Holland, 1976.

[Bordawekar & Lang, 2005] Rajesh Bordawekar, Christian A. Lang, "Analytical processing of XML documents: opportunities and challenges", *ACM SIGMOD Record*, vol.34(2), ACM Press, p. 27–32, mars 2005.

[Börzsönyi et al., 2001] Stephan Börzsönyi, Donald Kossmann, Konrad Stocker, "The Skyline Operator", *17th Intl. Conf. on Data Engineering* (*ICDE*), IEEE Computer Society, p. 421–430, 2001.

[Boussaid et al., 2006] Omar Boussaid, Riadh Ben Messaoud, Rémy Choquet, Stéphane Anthoard, "X-Warehousing: An XML-Based Approach for Warehousing Complex Data", *10th East European Conf. on Advances in Databases and Information Systems* (*ADBIS*) LNCS 4152, Springer, p. 39–54, 2006.

[Bruckner et al., 2001] Robert M. Bruckner, Beate List, Josef Schiefer, A. Min Tjoa, "Modeling Temporal Consistency in Data Warehouses", *1st Intl. Workshop on Knowledge Extraction for Enterprise Services* (KEES), *12th Intl. Workshop on Database and Expert Systems Applications* (*DEXA Workshop*), IEEE Computer Society, p. 901–905, 2001.

[Bültzingsloewen, 1987] Günter von Bültzingsloewen, "Translating and Optimizing SQL Queries Having Aggregates", *13th Intl. Conf. on Very Large Data Bases* (*VLDB*), Morgan Kaufmann, p. 235–243, 1987.

C

[Cabibbo & Torlone, 1997] Luca Cabibbo, Riccardo Torlone, "Querying Multidimensional Databases", *6th Intl. Workshop Database Programming Languages* (*DBPL*), LNCS 1369, Springer, p. 319–335, 1997.

[Cabibbo & Torlone, 1998] Luca Cabibbo, Riccardo Torlone, "From a Procedural to a Visual Query Language for OLAP", *10th Intl. Conf. on Scientific and Statistical Database Management* (*SSDBM*), IEEE Computer Society, p. 74–83, 1998.

[Cabibbo & Torlone, 2000] Luca Cabibbo, Riccardo Torlone "The Design and Development of a Logical System for OLAP", *2nd Intl. Conf. on Data Warehousing and Knowledge Discovery* (*DaWaK*), LNCS 1874, Springer, p. 1–10, 2000.

[Chakrabarti et al., 1998] Soumen Chakrabarti, Byron Dom, Rakesh Agrawal, Prabhakar Raghavan, "Scalable Feature Selection, Classification and Signature Generation for

Organizing Large Text Databases into Hierarchical Topic Taxonomies", *The VLDB Journal*, vol.7(3), Springer, p. 163–178, août 1998.

[Chaudhuri & Dayal, 1997] Surajit Chaudhuri, Umeshwar Dayal, "An Overview of Data Warehousing and OLAP Technology", *ACM SIGMOD Record*, vol.26(1), ACM Press, p. 65–74, mars 1997.

[Codd, 1970] E.F. Codd, "A Relational Model of Data for Large Shared Data Banks", *Communications of the ACM*, vol.13(6), ACM Press, juin 1970.

[Codd, 1972] E. F. Codd, "Relational Completeness of Data Base Sublanguages", *Database Systems*, R. Rustin (ed.), Prentice Hall & IBM Research Report RJ 987, p. 65–98, 1972.

[Codd, 1993] E.F. Codd, S.B. Codd, C.T. Salley, *Providing OLAP (On Line Analytical Processing) to user analyst: an IT mandate*, rapport technique, E.F. Codd and associates, (white paper de Hyperion Solutions Corporation), 1993.

[Colliat, 1996] George Colliat, "OLAP, relational, and multidimensional database systems", *ACM SIGMOD Record*, vol.25(3), ACM Press, p. 64–69, septembre 1996.

D

[Datta & Thomas, 1999] Anindya Datta, Helen Thomas, "The cube data model: a conceptual model and algebra for on-line analytical processing in data warehouses", *Decision Support Systems* (*DSS*), vol.27(3), Elsevier, p. 289–301, décembre 1999.

[Dublin Core, 2007] *The Dublin Core Metadata Initiative* de http://dublincore.org/ (Dublin Core Metadata Element Set, version 1.1) en date de mai 2007.

[Dudouet et al., 2005] François-Xavier Dudouet, Ioana Manolescu, Benjamin Nguyen, Pierre Senellart, "XML Warehousing Meets Sociology", *IADIS Intl. Conf. WWW/Internet* (*ICWI*), IADIS Press, Lisbonne, Portugal, Octobre 2005.

F

[Fankhauser & Klement, 2003] Peter Fankhauser, Thomas Klement, "XML for Data Warehousing Chances and Challenges" (Extended Abstract), *5th Intl. Conf. on Data Warehousing and Knowledge Discovery* (*DaWaK*), LNCS 2737, Springer, p.1-3, 2003.

[Franconi & Kamble, 2004] Enrico Franconi, Anand Kamble, "The GMD Data Model and Algebra for Multidimensional Information", *16th Intl. Conf. on Advanced Information Systems Engineering* (*CAiSE*), LNCS 3084, Springer, p. 446–462, 2004.

[Fuhr & Großjohann, 2001] Norbert Fhur, Kai Großjohann, "XIRQL: A Query Language for Information Retrieval in XML Documents", *24th ACM Conf. on Research and development in information retrieval* (*SIGIR*), ACM Press, p.172–180, 2001.

G

[Ghozzi, 2004] Faiza Ghozzi, *Conception et manipulation de bases de données dimensionnelles à contraintes*, Thèse de doctorat, Université Paul Sabatier Toulouse 3 (France), novembre 2004.

[Golfarelli et al., 1998] Matteo Golfarelli, Dario Maio, Stefano Rizzi, "The Dimensional Fact Model: A Conceptual Model for Data Warehouses", invited paper, *Intl. Journal of*

Cooperative Information Systems (*IJCIS*), vol.7(2-3), World Scientific Publishing, p. 215–247, juin & septembre1998.

[Golfarelli et al., 2001] Matteo Golfarelli, Stefano Rizzi, Boris Vrdoljak, "Data Warehouse Design from XML Sources", *4th ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 40–47, 2001.

[Golfarelli et al., 2002] Matteo Golfarelli, Stefano Rizzi, Ettore Saltarelli, "WAND: A CASE Tool for Workload-Based Design of a Data Mart", *Decimo Convegno Nazionale su Sistemi Evoluti per Basi di Dati* (*SEBD*), p. 422–426, 2002.

[Gray et al., 1996] Jim Gray, Adam Bosworth, Andrew Layman, Hamid Pirahesh, "Data Cube: A Relational Aggregation Operator Generalizing Group-By, Cross-Tab, and Sub-Total", *12th Intl. Conf. on Data Engineering* (*ICDE*), IEEE Computer Society, p. 152–159, 1996.

[Gray et al., 1997] Jim Gray, Surajit Chaudhuri, Adam Bosworth, Andrew Layman, Don Reichart, Murali Venkatrao, Frank Pellow, Hamid Pirahesh, "Data Cube: A Relational Aggregation Operator Generalizing Group-by, Cross-Tab, and Sub Totals", *Data Mining and Knowledge Discovery* , vol.1(1), Springer, p. 29–53, mars 1997.

[Gyssens et al., 1996] Marc Gyssens, Laks V. S. Lakshmanan, Iyer N. Subramanian, "Tables as a Paradigm for Querying and Restructuring", *15th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems* (*PODS*), ACM Press, p. 93–103, 1996.

[Gyssens & Lakshmanan, 1997] Marc Gyssens, Laks V. S. Lakshmanan, "A Foundation for Multi-dimensional Databases", *23rd Intl. Conf. on Very Large Data Bases* (*VLDB*), Morgan Kaufmann, p. 106–115, 1997.

H

[Hahn et al., 2000] Karl Hahn, Carsten Sapia, Markus Blaschka, "Automatically Generating OLAP Schemata from Conceptual Graphical Models", *3rd ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 9–16, 2000.

[Harel & Tarjan, 1984] Dov Harel, Robert Endre Tarjan, "Fast Algorithms for Finding Nearest Common Ancestors", *SIAM Journal on Computing* (*SICOMP*), vol.13(2), SIAM, p. 338–355, mai 1984.

[Harinarayan et al., 1996] Venky Harinarayan, Anand Rajaraman, Jeffrey D. Ullman, "Implementing Data Cubes Efficiently", *ACM Intl. Conf. on Management of Data* (*SIGMOD*), ACM Press, p. 205–216, 1996.

[Horner et al., 2004] John Horner, Il-Yeol Song, Peter P. Chen, "An analysis of additivity in OLAP systems", *7th ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 83–91, 2004.

[Horner et al., 2004] John Horner, Il-Yeol Song, "A Taxonomy of Inaccurate Summaries and Their Management in OLAP Systems", *24th Intl. Conf. on Conceptual Modeling* (*ER*), LNCS 3716, Springer, p. 433–448, 2005.

[Huang & Su, 2002] Shi-Ming Huang, Chun-Hao Su, "The Development of an XML-Based Data Warehouse System", *3rd Intl. Conf. on Intelligent Data Engineering and Automated Learning* (*IDEAL*), LNCS 2412, Springer, p. 206–212, 2002.

[Hümmer et al., 2003] Wolfgang Hümmer, Andreas Bauer, Gunnar Harde, "XCube: XML for data warehouses", *6th ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 33–40, 2003.

I, J

[Inmon, 1996] W. H. Immon, *Building the Data Warehouse*, John Wiley and sons, New York, NY, ISBN : 0764599445, 1996 (2ème ed.), 4ème ed. 2005.

[Jagadish et al., 1999] H. V. Jagadish, Laks V. S. Lakshmanan, Divesh Srivastava, "What can Hierarchies do for Data Warehouses?", *25th Intl. Conf. on Very Large Data Bases* (*VLDB*), Morgan Kaufmann, p. 530–541, 1999.

[Jensen et al., 2001] Mikael R. Jensen, Thomas H. Møller, Torben Bach Pedersen, "Specifying OLAP Cubes On XML Data", *13th Intl. Conf. on Scientific and Statistical Database Management* (*SSDBM*), IEEE Computer Society, p. 101–112, 2001.

[Johnson & Chatziantoniou, 1999] Theodore Johnson, Damianos Chatziantoniou, "Extending Complex Ad-Hoc OLAP", *8th Intl. Conf. on Information and Knowledge Management* (*CIKM*), ACM Press, p. 170–179, 1999.

K

[Keith et al., 2005] Steven Keith, Owen Kaser, Daniel Lemire, "Analyzing Large Collections of Electronic Text Using OLAP", *APICS 29th Conf. in Mathematics, Statistics and Computer Science*, Acadia University, p. 17–26, 2005.

[Kimball, 1996] Ralph Kimball, *The data warehouse toolkit: Practical Techniques for Building Dimensional Data Warehouses*, John Wiley and Sons, ISBN : 0-471-15337-0, 1996, 2ème ed. : Ralph Kimball, Margaery Ross, *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling*, *2nd Edition*, John Wiley & Sons, 2002.

[Klug, 1982] Anthony C. Klug, "Equivalence of Relational Algebra and Relational Calculus Query Languages Having Aggregate Functions", *Journal of the ACM* (*JACM*), vol.29(3), ACM Press, p. 699–717, juillet 1982.

[Khrouf, 2004] Kaïs Khrouf, *Entrepôts de documents : De l'alimentation à l'exploitation*, Thèse de doctorat, Université Paul Sabatier Toulouse 3 (France), juillet 2004.

[Khrouf & Soulé-Dupuy, 2004] Kaïs Khrouf, Chantal Soulé-Dupuy, "A Textual Warehouse Approach: A Web Data Repository", Chapitre de *Intelligent Agents for Data Mining and Information Retrieval*, Masoud Mohammadian (Ed.), Idea Publishing Group (IGP), ISBN : 1-59140-277-8, p. 101–124, 2004.

[Khrouf et al., 2007] Kaïs Khrouf, Mohamed Mbarki, Franck Ravat, Chantal Soulé-Dupuy, Nathalie Vallès-Parlangeau, "Les entrepôts de documents : gestion de versions", *5ème colloque Veille Stratégique Scientifique & Technologique* (*VSST*), octobre 2007.

[Kohonen, 1998] Teuvo Kohonen, "The self-organizing map", Neurocomputing, vol.21(1-3), Elsevier, p. 1–6, 1998.

[Kung et al., 1975] H. T. Kung, E Luccio, E P. Preparata, "On finding the maxima of a set of vectors", *Journal of the ACM* (*JACM*), ACM Press, vol.22(4), p. 469–476, 1975.

L

[Lassila & McGuiness, 2001] Ora Lassila, Deborah L. McGuinness, "The Role of Frame-Based Representation on the Semantic Web", Knowl-edge Systems Laboratory Report KSL-01-02, Stanford University, 2001 (publié aussi dans *Computer and Information Science*, vol.6(5), Linköping University, 2001).

[Lenz & Shoshani, 1997] Hans-Joachim Lenz, Arie Shoshani, "Summarizability in OLAP and Statistical Data Bases", *9th Intl. Conf. on Scientific and Statistical Database Management* (*SSDBM*), IEEE Computer Society, p. 132–143, 1997.

[Lenz & Thalheim, 2001] Hans-Joachim Lenz, Bernhard Thalheim, "OLAP Databases and Aggregation Functions", *13th Intl. Conf. on Scientific and Statistical Database Management* (*SSDBM*), IEEE Computer Society, p. 91–100, 2001.

[Lehner, 1998] Wolfgang Lehner, "Modelling Large Scale OLAP Scenarios", *6th Intl. Conf. on Extending Database Technology - Advances in Database Technology* (*EDBT*), LNCS 1377, Springer, p. 153–167, 1998.

[Li C. & Wang, 1996] Chang Li, Xiaoyang Sean Wang, "A Data Model for Supporting On-Line Analytical Processing", *5th Intl. Conf. on Information and Knowledge Management* (*CIKM*), ACM Press, p. 81–88, 1996.

[Li Y. & An, 2005] Yu Li, Aijun An, "Representing UML Snowflake Diagram from Integrating XML Data Using XML Schema", *Intl. Workshop on Data Engineering Issues in E-Commerce* (*DEEC*), IEEE Computer Society, p. 103–111, 2005.

[Luján-Mora et al., 2002] Sergio Luján-Mora, Juan Trujillo, Il-Yeol Song, "Extending the UML for Multidimensional Modeling", *5th Intl. Conf. on The Unified Modeling Language* (*UML*), LNCS 2460, Springer, p. 290–304, 2002.

[Luján-Mora, 2005] Sergio Luján-Mora, *Data Warehouse Design with UML*, Thèse de doctorat, Université d'Alicante (Espagne), juin 2005.

[Luján-Mora et al., 2006] Sergio Luján-Mora, Juan Trujillo, Il-Yeol Song, "A UML profile for multidimensional modeling in data warehouses", *Data & Knowledge Engineering* (*DKE*), vol.59(3), Elsevier, p. 725–769, décembre 2006.

M

[Malinowski & Zimányi, 2006] Elzbieta Malinowski, Esteban Zimányi, "Hierarchies in a multidimensional model: From conceptual modeling to logical representation", *Data & Knowledge Engineering* (*DKE*), Elsevier, vol.59(2), p. 348–377, novembre 2006.

[Mansmann & Scholl, 2006] Svetlana Mansmann, Marc H. Scholl, "Extending Visual OLAP for Handling Irregular Dimensional Hierarchies", *8th Intl. Conf. on Data Warehousing and Knowledge Discovery* (*DaWaK*), LNCS 4081, Springer, p. 95–105, 2006.

[Mass & Mandelbrod, 2004] Yosi Mass, Matan Mandelbrod, "Component Ranking and Automatic Query Refinement for XML Retrieval", *3rd Intl. Workshop of the Initiative for the Evaluation of XML Retrieval, Advances in XML Information Retrieval* (*INEX*), LNCS 3493, Springer, p. 73–84, 2004.

[Mbarki, 2007] Mohamed Mbarki, *Gestion d'hétérogénéité documentaire : le cas d'un entrepôt de documents multimédias*, thèse de doctorat, Université Paul Sabatier Toulouse 3 (France), 2007 (à paraître).

[McCabe et al., 2000] Catherine McCabe, Jinho Lee, Abdur Chowdhury, David A. Grossman, Ophir Frieder, "On the design and evaluation of a multi-dimensional approach to information retrieval", *23rd Intl. ACM Conf. on Research and Development in Information Retrieval* (*SIGIR*), ACM Press, p. 363–365, 2000.

[Messaoud et al., 2004] Riadh Ben Messaoud, Omar Boussaid, Sabine Rabaséda, "A new OLAP aggregation based on the AHC technique", *7th ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 65–72, 2004.

[Messaoud, 2006] Riadh Ben Messaoud, *Couplage de l'analyse en ligne et de la fouille de données pour l'exploration, l'agrégation et l'explication des données complexes*, thèse de doctorat, Université Lumière Lyon 2 (France), novembre 2006.

[Mothe et al., 2003] Josiane Mothe, Claude Chrisment, Bernard Dousset, Joël Alau, "DocCube: Multi-dimensional visualisation and exploration of large document sets", *Journal of the American Society for Information Science and Technology* (*JASIST*), vol.54(7), Wiley Periodicals, p. 650–659, mai 2003.

N

[Nassis et al., 2004] Vicky Nassis, Rajagopal Rajugan, Tharam S. Dillon, J. Wenny Rahayu, "Conceptual Design of XML Document Warehouses", *6th Intl. Conf. on Data Warehousing and Knowledge Discovery* (*DaWaK*), LNCS 3181, Springer, p. 1–14, 2004.

[Nassis et al., 2006] Vicky Nassis, Tharam S. Dillon, Rajugan Rajagopalapillai, Wenny Rahayu, "An XML Document Warehouse Model", 12*th Intl. Conf. on Database Systems for Advanced Applications* (*DASFAA*), LNCS 3882, Springer, p. 513–529, 2006.

[Nguyen et al., 2000] Thanh Binh Nguyen, A. Min Tjoa, Roland Wagner, "An Object Oriented Multidimensional Data Model for OLAP", *1st Intl. Conf. on Web-Age Information Management* (*WAIM*), LNCS 1846, Springer, p. 69-82, 2000.

[Nguyen et al., 2001] Thanh Binh Nguyen, A. Min Tjoa, Oscar Mangisengi, "Meta Cube-X: An XML Metadata Foundation for Interoperability Search among Web Data Warehouses", *3rd Intl. Workshop on Design and Management of Data Warehouses* (*DMDW*), CEUR Workshop Proceedings vol.39, CEUR-WS.org, p. 8.1–8.8, 2001.

[Nguyen et al., 2003] Thanh Binh Nguyen, A. Min Tjoa, Oscar Mangisengi, "MetaCube XTM: A Multidimensional Metadata Approach for Semantic Web Warehousing Systems", *5th Intl. Conf. Data Warehousing and Knowledge Discovery* (*DaWaK*), LNCS 27370 Springer, p. 76–88, 2003.

[Niemi Ta. et al., 2002] Tapio Niemi, Marko Niinimäki, Jyrki Nummenmaa, Peter Thanisch, "Constructing an OLAP cube from distributed XML data", *5th ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 22–27, 2002.

[Niemi Ti. et al., 2003] Timo Niemi, Lasse Hirvonen, Kalervo Järvelin: Multidimensional Data Model and Query Language for Informetrics. *Journal of the American Society for Information Science and Technology* (*JASIST*), vol.54(10), Wiley Periodicals, p. 939–951, mai 2003.

O

[Oracle Spatial, 2006] *Oracle Spatial, User's Guide and Reference 10g Release 2* (*10.2*), Oracle, B14255-03, mars 2006.

[Özsoyoglu et al., 1985] Gültekin Özsoyoglu, Zehra Meral Özsoyoglu, Francisco Mata, "A Language and a Physical Organization Technique for Summary Tables", *ACM Intl. Conf. on Management of Data* (*SIGMOD*), *ACM SIGMOD Record*, vol.14(4), ACM Press, p. 3–16, décembre 1985.

[Özsoyoglu et al., 1987] Gültekin Özsoyoglu, Zehra Meral Özsoyoglu, Victor Matos, "Extending Relational Algebra and Relational Calculus with Set-Valued Attributes and Aggregate Functions", *ACM Transactions on Database Systems* (*TODS*), vol.12(4), ACM Press, p. 566–592, 1987.

P

[Parent et al., 1999] Christine Parent, Stefano Spaccapietra, Esteban Zimányi, "Spatio-Temporal Conceptual Models: Data Structures + Space + Time", *7th Intl. Symposium on Advances in Geographic Information Systems* (*ACM-GIS*), ACM Press, p. 26–33, 1999.

[Park et al., 2005] Byung-Kwon Park, Hyoil Han, Il-Yeol Song, "XML-OLAP: A Multidimensional Analysis Framework for XML Warehouses", *7th Intl. Conf. on Data Warehousing and Knowledge Discovery* (*DaWaK*), LNCS 3589, Springer, p. 32–42, 2005.

[Pedersen D. et al., 2002] Dennis Pedersen, Karsten Riis, Torben Bach Pedersen, "Query optimization for OLAP-XML federations", *5th ACM Intl. workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 57–64, 2002.

[Pedersen D. et al., 2004] Dennis Pedersen, Jesper Pedersen, Torben Bach Pedersen, "Integrating XML Data in the TARGITOLAP System", industrial paper, *20th Intl. Conf. on Data Engineering* (*ICDE*), IEEE Computer Society, p. 778–781, 2004.

[Pedersen T.B., 2000] Torben Bach Pedersen, *Aspects of Data Modeling and Query Processing for Complex Multidimensional Data*, Thèse de doctorat, Université d'Aalborg (Danemark), 2000.

[Pedersen T.B. et al., 2001] Torben Bach Pedersen, Christian S. Jensen, Curtis E. Dyreson, "A foundation for capturing and querying complex multidimensional data", *Information Systems* (*IS*), vol.26(5), Elsevier, p. 383–423, juillet 2001.

[Pérez et al., 2005] Juan Manuel Pérez, Rafael Berlanga Llavori, María José Aramburu, Torben Bach Pedersen, "A relevance-extended multi-dimensional model for a data warehouse contextualized with documents", *8th ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 19–28, 2005.

[Pérez et al., 2006] Juan Manuel Pérez-Martínez, Rafael Belanga, María José Aramburu, Torben Bach Pedersen, *Integrating Data Warehouses with Web Data : A Survey*, DB technical report, TR-18, http://www.cs.aau.dk/DBTR, 2006.

[Pérez et al., 2007] Juan Manuel Pérez-Martínez, Rafael Belanga-Llavori, María José Aramburu-Cabo, Torben Bach Pedersen, "Contextualizing data warehouses with documents", *Decision Support Systems* (*DSS*), Elsevier, (In Press) disponible en ligne depuis février 2007.

[Pokorný, 2001] Jaroslav Pokorný, "Modelling Stars Using XML", *4th ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 24–31, 2001.

[Pourrabas & Rafanelli, 2000] Elaheh Pourabbas, Maurizio Rafanelli, "Hierarchies and Relative Operators in the OLAP Environment", *ACM SIGMOD Record*, vol.29(1), ACM Press, p. 32–37, 2000.

[Pourrabas & Rafanelli, 2003] Elaheh Pourabbas, Maurizio Rafanelli, "Hierarchies", Chapitre IV, *Multidimensional Databases: Problems and Solutions*, Maurizio Rafanelli (Ed.), Idea Publishing Group (IGP), ISBN 1-59140-053-8, p. 91–115, 2003.

R

[Rafanelli, 2003] Maurizio Rafanelli, "Operators for Multidimensional Aggregate Data", Chapitre V, *Multidimensional Databases: Problems and Solutions*, Maurizio Rafanelli (Ed.), Idea Publishing Group (IGP), ISBN 1-59140-053-8, p. 116–165, 2003.

[Rajugan et al., 2003] Rajagopal Rajugan, Elizabeth Chang, Tharam S. Dillon, Ling Feng, "XML Views: Part 1", *14th Intl. Conf. on Database and Expert Systems Applications* (*DEXA*), LNCS 2736, Springer, p. 148–159, 2003.

[Ravat et al., 2006] Franck Ravat, Olivier Teste, Gilles Zurfluh, "Algèbre OLAP et langage graphique", *Actes du XXIVème Congrès INFormatique des ORganisations et Systèmes d'Information et de Décision* (*INFORSID*), Inforsid (Ed.), ISBN 2-906855-22-7, p. 1039–1054, 2006.

[Ravat et al., 2006b] Franck Ravat, Olivier Teste, Gilles Zurfluh, "A Multiversion-based Multidimensional Model", *8th Intl. Conf. on Data Warehousing and Knowledge Discovery* (*DAWAK*), LNCS 4081, Springer, p. 75–84, 2006.

[Ravat & Teste, 2006] Franck Ravat, Olivier Teste, "Supporting Data Changes in Multidimensional Data Warehouses", *Intl. Review on Computers and Software*, vol.1(3) ,Praize Worthy Prize, Wantag - USA, p. 251–259, novembre 2006.

[Ravat et al., 2007a]* **Franck Ravat, Olivier Teste, Ronan Tournier, "OLAP Aggregation Function for Textual Data Warehouse", *International Conference on Enterprise Information Systems* (*ICEIS 2007*), Funchal, Madeira - Portugal, 12-17 juin 2007, Vol. DISI, INSTICC Press, p. 151–156, juin 2007.**

[Ravat et al., 2007b]* **Franck Ravat, Olivier Teste, Ronan Tournier, Gilles Zurfluh, "Modèle conceptuel pour l'analyse multidimensionnelle de documents", *3ème journées francophones sur les Entrepôts de Données et Analyse en ligne* (*EDA*), Revue des Nouvelles Technologies de l'Information (RNTI), numéro spécial, vol.RNTI-B-3, Cépaduès Editions, p. 161–175, 2007.**

[Ravat et al., 2007c]* **Franck Ravat, Olivier Teste, Ronan Tournier, Gilles Zurfluh, "Integrating Complex Data into a Data Warehouse", *Intl. Conf. on Software Engineering and Knowledge Engineering* (*SEKE*), Knowledge Systems Institute (KSI), p. 483–486, 2007.**

[Ravat et al., 2007d]* **Franck Ravat, Olivier Teste, Ronan Tournier, Gilles Zurfluh, "Querying Multidimensional Databases", *11th East-European Conference on Advances in Databases and Information Systems* (*ADBIS*), LNCS 4690, Springer, p. 298–313, 2007.**

[Ravat et al., 2007e]* **Franck Ravat, Olivier Teste, Ronan Tournier, Gilles Zurfluh, "Algebraic and graphic languages for OLAP manipulations", *Intl. Journal of Data Warehousing and Mining* (*ijDWM*), Idea Group Publishing (IGP), juin 2007 (à paraître).**

[Ravat et al., 2007f]* **Franck Ravat, Olivier Teste, Ronan Tournier, Gilles Zurfluh, "A Conceptual Model for Multidimensional Analysis of Documents", *26th Intl. Conf. on Conceptual Modeling* (*ER*), LNCS, Springer, 2007 (à paraître).**

[Ravat et al., 2007g]* **Franck Ravat, Olivier Teste, Ronan Tournier, "Analyse multidimensionnelle de documents via des dimensions OLAP", *Document numérique, n° spécial : Entreposage de documents et données semi-structurées*, vol.9, Hermès, 2007 (à paraître).**

[Ravat, 2007] Franck Ravat, *Outils pour la conception et la manipulation de systèmes d'aide à la décision*, habilitation à diriger les recherches (HDR), Université de Toulouse 1 (France), à paraître, 2007.

[Rivest et al., 2005] Sonia Rivest, Yvan Bédard, Marie-Josée Proulx, Martin Nadeau, Frederic Hubert, Julien Pastor, "SOLAP technology: Merging business intelligence with geospatial technology for interactive spatio-temporal exploration and analysis of data", *ISPRS Journal of Photogrammetry and Remote Sensing* (*JPRS*), Elsevier, vol.60(1), p. 17–33, December 2005.

[Rizzi et al., 2006] Stefano Rizzi, Alberto Abelló, Jens Lechtenbörger, Juan Trujillo, "Research in data warehouse modeling and design: dead or alive?", *9th ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 3–10, 2006.

[Rusu et al., 2004] Laura Irina Rusu, J. Wenny Rahayu, David Taniar, "On Building XML Data Warehouses", *5th Intl. Conf. on Intelligent Data Engineering and Automated Learning* (*IDEAL*), LNCS 3177, Springer, p. 293–299, 2004.

S

[Sapia et al., 1998] Carsten Sapia, Markus Blaschka, Gabriele Höfling, Barbara Dinter, "Extending the E/R Model for the Multidimensional Paradigm", *Advances in Database Technologies, ER '98 Workshops on Data Warehousing and Data Mining, Mobile Data Access, and Collaborative Work Support and Spatio-Temporal Data Management* (*ER Workshops*), LNCS 1552, Springer, p. 105–116, 1998.

[Schneider, 2003] Michel Schneider, "Well-formed data warehouse structures", *5th Intl. Workshop on Design and Management of Data Warehouses* (*DMDW*), CEUR Workshop Proceedings vol.77, CEUR-WS.org, p. 2.1–2.13, 2003.

[Schneider, 2007] Michel Schneider, "A general model for the design of data warehouses", *Intl. Journal of Production Economics*, Elsevier, disponible en ligne 2007 (à paraître).

[Shoshani, 2003] Arie Shoshani, "Multidimensionality in Statistical, OLAP, and Scientific Databases. Multidimensional Databases", Chapitre II, *Multidimensional Databases: Problems and Solutions*, Maurizio Rafanelli (Ed.), Idea Publishing Group (IGP), ISBN 1-59140-053-8, p. 46–68, 2003.

[Stefanovic et al., 2000] Nebojsa Stefanovic, Jiawei Han, Krzysztof Koperski, "Object-Based Selective Materialization for Efficient Implementation of Spatial Data Cubes", *IEEE Transactions on Knowledge and Data Engineering* (*TKDE*), IEEE Computer Society, vol.12(6), p. 938–958, novembre 2000.

[Sullivan, 2001] Dan Sullivan, *Document Warehousing and Text Mining*, Wiley John & Sons, ISBN: 0471399590, 2001.

T

[Teste, 2000] Olivier Teste, *Modélisation et Manipulation d'Entrepôts de Données Complexes et Historisées*, thèse de doctorat, Université Paul Sabatier Toulouse 3 (France), décembre 2000.

[Torlone, 2003] Riccardo Torlone, "Conceptual Multidimensional Models", Chapitre III, *Multidimensional Databases: Problems and Solutions*, Maurizio Rafanelli (Ed.), Idea Publishing Group (IGP), ISBN 1-59140-053-8, p. 69–90, 2003.

[Tournier, 2004] Ronan Tournier, *Bases de données multidimensionnelles : étude et implantation d'un langage graphique*, rapport de master de recherche, IRIT, Université Paul Sabatier Toulouse 3 (France), juin 2004.

[Tourwé et al., 2003] Tom Tourwé, Luk Stoops, Stijn Decneut, "Automated support for data exchange via XML", *5th Intl. Symposium on Multimedia Software Engineering* (*ISMSE*), IEEE Computer Society, p. 70–77, 2003.

[Trujillo et al., 2003] Juan Trujillo, Sergio Luján-Mora, Il-Yeol Song, "Applying UML For Designing Multidimensional Databases And OLAP Applications", *Advanced Topics in Database Research*, Keng Siau (Ed.), vol.2, Idea Group Publishing (IGP), p. 13–36, 2003.

[Tryfona et al., 1999] Nectaria Tryfona, Frank Busborg, Jens G. Borch Christiansen, "starER: A Conceptual Model for Data Warehouse Design", *2nd ACM Intl. Workshop on Data Warehousing and OLAP* (DOLAP), ACM Press, p. 3–8, 1999.

[Tseng & Chou, 2006] Frank S.C. Tseng, Annie Y.H. Chou, "The concept of document warehousing for multi-dimensional modeling of textual-based business intelligence", *journal of Decision Support Systems* (*DSS*), vol.42(2), Elsevier, p. 727–744, novembre 2006.

V, W

[Vrdoljak et al., 2003] Boris Vrdoljak, Marko Banek, Stefano Rizzi, "Designing Web Warehouses from XML Schemas", *5th Intl. Conf. on Data Warehousing and Knowledge Discovery* (*DaWaK*), LNCS 2737, Springer, p. 89–98, 2003.

[Vrdoljak et al., 2006] Boris Vrdoljak, Marko Banek, Zoran Skocir, "Integrating XML Sources into a Data Warehouse", *2nd Intl. Workshop on Data Engineering Issues in E-Commerce and Services* (*DEECS*), LNCS 4055, Springer, p. 133–142, 2006.

[W3C-XML, 2006] *Extensible Markup Language (XML) 1.0 (Fourth Edition)*, recommandation du W3C (29/09/2006), http://www.w3.org/TR/xml/

[W3C-XSL, 1999] *XSL Transformations (XSLT) version 1.0*, recommandation du W3C (19/09/1999) http://www.w3.org/TR/xslt

[W3C-XSchema, 2006] *XML Schema 1.1 Part 1 : Structures*, document de travail du W3C (31/08/2006) http://www.w3.org/TR/xmlschema11-1/ et *XML Schema 1.1 Part 2: Datatypes*, document de travail du W3C (17/02/2006) http://www.w3.org/TR/xmlschema11-2/

[W3C-XQuery, 2007] *XQuery 1.0 and XPath 2.0 Formal Semantics*, recommandation du W3C (23/01/2007) http://www.w3.org/TR/xquery-semantics/

[Wang et al., 2003] Hongzhi Wang, Jianzhong Li, Zhenying He, Hong Gao, "Xaggregation: Flexible Aggregation of XML Data", *4th Intl. Conf. on Advances in Web-Age Information Management* (*WAIM*), LNCS 2762, Springer, p. 104–115, 2003.

[Wang et al., 2005] Hongzhi Wang, Jianzhong Li, Zhenying He, Hong Gao, "OLAP for XML Data", *5th Intl. Conf. on Computer and Information Technology* (*CIT*), IEEE Computer Society, p. 233–237, 2005.

[Wiwatwattana et al., 2007] Nuwee Wiwatwattana, H. V. Jagadish, Laks V. S. Lakshmanan, Divesh Srivastava "X^3: A Cube Operator for XML OLAP", *23rd Intl. Conf. on Data Engineering* (*ICDE*), IEEE Computer Society, p. 916–925, 2007.

Y, Z

[Yin & Pedersen T.B., 2004] Xuepeng Yin, Torben Bach Pedersen, "Evaluating XML-extended OLAP queries based on a physical algebra", *7th ACM Intl. Workshop on Data Warehousing and OLAP* (*DOLAP*), ACM Press, p. 73–82, 2004.

[Zhang et al., 2003] Ji Zhang, Tok Wang Ling, Robert M. Bruckner, A. Min Tjoa, "Building XML Data Warehouse Based on Frequent Patterns in User Queries", *5th Intl. Conf. on Data Warehousing and Knowledge Discovery* (*DaWaK*), LNCS 2737, Springer, p. 99–108, 2003.

* La liste des auteurs de ces articles est ordonnée par ordre alphabétique.

UNIVERSITE TOULOUSE III – PAUL SABATIER
U.F.R. MATHEMATIQUES INFORMATIQUE GESTION (MIG)

THÈSE

en vue de l'obtention du

DOCTORAT DE L'UNIVERSITE DE TOULOUSE
délivré par l'Université Toulouse III – Paul Sabatier

Discipline : Informatique

présentée et soutenue
par

Ronan Tournier

le 13 décembre 2007

Titre :

Analyse en ligne (OLAP) de documents

Jury :

Omar Boussaid	Maître de conférence habilité, université Lyon 2	Rapporteur
Patrick Marcel	Maître de conférence, université de Tours	Examinateur
Franck Ravat	Maître de conférence, université Toulouse 1	Examinateur
Michel Schneider	Professeur, université Clermont 2	Rapporteur
Chantal Soulé-Dupuy	Professeur, université Toulouse 1	Examinatrice
Olivier Teste	Maître de conférence, université Toulouse 3	Examinateur
Gilles Zurfluh	Professeur, université Toulouse 1	Directeur

IRIT, INSTITUT DE RECHERCHE EN INFORMATIQUE DE TOULOUSE (UMR 5505),
Université Paul Sabatier Toulouse 3, 118 route de Narbonne, F-31062 Toulouse Cedex 9, France.

Résumé (court)

Les entrepôts de données et les systèmes d'analyse en ligne OLAP (On-Line Analytical Processing) fournissent des méthodes et des outils permettant l'analyse de données issues des systèmes d'information des entreprises. Mais, seules 20% des données d'un système d'information est constitué de données analysables par les systèmes OLAP actuels. Les 80% restant, constitués de documents, restent hors de portée de ces systèmes faute d'outils ou de méthodes adaptés. Pour répondre à cette problématique nous proposons un modèle conceptuel multidimensionnel pour représenter les concepts d'analyse. Ce modèle repose sur un unique concept, modélisant à la fois les sujets et les axes d'une analyse. Nous y associons une fonction pour agréger des données textuelles afin d'obtenir une vision synthétique des informations issues de documents. Cette fonction résume un ensemble de mots-clefs par un ensemble plus petit et plus général. Nous introduisons un noyau d'opérations élémentaires permettant la spécification d'analyses multidimensionnelles à partir des concepts du modèle ainsi que leur manipulation pour affiner une analyse. Nous proposons également une démarche pour l'intégration des données issues de documents, qui décrit les phases pour concevoir le schéma conceptuel multidimensionnel, l'analyse des sources de données ainsi que le processus d'alimentation. Enfin, pour valider notre proposition, nous présentons un prototype.

Mots-clefs

OLAP, Document XML, Galaxie, Modélisation multidimensionnelle, Démarche de modélisation, Base de données multidimensionnelle, Entrepôts de données, Entrepôts de documents.

Summary

Data warehouses and OLAP systems (On-Line Analytical Processing) provide methods and tools for enterprise information system data analysis. But only 20% of the data of a corporate information system may be processed with actual OLAP systems. The rest, namely 80%, i.e. documents, remains out of reach of OLAP systems due to the lack of adapted tools and processes. To solve this issue we propose a multidimensional conceptual model for representing analysis concepts. The model rests on a unique concept that models both analysis subjects as well as analysis axes. We define an aggregation function to aggregate textual data in order to obtain a summarised vision of the information extracted from documents. This function summarises a set of keywords into a smaller and more general set. We introduce a core of manipulation operators that allow the specification of analyses and their manipulation with the use of the concepts of the model. We associate a design process for the integration of data extracted from documents within an OLAP system that describes the phases for designing the conceptual schema, for analysing the document sources and for the loading process. In order to validate these propositions we have implemented a prototype.

Keywords

OLAP, XML document, Galaxy, Multidimensional modelling, modelling process, Multidimensional database, Data warehouse, Document warehouse.

Résumé

Les entrepôts de données et les systèmes d'analyse en ligne OLAP (On-Line Analytical Processing) fournissent des méthodes et des outils puissants permettant l'analyse de données issues des systèmes d'information des entreprises. Mais, seules 20% des données d'un système d'information d'entreprise est constitué de données analysables par les systèmes OLAP actuels. Les 80% restant sont constitués de documents (rapports, notes, articles…). Ces documents restent hors de portée des systèmes OLAP faute d'outils ou de méthodes adaptés. Dans le cadre des systèmes d'aide à la décision, l'omission des données contenues dans ces documents peut mener à des analyses imprécises voire erronées engendrant un risque d'erreur pour la prise de décision. Les documents représentent une capitalisation de connaissances, au même titre que les données analysables du système d'information représentent une capitalisation d'évènements (ventes, achats…). Il est donc naturel de prévoir l'ajout dans l'analyse en ligne des documents. De nos jours, un décideur maîtrise très bien les processus OLAP. Ainsi se pose la question : comment lui fournir un environnement permettant l'analyse en ligne de 100% des données disponibles avec des méthodes et des moyens qu'il maîtrise ?

Pour répondre à cette problématique nous proposons un modèle conceptuel multidimensionnel. Contrairement aux modèles multidimensionnels classiques qui s'appuient sur la dualité de concepts fait / dimension, notre modèle ne repose que sur un unique concept permettant de modéliser à la fois les sujets et les axes d'une analyse. Le modèle fournit au décideur une vision des éléments multidimensionnels disponibles pour exprimer les analyses.

Les analyses multidimensionnelles reposent sur une capacité à résumer les informations en les agrégeant avec des fonctions d'agrégation. Toutefois, il n'existe pas de moyen dans un environnement OLAP pour agréger des données textuelles qui représentent le cœur des documents analysés. Ainsi nous proposons une fonction capable d'agréger des données textuelles pour permettre d'obtenir une vision synthétique des informations. Cette fonction d'agrégation cherche à résumer un ensemble de mots-clefs par un ensemble plus petit et plus général.

Afin de pouvoir spécifier des analyses sur les données issues de documents, nous introduisons des opérations permettant la manipulation des concepts du modèle. Dans un premier temps, ces opérations permettent la spécification d'une analyse multidimensionnelle à partir des éléments représentés par le modèle. Dans un second temps, nous définissons un noyau d'opérations élémentaires permettant la modification d'une analyse afin que le décideur puisse affiner ses observations et prendre la meilleure décision possible.

Nous proposons une démarche pour l'intégration des données issues de documents dans un système OLAP, car elle diffère des processus classiques. Cette méthode décrit les phases nécessaires pour concevoir le schéma conceptuel multidimensionnel à partir des besoins d'analyse, l'analyse des sources de données qui serviront à alimenter le système ainsi que le processus final d'alimentation.

Enfin, pour valider notre proposition, nous présentons un prototype écrit en Java. Des structures multidimensionnelles, implantées au sein d'un SGBD, représentent les concepts manipulés par le décideur. Les analyses sont spécifiées par l'intermédiaire d'une interface et les données de ces analyses sont synthétisées et restituées à l'utilisateur.