OLAP AGGREGATION FUNCTION FOR TEXTUAL DATA WAREHOUSE

Franck Ravat(2), Olivier Teste(1), Ronan Tournier(1)
IRIT, team SIG-ED: Systèmes d'Informations Généralisés – Entrepôts de Données,
(1)IRIT,Université Toulouse 3, 118 rte de Narbonne *(2)IRIT,Université Toulouse 1, 2 rue du doyen G. Marty,*
F-31062 Toulouse Cedex 9, FRANCE *F-31042 Toulouse Cedex 9, FRANCE*
ravat@irit.fr, teste@irit.fr, tournier@irit.fr

Keywords: OLAP, Data Warehouse, Aggregation Function, Document Warehouse, Non-Additive Measure.

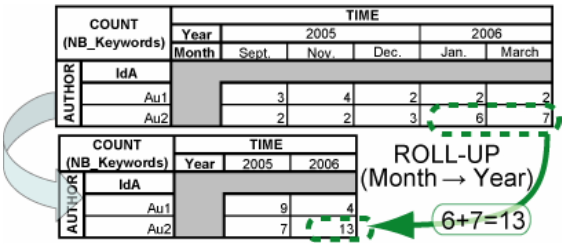
Abstract: For more than a decade, OLAP and multidimensional analysis have generated methodologies, tools and resource management systems for the analysis of numeric data. With the growing availability of semi-structured data there is a need for incorporating text-rich document data in a data warehouse and providing adapted multidimensional analysis. This paper presents a new aggregation function for keywords allowing the aggregation of textual data in OLAP environments as traditional arithmetic functions would do on numeric data. The AVG_KW function uses an ontology to join keywords into a more common keyword.

1 INTRODUCTION

OLAP (On-Line Analytical Processing) systems allow analysts to improve decision-making process by analysing aggregated historical business data. These analyses are based on a centralized data repository, called a data warehouse (Kimball, 1996). Within data warehouses, the use of Multidimensional DataBases (MDB) enables decision-makers to gain insight into an enterprise performance.

1.1 Context and Motivations

Multidimensional OLAP analysis displays analysis subject data according to various levels of detail (data granularity). The process aggregates the data according to the level of detail with functions such as sum, average, maximum, minimum… Drilling operations are the most common OLAP operations. They consist in allowing the analyst to change the displayed data granularity, thus the analysed data is aggregated according to a new granularity level. In Figure 1, a decision-maker analyses the *number of keywords monthly* used by *authors*. In order to get a more global view on the data, he changes the display by *years* (he "rolls-up"). As a consequence, the monthly values are aggregated into a value for each year.



COUNT (NB_Keywords)		TIME				
	Year	2005			2006	
	Month	Sept.	Nov.	Dec.	Jan.	March
AUTHOR	IdA					
	Au1	3	4	2	2	2
	Au2	2	2	3	6	7

COUNT (NB_Keywords)		TIME	
	Year	2005	2006
AUTHOR	IdA		
	Au1	9	4
	Au2	7	13

Figure 1: Multidimensional analysis of keyword counts displayed by authors and by months and rolled-up to years.

According to (Tseng and Chou, 2006) 20% of corporate information system data is transactional, i.e. numeric. This may easily be processed because multidimensional analysis is robust and it is a mastered technique on numeric-centric data warehouses (Sullivan, 2001). The remaining 80%, namely traditional "paperwork," stays out of reach of OLAP processes due to the lack of tools and resource management for non-numeric textual data such as text-rich documents. OLAP provides powerful tools and methods but within a rigid framework. Unstructured documents do not fit in this framework. Recently, XML technology has provided a wide framework for sharing, spreading and working with documents within corporate networks or over the web. Thus, storing documents and semi-structured data was integrated within data warehouses and repositories.

Document warehousing slowly emerged as solutions were created (Sullivan, 2001), e.g. Xyleme[1].

We argue that, to provide more exhaustive multidimensional analyses, OLAP decision support systems should provide the use of a 100% of corporate information system data. But, up to now, the OLAP framework lack the ability to cope with the analysis of semi-structured text-rich document data. As a consequence, there is a need for adapted conceptual models and textual aggregation processing.

1.2 Related Works

Related works may be divided according to two major categories. Firstly is the integration of XML data with 1) physical integration of XML data into a data warehouse. (Pokorný, 2001) builds a star schema on a logical XML structure; (Niemi et al., 2002) assembles "on the fly" XML data cubes from user queries; (Zhang et al., 2003) deals with building data warehouses on top of XML data and (Vrdoljak et al., 2003) creates a data warehouse multidimensional schema from XML schemas; and 2) the association of XML data with a data warehouse (logical integration). In (Yin and Pedersen, 2004), the authors federate XML data and traditional multidimensional data into an OLAP system. Although all these works consider textual data through the use of XML documents, they are all based on numeric-centric analysis and lack support for text-rich document-centric data analysis.

The second category concerns multidimensional analysis of documents within an OLAP framework. In (Pérez et al., 2005) the authors combine traditional numeric analysis and information retrieval techniques to assist multidimensional analysis by providing relevant documents to the ongoing analysis context. In (McCabe et al., 2000) and (Mothe et al., 2003), the authors propose the use of traditional OLAP framework to count documents according to keywords or topics in order to query more precisely a document collection. Similarly in (Chakrabarti et al., 1998) and (Agrawal et al., 2000), the authors offer tools and methods to efficiently build a hierarchical classification of documents based on typical keywords. In (Tseng and Chou, 2006) and (Keith et al., 2005), the authors suggest to build a specific keyword dimension to allow multidimensional analysis of documents. Nowadays, industrial solutions start to appear such as Text OLAP[2]. In (Khrouf et al., 2004) the authors describe a document warehouse where documents are grouped by similar structures; multidimensional analysis may be performed but still with the use of numeric analysis.

These advanced propositions show the following limitations: 1) textual data is difficult to analyse as systems use numeric measures to get round the analysis of non-numeric data; 2) the most advanced systems are limited to counting keywords in document sets; and 3) non numeric indicators may not be processed. Finally, in (Park et al., 2005), the authors introduce the concept of multidimensional document analysis within an XML framework. Unfortunately, all aggregation functions using text mining techniques are not detailed.

1.3 Aims and Contributions

The next step of decision making is to leap ahead of numeric indicators and to allow the powerful OLAP framework to operate on non-numeric data. Contrarily to previously stated works, we wish to focus the analysis on text. Our approach has the advantage of combining qualitative analysis with quantitative analysis, e.g. the analysis of the keywords of a specific publication, in order to provide an overview of publication contents. To allow multidimensional OLAP analysis of documents, we provide an aggregation function for textual OLAP analysis. This function is based on a conceptual model that provides: 1) adapted concepts to support non-numeric textual measures; and 2) a new concept to drive OLAP textual aggregation processing with the use of a domain ontology.

The rest of this paper is organised as follows: section 2 defines the conceptual model and section 3 describes the aggregation function AVG_KW.

2 CONCEPTUAL MODEL

In this section we define an extension a traditional multidimensional model to handle textual data analysis. We provide the addition of specific textual measures as well as a hierarchical representation of the analysed concepts with the use of an ontology.

2.1 Multidimensional Model

Multidimensional models have been used for over a decade. See (Torlone, 2003) for recent survey. Most use facts and dimensions to model multidimensional structures.

Dimensions model analysis axes and are composed of a set of parameters which are organised into one or more hierarchies. Each *hierarchy* represents

[1] Xyleme server from http://www.xyleme.com

[2] http://www.megaputer.com/products/pa/

an analysis perspective along the axis. The *parameters* represent different levels according to which analysis data may be observed.

The subject of analysis, namely a *fact*, is a conceptual grouping of measures which are numeric indicators. These *measures* are traditionally numeric and may be additive, semi-additive or non-additive (Kimball, 1996), (Horner et al., 2004). Here, analysis of textual data requires textual measures that fall into non-numeric and non-additive categories.

Definition 1. A *textual measure* is a measure that holds textual data, i.e. non-numeric and non-additive data.

A textual measure represents words, strings, paragraphs or even whole documents. Within these measures, we define the following categories:

Definition 2. A *raw textual measure* is a textual measure that corresponds to the full text of a document or to a fragment of that document.

Definition 3. A *keyword measure* is an elaborated textual measure, where each measure instance x_i is represented by $x_i = (kw_i, d_i)$ such that kw_i is a keyword and d_i a distance.

Raw textual measures are provided for flexibility, allowing the user to consult document contents.

Keyword measures require a certain amount of pre-processing in order to be created. The domain of all keywords is *dom*(*kw*). Notice that $x_i \in X$ with $X = dom(kw) \times \mathbb{N}$ and all distances d_i=0, this value will be used during the aggregation process.

For example, to get a view of the subjects of a collection of scientific articles, a decision-maker analyses keywords used by authors. The fact *Articles* has a numeric measure: *Acceptance*, corresponding to the acceptance rate of each article; and two textual measures: the raw textual measure representing the complete article (*Text*) and the elaborated textual measure (*Keywords*) which holds keywords extracted from article bodies. The resulting multidimensional schema is displayed in Figure 2. Graphic notations are inspired by (Golfarelli et al., 1998).

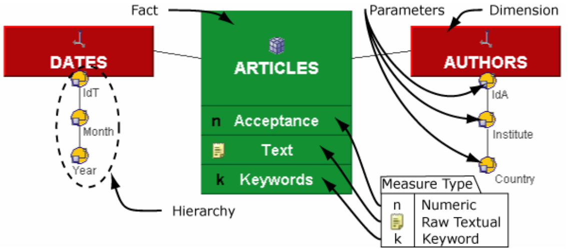


Figure 2: Example of a multidimensional conceptual schema for textual analysis.

2.2 Ontology and Operations

In order to allow analysis of textual measures, we use a hierarchical representation of domain concepts. These concepts are modelled through a “light” or “informal is-a” ontology (Lassila and McGuinness, 2001). It corresponds to a hierarchy of domain concepts where each node represent a concept (a keyword) and each link between nodes models a more complex relation than an “is-a” relation.

Definition 4. Given an ontology *O*, the *domain* of *O*, noted *dom*(*O*), represents all the keywords of *O*.

For example in Figure 3, $OLAP \in dom(O\_IS)$.

Definition 5. We call *depth* of an ontology the maximum number of nodes between the root node and lowest nodes, i.e. the leaves.

In our example the *depth*(*O_IS*) = 8.

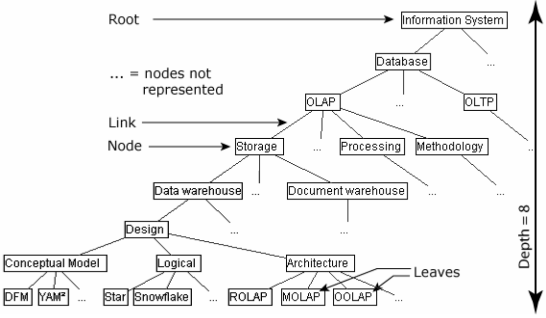


Figure 3: Example of a simple domain ontology on information systems named *O_IS*.

To allow the model to operate with the ontology, we provide two operations that take two nodes—keywords—as input: n_1 and n_2.

Definition 6. The *Least Common Ancestor*:

$$lca : (dom(O))^2 \rightarrow dom(O)$$
$$(n_1, n_2) \mapsto n_{LCA}$$

is a function returning the least common ancestor (n_{LCA}) within *O* between n_1 and n_2.

Definition 7. The *Distance* between two nodes:

$$d : (dom(O))^2 \rightarrow N$$
$$(n_1, n_2) \mapsto \max(d(n_1, lca(n_1, n_2)), d(n_2, lca(n_1, n_2)))$$

is a function that returns the number of nodes between the least common ancestor (LCA) and the lowest node.

In *O_IS*, lca(*ROLAP, Document Warehouse*)= *Storage*. The distance between these two keywords is 4: *d(ROLAP, Document Warehouse) = max (d(ROLAP, Storage), d(Document Warehouse, Storage)) = max (4, 1) = 4.*

3 AGGREGATION FUNCTION

Multidimensional OLAP analysis on non-additive measures is very limited because actual systems provide only two aggregation functions: *COUNT* and *LIST* (Kimball, 1996). We redefine the *LIST* function in order to operate on a keyword measure.

Definition 8. *LIST* aggregation function:

$$LIST : X^n \rightarrow (dom(O))^n \quad where\ X = dom(kw) \times \mathbb{N}$$
$$(x_1,\ldots,x_n) \mapsto (kw_1,\ldots,kw_n)$$

generates the list of keywords without performing any aggregation and removes the keyword distance.

In this section we define the aggregation function for domain keyword measures.

3.1 Keyword Aggregation Function

The aggregation function *AVG_KW* is designed to aggregate sets of keywords. Given a set of keywords as input, the function generates a new set of aggregated keywords. The aggregation process uses the domain ontology defined in the conceptual model (ontology and document sources are supposed to be from the same domain). For each pair of keywords, the function finds the corresponding least common ancestor (LCA). But, when aggregating very distant keywords, no matter how deep the ontology is, there is a high probability of systematically returning the root keyword of the ontology. To avoid this, a limit within the aggregation process must be specified. Indeed, the further keywords are from one another, the more sense is lost during aggregation process. In order to overcome this problem, the function uses a maximum authorized distance when aggregating keywords: D_{MAX}. So far heuristics suggest a distance of 3 or 4 nodes and a domain ontology as deep as possible. So far, the ontology research field has not solved this problem.

To display results, we use a bi-dimensional table displaying a fact and two dimensions (Gyssens and Lakshmanan, 1997), (Ravat et al., 2006). For each combination of analysis axis values, the table contains a cell. *AVG_KW* takes as input the content of these cells (sets of keywords) and produces a new set as output. The new set is composed of aggregated keywords and/or keywords from the original cell if aggregation failed due to excessive distances between the keywords.

Definition 9. We define the aggregation function:

$$AVG\_KW: X^n \rightarrow X^m \qquad X= dom(kw) \times \mathbb{N}$$
$$(x_1,\ldots,x_n) \mapsto (y_1,\ldots,y_m),\ m \le n$$

Input: $(x_1,\ldots,x_n) \in X^n$ is an ordered set of keywords such that $\forall x_i \in X,\ x_j \in X \mid i<j,\ d(x_i, x_{ROOT}) \le d(x_j, x_{ROOT})$ (i.e. the furthest nodes from the root are first) and $x_i=(kw_i,d_i)$ with $kw_i \in dom(O)$ and $d_i \le D_{MAX}$.
Output: $(y_1,\ldots,y_m) \in X^m$ is a set of aggregated keywords.

Output is generated using the following function:

Definition 10.

$$(x_i,x_j) \mapsto \begin{cases} x_{LCA} = (kw_{LCA}, l(x_i,x_j)) \text{ if } l(x_1,x_2) \le D_{MAX} \\ (x_i,x_j) \text{otherwise} \end{cases}$$

where:

$$l(x_i,x_j) = d(kw_i,kw_j) + d_i + d_j$$
$$kw_{LCA} = LCA(kw_i,kw_j)$$

If x_i and x_j are aggregated into x_{LCA} then x_i and x_j are removed from the input set X and x_{LCA} is added to X. The aggregation process is iterated on X until no more aggregation may be performed:

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

Notice that for a given y_k of X, if $d_k=0$, then the corresponding keyword kw_k was not aggregated during the process and $\exists\ x_i \in X \mid x_i=y_k$. Notice also that if $\forall\ x_i,\ x_j \in X$, $l(x_i,\ x_j)>D_{MAX}$, then there is no aggregation possible and $(y_1,\ldots,y_m) = (x_1,\ldots,x_n)$ with $m=n$.

3.2 Algorithm

The algorithm takes as input a list of keywords to be aggregated: `KW_LIST`={kw$_1$, kw$_2$,…, kw$_n$} and an ontology `O`. It produces as output an aggregated keyword list: `Output_List`. `d(keyword`$_1$`, keyword`$_2$`)` is function that computes the distance between both keywords. `Order_List` is a function that orders a list of keywords such that $d(kw_i, kw_{ROOT}) \le d(kw_j, kw_{ROOT})$. That is, keywords are ordered by the level they may be found in `O`, starting by the lowest levels, i.e. the keywords furthest from the root. `LCA` is a function finding the least common ancestor of a pair of nodes in a tree. See (Harel and Tarjan, 1984) and more recently (Bender and Farach-Colton, 2000) for discussion and implementation of the LCA problem.

```
{KW_List = OrderList(KW_List,O);
 For each KW$_i$ of KW_LIST Do
   l$_i$ = 0;
   For each KW$_j$ of KW_List, (j>i) Do
      KW$_{LCA}$=LCA(KW$_i$,KW$_j$) ;
      l$_{LCA}$=MAX(d(KW$_i$,KW$_{LCA}$),d(KW$_j$,KW$_{LCA}$))+l$_i$
      If ( l$_{LCA}$ $\le$ D$_{MAX}$ ) Then
         KW_List=KW_List-{KW$_i$, KW$_j$};
         KW$_i$=KW$_{LCA}$; l$_i$=l$_{LCA}$;
      end_If;
   end_For;
```

```
  Add KW_i to Output_List;
end_For;}
```

3.3 AVG_KW Example

The use of drilling operations makes intensive use of aggregations. Thus, for this example, we shall use the Roll-Up operation presented in introduction and the conceptual schema displayed in Figure 2. Table 1 presents a sample dataset of three documents. Two keywords have been extracted from each document.

Table 1: A sample dataset of three documents.

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

In Figure 4, the positions of the different keywords of the previous table are pointed out by rectangles in the ontology. Arrows show possible aggregation process (with distances between nodes specified). Here, D_{MAX}=3.

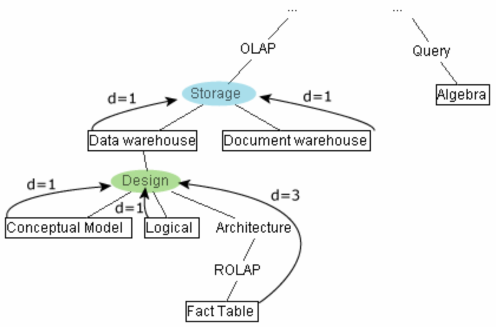


Figure 4: The position of the different keywords in the domain ontology *O_IS* (only partly represented).

In Figure 5-(a), the decision-maker analyses the publications of author *Au_1* during *2004* displaying results by *months*. The keywords of the two publications in September are aggregated: *Fact Table*, *Conceptual Model* and *Logical* are aggregated into *Design* with a distance of 3. *Data Warehouse* is too far away from the lowest keyword of the set that generated *Design*, thus it is not aggregated: *d*(*Fact Table*, *Data Warehouse*) = 4 > D_{MAX}.
In the document from November, the keyword *Algebra* is also too far from *Document Warehouse*, thus they are not aggregated either.

In Figure 5-(b), to get a more general view, the analyst "rolls-up" the analysis to a more general level of detail. Instead of observing results by *months* he will analyse them by *year*. Thus the system aggregates the two sets of keywords of the table (a) into a unique set in table (b). The keyword *Data Warehouse* has a distance of 1 with *Document Warehouse* and thus will be aggregated into *Storage* but *Design* has already a distance of 3 (D_{MAX}), thus, as *Algebra* these keywords are too far and are not aggregated. The resulting cell in the mTable is: AVG_KW((*Data Warehouse*, 0), (*Design*, 3), (*Document Warehouse*, 0), (*Algebra*, 0)) = ((*Storage*, 1), (*Design*, 3), (*Algebra*, 0)).

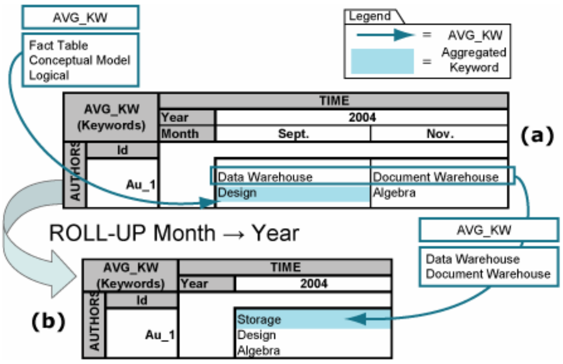


Figure 5: Analysis of keywords by months (a) and Roll-Up operation from TIME.Month to TIME.Year (b).

4 CONCLUSION

Up to now, OLAP systems are based on quantitative analysis with the use of numeric measures. As a first step towards multidimensional OLAP textual analysis, we presented in this paper a framework for the use of textual measures. In order to focus the analysis on textual data, textual measures were added to traditional multidimensional modelling. These measures allow the specification of elaborated textual measures such as keyword. We provide an aggregation function used during operations of the analysis process (such as drilling operations). This aggregation function aggregates keywords into more general ones with the use of a light domain ontology.

We are currently implementing our approach on top of an existing OLAP analysis tool: Graphic OlapSQL. This tool is based on a ROLAP data warehouse held in an Oracle *10g* RDBMS. The tool is a Java 5 client composed of a hundred classes.

We intend to continue our researches on several fields. The use of a light ontology (hierarchy of concepts) as a domain ontology is simplistic. The idea would be to use an ontology with greater expressive power to be closer to domain semantics and concepts. Thus further studies should be conducted on the desirable ontology characteristics. Most end-user reporting tools display results with a tabular display such as the one used in this paper. This graphic interface is far from being adapted to display loads of keywords or textual data. Future efforts should also be oriented on a new display with a greater expressive power. Finally, keyword measures are part of a greater family of textual measures: *elaborated textual measures*, we intend to focus on a more general framework for all types of textual measures.

REFERENCES

Agrawal R., Bayardo R.J., Srikant R., 2000. "Athena: Mining-based Interactive Management of Text Databases", *7th Int. Conf. on Extending Database Technology* (*EDBT 2000*), LNCS 1777, Springer, pp. 365-379.

Bender M.A., Farach-Colton M., 2000. "The LCA Problem Revisited", *4th Latin American Symposium on Theoretical Informatics* (*LATIN 2000*), LNCS 1776, Springer-Verlag, pp. 88-94.

Chakrabarti S., Dom B., Agrawal R., Raghavan P., 1998. "Scalable Feature Selection, Classification and Signature Generation for Organizing Large Text Databases into Hierarchical Topic Taxonomies", in *The VLDB Journal*, vol.7(3), pp. 163-178.

Golfarelli M., Maio D., Rizzi S., 1998. "The Dimensional Fact Model: a Conceptual Model for Data Warehouses", in *Int. Journal of Cooperative Information Systems*, vol. 7, n. 2&3.

Gyssen M., Lakshmanan L.V.S., 1997. "A Foundation for Multi-Dimensional Databases", in *23rd Int. Conf. on Very Large Data Bases* (*VLDB 1997*), pp. 106–115.

Harel D., Tarjan R.E., 1984. "Fast algorithms for finding nearest common ancestors", in *SIAM Journal on Computing Archive*, vol.13(2), pp. 338-355.

Horner J., Song I-Y., Chen P.P., 2004. "An analysis of additivity in OLAP systems", *ACM 7th Int. Workshop on Data Warehousing and OLAP* (*DOLAP 2004*), ACM, pp. 83-91, 2004.

Keith S., Kaser O., Lemire D., 2005. "Analyzing Large Collections of Electronic Text Using OLAP", *29th Conf. Atlantic Provinces Council on the Sciences* (*APICS 2005*), Wolville, Canada.

Khrouf K., Soulé-Dupuy C., 2004. "A Textual Warehouse Approach: A Web Data Repository", in *Intelligent Agents for Data Mining and Information Retrieval*, Masoud Mohammadian (Ed.), Idea Publishing Group, pp. 101-124.

Kimball R., 1996. "*The data warehouse toolkit*", Ed. John Wiley and Sons, 2nd ed. 2003.

Lassila O., McGuinness D.L., 2001. "The Role of Frame-Based Representation on the Semantic Web", *Knowledge Systems Laboratory Report KSL-01-02*, Stanford University. (Also appeared in *Computer and Information Science*, Vol.6(5), Linköping University, 2001).

McCabe C., Lee J., Chowdhury A., Grossman D. A., Frieder O., 2000. "On the design and evaluation of a multi-dimensional approach to information retrieval", *23rd Annual Int. ACM Conf. on Research and Development in Information Retrieval* (*SIGIR 2000*), ACM, pp. 363-365.

Mothe J., Chrisment C., Dousset B., Alau J., 2003. "Doc-Cube: Multi-dimensional visualisation and exploration of large document sets", in *Journal of the American Society for Information Science and Technology* (*JASIST*), vol.54(7), pp. 650-659.

Niemi T., Niinimäki M., Nummenmaa J., Thanisch P., 2002. "Constructing an OLAP cube from distributed XML data", *5th ACM Int. Workshop on Data Warehousing and OLAP* (*DOLAP 2002*), ACM, pp.22-27.

Park B-K., Han H., Song I-Y., 2005. "XML-OLAP: A Multidimensional Analysis Framework for XML Warehouses", *6th Int. Conf. on Data Warehousing and Knowledge Discovery* (*DaWaK 2005*), LNCS 3589, Springer, pp.32-42.

Pérez J.M., Llavori R.B., Aramburu M.J., Pedersen T.B., 2005. "A relevance-extended multi-dimensional model for a data warehouse contextualized with documents", *8th ACM Int. Workshop on Data Warehousing and OLAP* (*DOLAP 2005*), ACM, pp.19-28.

Pokorný J., 2001. "Modelling Stars Using XML", in *Proc. 4th ACM Int. Workshop on Data Warehousing and OLAP* (*DOLAP 2001*), pp.24-31.

Ravat F., Teste O., Zurfluh G., 2006. "Constraint-Based Multi-Dimensional Databases", Chapter XI in *Database Modeling for Industrial Data Management*, Zongmin Ma (ed.), IDEA Group, pp.323-368.

Sullivan D., 2001. *Document Warehousing and Text Mining*, Wiley John & Sons.

Torlone R., 2003. "Conceptual Multidimensional Models", Chapter III in *Multidimensional Databases: Problems and Solutions*, M. Rafanelli (ed.), Idea Group, pp.69-90.

Tseng F.S.C., Chou A.Y.H, 2006. "The concept of document warehousing for multi-dimensional modeling of textual-based business intelligence", in *journal of Decision Support Systems* (*DSS*), vol.42(2), Elsevier, pp. 727-744.

Vrdoljak B., Banek M., Rizzi S., 2003. "Designing Web Warehouses from XML Schemas", *5th Int. Conf. on Data Warehousing and Knowledge Discovery* (*DaWaK 2003*), LNCS 2737, Springer, pp.89-98.

Yin X., Pedersen T.B., 2004. "Evaluating XML-extended OLAP queries based on a physical algebra", *7th ACM Int. Workshop on Data Warehousing and OLAP* (*DOLAP 2004*), ACM, pp.73-82.

Zhang J., Ling T.W., Bruckner R.M., Tjoa A.M., 2003. "Building XML Data Warehouse Based on Frequent Patterns in User Queries", *5th Int. Conf. on Data Warehousing and Knowledge Discovery* (*DaWaK 2003*), LNCS 2737, Springer, pp.99-108.