A Conceptual Model for Multidimensional Analysis of Documents

Franck Ravat[2], Olivier Teste[1], Ronan Tournier[1], Gilles Zurlfluh[2]

[1]IRIT, Université Toulouse 3, 118 route de Narbonne
F-31062 Toulouse Cedex 9, FRANCE
[2]IRIT, Université Toulouse 1, 2 rue du doyen G. Marty
F-31042 Toulouse Cedex 9, FRANCE
{ravat,teste,tournier,zurfluh}@irit.fr

Abstract. Data warehousing and OLAP are mainly used for the analysis of transactional data. Nowadays, with the evolution of Internet, and the development of semi-structured data exchange format (such as XML), it is possible to consider entire fragments of data such as documents as analysis sources. As a consequence, an adapted multidimensional analysis framework needs to be provided. In this paper, we introduce an OLAP multidimensional conceptual model without facts. This model is based on the unique concept of dimensions and is adapted for multidimensional document analysis. We also provide a set of manipulation operations.

Keywords. Conceptual modelling, OLAP, Data warehouse, Document warehouse, Multidimensional analysis.

1 Introduction

OLAP (On-Line Analytical Processing) systems allow analysts to improve decision-making process by consulting and analysing aggregated historical business or scientific data. Such analyses are based on a centralized data resource management system, called a data warehouse [11].

1.1 Context and Motivation

The use of Multidimensional Databases (MDB) provides a global view of corporate data warehouse, and allows decision makers to gain insight into an enterprise performance through fast and interactive access to data. Within these databases, multidimensional modelling [11] represent data as points in a multidimensional space. To design MDBs, structures have been defined. These structures model data through the concepts of subjects of analysis, named *facts*, linked to the concept of analysis axes, named *dimensions* [11]. They compose a *star schema* [11]. Facts are groupings of analysis indicators or *measures*. Dimensions are composed of hierarchically ordered *parameters* which model the different detail levels or data granularities [11].

Transactional data may easily be processed because multidimensional analysis is robust and it is a well-mastered technique on numeric-centric data warehouses [24]. But decision support systems have only excavated the surface layers of the task. Only 20% of corporate information system data is transactional, i.e. numeric [27]. The remaining 80%, namely "digital paperwork," stays out of reach of OLAP technology due to the lack of tools and resource management for non-numeric data such as text-rich documents. In order to provide increased analysis capacities, decision support systems should provide the use of 100% of all available data from corporate information systems. Analysts should be able to integrate text-rich documents or web data directly into the analysis process along with business data. Not taking into account these data sources would inevitably lead to the omission of relevant information during an important decision-making process or the inclusion of irrelevant information and thus producing inaccurate analyses [27].

OLAP provides powerful tools and methods but within a rigid framework. Unfortunately text is not as structured as data warehouse systems would tolerate. Recently, XML technology has provided a wide framework for sharing and working with documents within corporate networks or over the web. Thus, documents stored as semi-structured data were slowly integrated within data warehouses and repositories. Document warehousing slowly emerged [24], e.g. Xyleme[1]. On-line text and documents are now becoming conceivable data sources for multidimensional analysis.

By *multidimensional document analysis* throughout this paper we mean to analyse in an OLAP environment text-rich document data sources. To cope with the rigid framework inherited from the context, i.e. the data warehouse environment, we shall only consider structured or semi-structured sources (e.g. XML documents). For example, conference proceedings, patient files from a hospital information system, quality control reports…

In text-rich documents, internal data is almost exclusively text. As this data type is non-additive and non-numeric, traditional aggregation operations (sum, average…) will not work, thus there is a necessity for adapted aggregation operations. [18] lists a few ones inspired from text mining techniques and we defined in [21] an aggregation function for keywords. Within this paper, as an illustration, we shall use: TOP_KEYWORDS [18]. This aggregation function selects the major keywords of a text.

1.2 Motivating Example

In the following example, an analyst wishes to analyse the citations of some works of a research institute. The analysis task would be counting each time an author is cited in conference proceedings and display the results by author and by conference. For example, in the following table (Table 1a), author *A1* has been cited three times by *ER* authors. The analyst may then wish to determine the range of the researcher's works and analyse the subjects of the publications where the researcher's works are cited. As this analysis does not rest on traditional numerical data, but on factual data, the analyst will use the TOP_KEYWORDS function in order to display the two main keywords of the documents. These keywords will be aggregated per conference,

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

hence giving a list of subjects instead of a number of publications. For example, in Table 1b, the three citations of the works of the author *A1* in *ER* conferences are related to *XML* and *Documents* topics. Author *A3* has always been cited in the same context (data mining), A3's works have a narrower range than works of A1 and A2.

Table 1. The number of times an author has been cited in a particular conference and the same analysis with the major keywords of the publications that cite the authors.

a)

Institute	Inst1		
Author / Conference	A1	A2	A3
ER	3	2	1
SSDBM	2	-	-
DaWaK	1	1	2

b)

Institute	Inst1		
Author / Conference	A1	A2	A3
ER	XML, Docments	XML, Data Warehouse	Data Mining, Clustering
SSDBM	XML, Temporal DB	-	-
DaWaK	Data mining	Data mining	Data Mining, Clustering

The combination of these analyses would be very complex to model using traditional multidimensional modelling. Firstly, the analysis of textual data is not taken into account. Secondly, the analyst would need more than one data mart [11]. And thirdly, multidimensional modelling approaches explode the document structure into many separate elements requiring complex and tedious tasks from the data warehouse administrator. The two following subsections present the related works, the objectives and contributions of the paper.

1.3 Related Works

According to [5], two types of semi-structured documents may be found:

- *Data-centric documents* are raw data documents, mainly used by applications to exchange data (as in e-business application strategies). In this category, one may find lists and logs such as: invoices, orders, spreadsheets, or even "dumps" of databases.
- *Document-centric documents* also known as text-rich documents are the traditional paper documents, e.g. scientific articles, e-books, website pages.

The analysis of data-centric documents has been introduced in several propositions such as [9]. See [29,28,17] for a more complete list of these works. Although all these works consider textual data through the use of XML documents, these propositions do no take into consideration the more complex document-centric documents. As a consequence, this paper focuses on the analysis of document-centric documents.

We divide related works into two categories. The first category concerns multidimensional modelling and may be divided into two approaches:

1. Traditional conceptual multidimensional modelling: lists of works may be found in [25,22] and current issues are highlighted in [23]. Multidimensional modelling is based on the concepts of facts and dimensions. These models, conceptually or logically oriented, have been conceived for numeric data analysis and do not deal with documents where non-numeric data must be integrated. Extending these mod-

els is possible but would require dimensions using abnormal hierarchies and complex logical implementations.
2. Multidimensional modelling allowing complex data analysis: in [17] the authors defines an xFACT, a complex hierarchical structure containing structured and unstructured data (such as documents), where measures, called contexts, can be seen as complex object data. In [3], the authors present a complete XML approach for modelling complex data analysis. Although this proposition takes into account complex data, the authors only use data-centric documents.

Current multidimensional modelling propositions are incomplete as they address only numeric analysis. To our knowledge there are no propositions taking into account document-centric document properties.

The second category concerns the addition of document-centric documents into multidimensional analysis. Within this category there are three approaches:

1. To assist multidimensional analysis by providing complementary documents: in [19] the authors combine traditional numeric analysis and information retrieval techniques to assist an analysis by providing documents relevant to the ongoing analysis context. The user must then read all retrieved documents.
2. To provide multidimensional OLAP analysis of documents: in [15,16,10,27], the authors present applications of document-centric document multidimensional analysis using a star schema. The authors propose the use of traditional OLAP framework to count documents according to keywords or topics organised into dimensions. These dimensions allow the user to analyse the number of documents represented by each keyword according to several analysis axes. Using a keyword dimension is limitative as no text analysis may be performed. In [27] the authors classify dimensions according to categories, but exclusively work on document meta-data (except for the keyword dimension). They neither take into account document structures nor document contents. Although limited, some commercial solutions start to appear, such as Text OLAP[2].
3. To analyse textual data directly: in [12] the authors describe a document warehouse where documents are grouped by structure families. Users can perform multidimensional analysis on documents or on structures, but limited to numeric analysis (numbers of documents or structure types). Finally, in [18], the authors describe a logical model based on the xFACT and specific aggregation functions inspired by text mining techniques for the multidimensional analysis of document-centric documents. More complete than [26], the authors provide adapted aggregation functions but do not detail them. The model lacks high level concepts that may easily be manipulated by decision makers. Moreover, apart from the informal description of aggregation functions, no adapted manipulation operations are presented.

These advanced propositions clearly show the limit of traditional models for analysing documents: 1) the suggested implementations never preserve document structure; 2) these structures remain unexploited; 3) non-numeric indicators cannot be

[2] Text OLAP, Megaputer, Polyanalyst Suite from http://www.megaputer/com/products/pa

handled easily; and as a consequence, 4) no flexibility is provided to the user in changing the focus of analysis.

So far, to our knowledge, there is no proposal for an adapted conceptual model for document-centric document analysis. Up to now, apart from [18], research works are based on quantitative analysis, e.g. the number of publications that contain a specific keyword. Textual data is provided for analysis through dimensions modelling analysis axes and not subjects of analysis. Analysis indicators (measures) are always numeric.

1.4 Aims and Contributions

In this paper, as a first step to a more global framework for integrating documents in an OLAP system, we define a conceptual model adapted to the multidimensional analysis of documents. The aim of this model is to provide the analyst with a simple and adapted conceptual view [6], withdrawing all logical and physical constraints. In order to manipulate the concepts of the model, OLAP operations are revised.

The model has been designed to be used for scientific data sets, like the IASI[3] archive of the UMARF[4] facility. Although the facility holds numeric data, it mainly holds complex factual data (e.g. spectral data). Atmospheric research requires several complex analyses with many facts and dimensions. This would lead to the design of several data marts with a high redundancy between factual and dimensional data. Moreover, in a traditional model, an extensive use of complex operations to convert dimensions into facts and vice-versa would have been necessary [20]. Due to space limitation, we shall use throughout this paper an example of analysis of scientific publications and conference proceedings to monitor research activities.

The conceptual model has to ease the analyst's tasks and take into account document-centric documents characteristics. Firstly, these documents are composed of tree-like hierarchically structured data. Secondly, a document might refer to itself or to other documents (*e.g. hypertext links*). These links should be explicitly shown in order to ease understanding and navigation through data during analyses. For example, when analysing the references of a publication, the analyst has to clearly see (and not to guess) that a reference is nothing else than another publication. And thirdly and most important of all, when analysing documents, textual computer assisted analysis does not necessarily make sense. That is, when analysing a particular subject, the analyst may find himself in front of something lacking sense. Thus, the analyst must be able to easily change the subject and not be restrained by predefined subjects of analysis. In conclusion, the model needs to be: 1) able to represent document-centric data specificities; 2) ease the representation avoiding to provide the analyst with predefined and limited analysis solutions; and 3) ease manipulations of the concepts through a set of operations. To answer these objectives, we define a Galaxy model associated to a set of manipulation operations.

---

[3] **IASI**: Infrared Atmospheric Sounding Interferometer (http://smsc.cnes.fr/IASI/)

[4] **UMARF**: Unified Meteorological Archive and Retrieval Facility of EuMetSat (European Meteorology).

The rest of the paper is organized as follows: section 2 defines an adapted multi-dimensional model; section 3 presents a set of multidimensional operations. Finally section 4 concludes the paper and states future works.

2 Multidimensional model

The model defined in the following section is based on a "factless multi-dimension" representation of a constellation schema. In this model, there are only analysis axes, named dimensions. These dimensions are gathered into groups to indicate compatible dimensions for a common analysis.

2.1 Grouping dimensions in "Galaxies"

A dimensional scheme is conceptual grouping of dimensions. It is a generalisation of a constellation [11], and is nicknamed a "Galaxy" schema. Dimensions are grouped around nodes that model the dimensions that may be used together in a same analysis.

Definition: A *Galaxy* $G = (D^G, Star^G, Lk^G)$ where
- $D^G = \{D_1,\ldots, D_n\}$ is a set of *dimensions*,
- $Star^G : D_i \rightarrow 2^{D_j}$ is a function that associates each dimension D_i to its linked dimensions $D_j \in D^G$ ($D_j \neq D_i$). This expression models nodes c_z that may be expressed through: $\{D_{c1},\ldots,D_{cn}\} \subseteq D^G \mid \forall i,j \in [c_1..c_n],\ i \neq j,\ \exists\ D_i \rightarrow 2^{Dj} \in Star^G$. This represents dimensions compatible within a same analysis.[5]
- $Lk^G = \{g_1, g_2,\ldots\}$ is a set of functions associating some attribute instances together through links, where $g^G : a_u^{D_i}\left(i_x^{D_i}\right) \rightarrow a_v^{D_j}\left(i_y^{D_j}\right)$ is the association of the instance ${i^{Di}}_x$ of ${a^{Di}}_u$ with the instance ${i^{Dj}}_y$ of ${a_v}^{Dj}$, where $(D_i=D_j)$ or $(D_i,D_j) \in D^G \mid D_j \in Star^G(D_i)$.

Links (Lk^G) represent "corresponds to" relationships between the values of the two attributes of the link. They are used within expressions of manipulation operations.

Notations. We note $D_j \in Star^G(D_i)$, the fact that D_i and D_j are linked together.

2.2 Dimension concept

A dimension models an analysis axis along which data may be analysed. A dimension is characterized by hierarchically organised attributes, each attribute being a graduation of the analysis axis, i.e. detail levels or granularity levels.

Definition: A *dimension* $D=(A^D, H^D, I^D, IStar^D)$ where:
- $A^D = \{{a^D}_1,\ldots, {a^D}_r\}$ is a set of *attributes*,
- $H^D = \{{H^D}_1,\ldots, {H^D}_s\}$ is a set of *hierarchies*,
- $I^D = \{{i^D}_1,\ldots, {i^D}_t\}$ is a set of *dimension instances*. Each attribute has a value for each instance ${a^{Di}}_u({i^{Di}}_x)$, called an *attribute instance*.

[5] The notation 2^E represents the powerset of E.

– $IStar^D : I^D \rightarrow \left(I^{D_1}\right)^* \times \ldots \times \left(I^{D_n}\right)^*$ is a function that associates the instances of the D dimension to the instances of other linked dimensions through $Star^G$ ($\forall k \in [1..n]$, $D_k \in D^G$, $D_k \neq D$ and $D_k \in Star^G(D)$, i.e. D_k is associated/linked to D).[6]

A hierarchy represents an analysis perspective within a dimension. It models the organisation of the different granularity levels, i.e. a particular view of the analysis axis graduations. A hierarchy H^D_i of D is an ordered list of attributes called parameters. It is an acyclic elementary path starting with the parameter of finest granularity and ending with one of coarsest granularity. Each parameter may be associated to weak attributes which represent complementary information.

Definition: A *hierarchy* noted H^D_i or $H = (Param^H, Weak^H)$ where:

– $Param^H = <p^H_1, \ldots, p^H_{np}>$ is an ordered set of attributes, called *parameters*, which represent the levels of granularity of the dimension, $\forall k \in [1..n_p]$, $p^H_k \in A^D$ and $p^H_1 = a^D_1$;

– $Weak^H : Param^H \rightarrow 2^{A^D - Param^H}$ is an application possibly associating *weak attributes* to parameters, completing the parameter semantic.

Attributes are of two types: a parameter is the data of a particular level of detail, e.g. a *research institute* or the *country* of a research institute; a weak attribute is complementary data of a parameter, such as the *name* or the *address* of a *research institute*. All hierarchies of a dimension start with a common root parameter ($\forall H_i \in H^D$, $p^{Hi}_1 = a^D_1$) and end by a parameter representing the coarser granularity (p^{Hi}_{np}).

To answer to document structure specificity, hierarchies are semantically richer than traditional hierarchies. This provides the analyst with a conceptual view as close as possible as document representation. Thus, dimensions modelling documents may use non-strict hierarchies [13].

Notations. $p_i \in H$ is a simplified notation for $p_i \in Param^H$. Whenever possible, if the context is obvious, notations H^D, $p^H_i(\ldots)$ will be simplified by H, $p_i(\ldots)$.

2.3 Example

In order to analyse the activity of research institutes, a decision-maker analyses scientific publications as well as reports produced by these institutes. To answer these requirements, the galaxy G_1 is created (see Fig. 1). It represents on the top part: articles published in a conference at a certain date and written by authors; and on the bottom part: scientific reports. Within this example, two recursive links may be used to navigate through 1) the references of articles and 2) the institutes of authors.

Galaxy G_1 example: $D^{G1} = \{D^{Conferences}, D^{Articles}, D^{Time}, D^{Authors}, \ldots\}$
$Star^{G1} = \{D^{Conferences} \rightarrow (D^{Article}, D^{Time}, D^{Authors}), \ldots\}$
$Lk^{G1} = \{g_{References}: a^{Articles}_{References} \rightarrow a^{Articles}_{Article}, \ldots\}$
Dimension example: $D^{Conferences} = \{A^{Conferences}, H^{Conferences}, I^{Conferences}, IStar^{Conferences}\}$
$A^{Conferences} = \{a_{Conference}, a_{Name}, a_{Publisher}, a_{Status}\}$; $H^{Conferences} = \{HPu, HSt\}$;
$I^{Conferences} = \{i^{Conference}_1, \ldots i^{Conference}_q\}$

[6] The notation $(I)^*$ represents a finite set of elements of I.

$IStar^{Conferences}=\{\ i^{Conference}{}_k \rightarrow \{(i^{Articles}{}_{rk})^*,\ (i^{Time}{}_{sk})^*,\ (i^{Authors}{}_{tk})^*\}\ |\ \forall k \in [1..q],\ i^{Conference}{}_k \in I^{Conferences} \wedge \exists\ i^{Articles}{}_{rk} \in I^{Articles} \wedge \exists\ i^{Time}{}_{sk} \in I^{Time} \wedge \exists\ i^{Authors}{}_{tk} \in I^{Authors}\}$

Hierarchy example: $HPu=\{Param^{HPu}, Weak^{HPu}\}$

$Param^{HPu}=<a^{Conference}, a^{Publisher}>$ and $Weak^{HPu}=\{\ a^{Conference} \rightarrow \{a^{Name}\}\}$

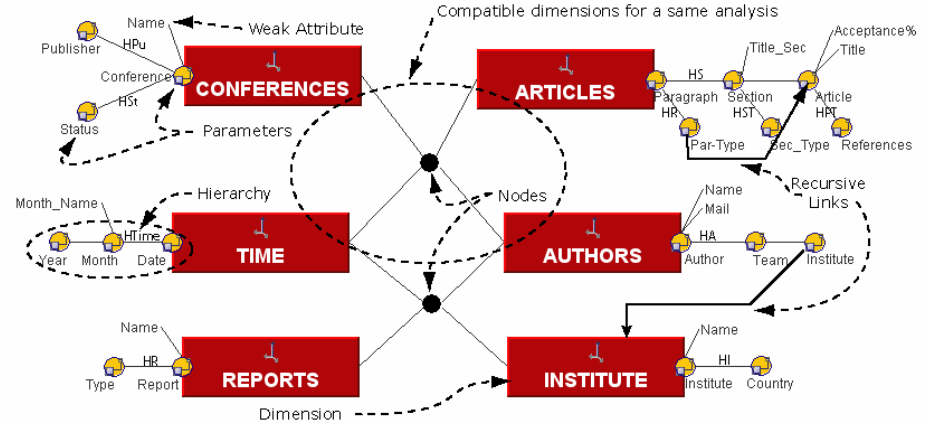

Fig. 1. Example of a Galaxy scheme (G_1): analysis of scientific publications and reports

Dimensions are used to define multidimensional analyses with a set of manipulation operations described in the following section. Links ease analysis expressions.

3 Multidimensional operations

In order to manipulate concepts represented by the galaxy model, analysts need four operations that slightly differ from traditional OLAP operations [20]. The operations are based on the following needs:

- A focussing operation is necessary to select the subject of analysis projecting the subject data on several analysis axes.
- To narrow the analysis spectrum the user needs an operation to select a particular subset of data, thus reducing the whole quantity of analysed data.
- To take advantage of hierarchically ordered parameters, the user will need drilling operations to change the level of detail of the analysed data. The user will need two drilling operations: One to zoom into the details of the analysed data and the other for reversing the process, zooming out of the data.
- To change the analysis criteria, an operation is necessary either to rotate the analysed subject around other analysis axes or to rotate the analysis axes around different subjects.

In some models, authors pointed out the necessity of symmetric treatment of parameters and measures to ease definition and conception of algebras or calculus [2,4,8]. But some specific operations such as drilling did not operate symmetrically between all attributes. This problem put aside with our model.

Notations. $dom(D_i)$ is the domain of the dimension D_i, i.e. all $i_x \in I^{Di}$. We note $(dom(D_i))^*$ a finite set of elements of $dom(D_i)$.The instances of a galaxy G, composed of n dimensions, are represented by (1). All the instances of the attributes $a_j \in A^{Di}$ of dimension D_i are represented by (2). We define an aggregation function f_{AGG} (3) where $dom(f_{AGG}(dom(D_i)))$ corresponds to the domain of the aggregated values of the domain of the dimension D_i. In order to compare levels between parameters within a hierarchy H, we introduce the function *level* (4).

$$dom(D_1) \times \ldots \times dom(D_n) = \prod_{i=1}^{n} dom(D_i) = dom(G) \quad \textbf{(1)}$$

$$dom(D_i.a_1) \times \ldots \times dom(D_i.a_n) = \prod_{j=1}^{\|A^{D_i}\|} dom(D_i.a_j) = dom(D_i) \quad \textbf{(2)}^7$$

$$f_{AGG} : (dom(D_i.p_j))^* \rightarrow dom(f_{AGG}(dom(D_i.p_j))) \quad \textbf{(3)}$$
$$(x_1, \ldots, x_m) \mapsto f_{AGG}(x_1, \ldots, x_m)$$

$$\text{Given } Param^H = < p_1, \ldots, p_{np} >,\ level^H(p_1) = 1, \ldots, level^H(p_{np}) = n_p \quad \textbf{(4)}$$
$$\text{and } \forall j \in [1..n],\ level^H(p_j) \leq level^H(p_{np})$$

All operations produce compatible outputs. The focus operation generates as output a subset of the galaxy, named s^G, and this subset is used as input for all other operations. In their turn these operations produce a subset, that allows chaining operations one after the other. The operation syntax is as follows:

OPERATION_NAME(*input*, *operation_parameters*) = *output* .

3.1 Focussing and Selection Operations

These two major manipulation operations allow the specification of analysis datasets.

Focussing is used to define an analysis subject and to project subject data on several analysis axes. Concretely, this operation allows the specification of a subject of analysis (*DS*) aggregating the analysis data through an aggregation function (f_{AGG}) for each selected measure according to the detail levels selected in the analysis axes.

Syntax: *FOCUS*(G, S, P)=s^G where G is the input (a galaxy), S=(f_{AGG}($DS.HS.p_i$)) is the focused subject of analysis with the parameter p_i of the hierarchy *HS* of dimension *DS* aggregated through the function f_{AGG} and P=(($D_x.H_x$, $Param_x$),($D_y.H_y$, $Param_y$),…) is the set of projection axes with D_x being the dimension selected as the first analysis axis, D_y the second,… H_x is the current hierarchy of the axis represented by D_x, H_y is the current hierarchy of D_y,… $Param_x$=<p_{x_min},…,p_{x_max}> is an ordered set of parameters of H_x, where given $Param^{Hx}$=<p_1,…,p_{np}>, $level^{Hx}(p_{x\_min}) \geq level^{Hx}(p_1)$ and $level^{Hx}(p_{x\_max}) \leq level^{Hx}(p_{np})$. $Param_x$ represents the selected parameters of D_x (it is a subset of $Param^{Hx}$). In the same way $Param_y$ is a subset of $Param^{Hy}$.

[7] We recall that $\|A^{Di}\|$ is the cardinality of A^{Di}. Here, the number of attributes within A^{Di}, i.e. r.

Conditions: $\forall D_i \in P$, $D_i \in Star^G(DS)$, i.e. the dimensions selected as analysis axes are linked to the dimension selected as subject (DS). The aggregation function f_{AGG} must be compatible with the parameter instances of p_i that are to be aggregated.
Mathematically: *Focus* (7) = *Aggregation* (6) ∘ *Projection* (5) where:

$$\prod_{i=1}^{n} dom(D_i) \xrightarrow{PROJECT} (dom(DS.p_i))^* \times \prod_{j=1}^{\|P\|} \left(\prod_{k=\min}^{\max} dom(D_j.p_k) \right) \quad (5)$$

$$(dom(DS.p_i))^* \times \prod_{j=1}^{\|P\|} \left(\prod_{k=j\_\min}^{j\_\max} dom(D_j.p_k) \right) \xrightarrow{AGGREGATE} dom(f_{AGG}(dom(DS.p_i))) \times \prod_{j=1}^{\|P\|} \left(\prod_{k=j\_\min}^{j\_\max} dom(D_j.p_k) \right) \quad (6)$$

$$\prod_{i=1}^{n} dom(D_i) \xrightarrow{FOCUS} dom(f_{AGG}(dom(DS.p_i))) \times \prod_{j=1}^{\|P\|} \left(\prod_{k=j\_\min}^{j\_\max} dom(D_j.p_k) \right) \quad (7)$$

We also define a simplified notation (8), where s^G represents a subpart of the galaxy with a dimension designated as subject (S_{AGG}) analysed (projected and aggregated) according to the dimensions of the projection set (P).

$$dom(G) \xrightarrow{FOCUS} dom(s^G) \text{ with } dom(s^G) = dom(S_{AGG}) \times dom(P) \quad (8)$$

Example. Within the galaxy presented in figure 1 (G_1), the analyst may use any dimension as a subject of analysis. Here, the analyst focuses his analysis on major keywords of articles displayed by author and by year. We will suppose that the user uses a bi-dimensional table to produce the output, [8,22]. The user will thus focus on a dimension (DS) and project its data onto two analysis axes: a line dimension and the column dimension. The aggregation function TOP_KEYWORDS returns the two major keywords. The following instruction produces the table displayed in the following figure.

FOCUS (G_1, *TOP_KEYWORDS*(*ARTICLES.HS.Section*),
((*TIME.HTime*, <*Year*>), (*AUTHORS.HA*, <*Author*>))) = $s^{G1}{}_1$

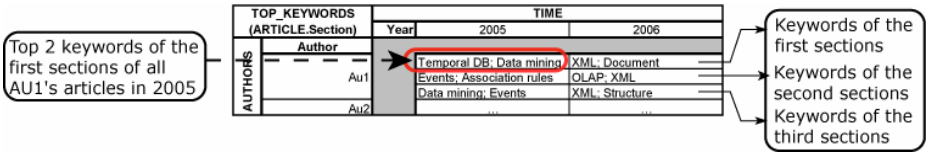

Fig. 2. Example of manipulations: focus instruction projecting analysis subject data onto two analysis axes (years and authors)

Selection is used to restrict the analysis data. By specifying a restrictive predicate, the user may restrict analysis data on an analysis axis or on the analysis subject. All instances selected by a predicate p are maintained in the current data selection. All other instances are removed. Notice that if this operation is applied directly on the galaxy, this allows the removal of instances before aggregation process.
Syntax: *SELECT*(G, p)= $s^{G'}$ or *SELECT*(s^G, p)= $s^{G'}$ where G (or s^G) is the input and p is a restrictive predicate on an attribute a_j of a dimension D_i.
Conditions: $a_j \in D_i$ and $D_i \in Star^G(DS)$.

Mathematically:

$$dom(G) \xrightarrow{SELECT} dom(G) - dom(\neg p) \text{ or } dom(s^G) \xrightarrow{SELECT} dom(s^G) - dom(\neg p) \quad (9)$$

The notation $dom(\neg p)$ is the subset of the domain that does not satisfy the predicate p. The reverse operation, $UNSELECT(s^G) = s^{G'}$, removes all restrictive predicates.

Example. In order to narrow the analysis spectrum, the analyst decides to reduce the analysis to only *Au1*'s articles and to analyse major keywords only in introductions. Using the previously defined subset of data (s^G), the following instructions produce the table (b) displayed in the following figure:

SELECT(*SELECT*(s^{G1}_1, *ARTICLE.Sec_Type*='*Introduction*'),
AUTHORS.Author='*Au1*') = s^{G1}_2

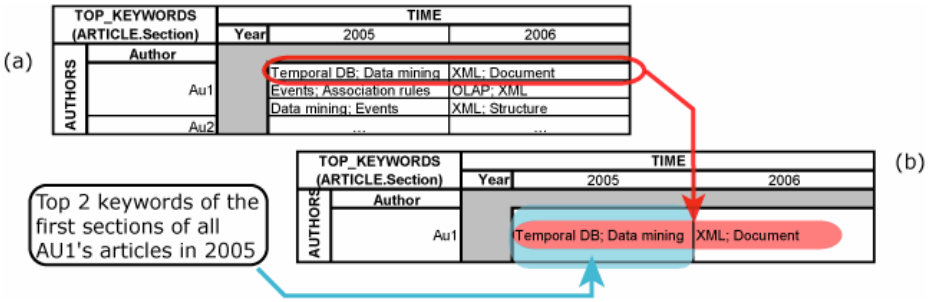

Fig. 3. Example of manipulations: application of two restrictions

3.2 Drilling Operations

Once an analysis has been specified, i.e. s^G has been defined, the user may wish to change the level of detail at which analysis data is being projected.

Using a drill-down operation, the analyst may zoom into more detailed data. This operation consists in adding to the parameter list of a projection axis ($Param_i$) a new parameter p_{new}, from the current hierarchy, whose level is inferior to the lowest currently selected parameter (p_{min}).

Syntax: *DRILLDOWN*(s^G, D_i, p_{new})=$s^G{}_1$ where s^G is the input, D_i is a dimension of the projection set P of s^G, i.e. $\exists(D_i.H_i, Param_i) \in P$ and $p_{new} \in H_i$.

Condition: The parameter must be of a lower level than the lowest one already selected: $level^{Hi}(p_{new}) < level^{Hi}(p_{min})$

Mathematically:

$$DrillDown: dom(s^G) \xrightarrow{DRILLDOWN} dom(S_{AGG}) \times dom(P) \times dom(D_i.p_{new})$$
$$\text{where} \quad dom(P) = \prod_{\substack{j=1 \\ j \neq i}}^{\|P\|} \left(\prod_{k=j\_\min}^{j\_\max} dom(D_j.p_k) \right) \times \prod_{k'=i\_\min}^{i\_\max} dom(D_i.p_{k'}) \quad (10)$$

Note that, $dom(P)$ represents the domains of the selected parameters of the dimensions not taking part in the drilling operations ($\forall D_j \mid \exists(D_j.H_j, Param_j) \in P$ and j≠i) as

well as the domains of the selected parameters of the dimensions taking part in the drilling operation (D_i). We recall that $Param_j$=<p_{j_min},…,p_{j_max}>.

The opposite operation, roll-up, is used to gain a more global view of the analysis data. This operation is used to zoom out of the analysis data. This operation consists in removing all parameters from the selected parameter list ($Param_i$) whose levels are lower than a selected parameter. The operation will eventually add the parameter had it not been in the list.

Syntax: $ROLLUP(s^G, D_i, p_{sup})= s^G{}_1$ where s^G is the input, D_i is a dimension of the projection set P of s^G, i.e. $\exists(D_i.H_i, Param_i)\in P$ and $p_{sup}\in H_i$.

Condition: The parameter must be of a higher level than the lowest one already selected: $level^{Hi}(p_{sup})>level^{Hi}(p_{min})$.

Mathematically: in the following, we express $level^{Hi}(p_{sup})=sup$

$$RollUp: dom(s^G) \xrightarrow{ROLLUP} dom(S_{AGG}) \times \prod_{\substack{j=1 \\ j\neq i}}^{\|P\|} \left(\prod_{k=j\_min}^{j\_max} dom(D_j.p_k) \right) \times \prod_{k'=sup}^{i\_max} dom(D_i.p_{k'}) \quad (11)$$

Here, $\prod_{k'=sup}^{i\_max} dom(D_i.p_{k'})$ is the domain of the parameters of the dimension taking part in the drilling operation (D_i). The domains of the parameters whose levels are inferior to p_{sup} are removed (thus k' minimal bound is $level^{Hi}(p_{sup})=sup$).

Example. As in traditional models, the drill-down operation could be used to display the keywords by months rather than by year. But in our model, this operation may also be applied on the current hierarchy of the focused dimension. This is critical when textual aggregation functions produce results lacking sense as it enables users to gain insight within the aggregation process. In the following example, rather than analyzing keywords by section, the analyst decides to analyse then by subsections. The following instruction produces the table displayed in the following figure:

$DRILLDOWN(s^{G1}{}_2, ARTICLE, Subsection) = s^{G1}{}_3$

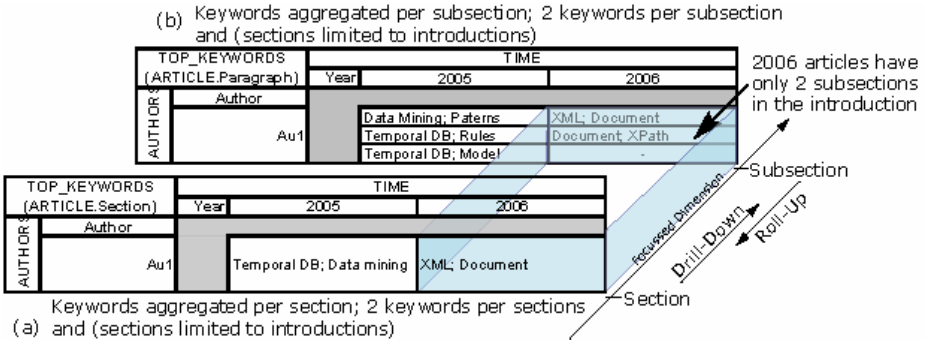

Fig. 4. Example of manipulations: drilling on the focused dimension: *Article*.

Drilling on the focused hierarchy allows powerful combination of 1) the usage of the hierarchical model provided by the hierarchical structure of the dimension data; and 2) the usage of the aggregation process allowing the summarisation of selected data. Drilling on the focussed hierarchy may be seen as adding in the manipulation a "third" analysis axis.

This operation allows the user to gain insight within the aggregation process. This is due to the fact that textual aggregation functions do not operate like numeric aggregation functions. Indeed, extracting the major keywords of an article does not necessarily correspond to the extraction of the major keywords of each section. This is a common problem of 1) holistic functions [7] which may not be computed from lower results (e.g. median function); and 2) component ranking such as pointed out in [14] where in an information retrieval framework different granularities tend to mess up statistics. Physically, when drilling on textual data and using a holistic aggregation function, aggregates are recomputed with the newly designated granularity of the dataset. Thus the analyst may get a better understanding by seeing these different aggregates.

3.3 Analysis Reorganisation Operation

In some cases, the user might wish to reorganise the analysed dataset. To do this, he uses an operation that will change structural elements of the subpart of the galaxy s^G.

The rotation operation replaces by a new dimension one of the dimensions of s^G: the analysis subject (*DS*), or one of the analysis axes, i.e. a dimension from the projection set ($D_i \in P$).

Syntax: $Rotate(s^G, D_{old}, D_{new}.H_{new}, A)=s^G{}_1$ where s^G is the input, D_{old} is the dimension to be replaced, D_{new} is the new dimension, H_{new} its currently selected hierarchy and A depends on D_{old}. If $D_{old}=DS$ then $A=f'_{AGG}(p_{new})$, else if $D_{old} \in P$ then $A = Param_{new} = <p_{new\_min},\ldots, p_{new\_max}>$ is a subset of $Param^{Hnew}$.

Condition: if $D_{old}=DS$ then $\forall D_k \in P$, $D_k \in Star^G(D_{new})$ and $p_{new} \in H_{new}$. If $D_{old} \in P$ then $D_{new} \in Star^G(DS)$ $Param_{new} \subseteq Param^{Hnew}$ and $level^{Hnew}(p_{new\_min}) < \ldots < level^{Hnew}(p_{new\_min})$

Mathematically: if $D_{old}=DS$ then the operation corresponds to (12), else if $D_{old} \in P$, the operation corresponds to (13).

$$Rotate : dom(s^G) \xrightarrow{ROTATE} dom(f'_{AGG}(dom(D_{new}.p_{new}))) \times dom(P) \quad (12)$$

$$Rotate : dom(s^G) \xrightarrow{ROTATE} dom(S_{AGG}) \times \prod_{\substack{j=1 \\ j \neq old}}^{\|P\|} \left(\prod_{k=j\_\min}^{j\_\max} dom(D_j.p_k) \right) \times \prod_{k'=new\_\min}^{new\_\max} dom(D_{new}.p_{k'}) \quad (13)$$

Notice that if $D_{old} = D_{new}$, this allows to change one of the current selected hierarchy (*HS*, H_x, H_y…). Notice also that when rotating the subject of analysis, this is the equivalent of *FRotate* [22] or *DrillAcross* operations [1].

3.4 The Use of Recursive Links

Links within the Galaxy may be used as paths to access particular data. They allow flexibility when designating subparts of documents and simplify query specifications. For example, the following operation sequence uses the link between *Reference* and *ARTICLE* (see figure 1). It focuses on the major keywords of each section of articles that are cited by *Au1*, i.e. the articles in reference sections of all *Au1*'s publications.

SELECT (*SELECT* (*FOCUS* (*TOP_KEYWORDS* (*ARTICLES.HR.Reference.Section*), ((*TIME.HTime*, <*Year*>), (*ARTICLE.Reference.AUTHORS.HA*, <*IdA*>))), *AUTHORS.IdA*='*Au1'*), *ARTICLE.Reference.TIME.Year* > *2005*)

Where *ARTICLE.Reference.AUTHORS* are the authors of the articles referenced by *Au1*'s publications, *ARTICLE.Reference.TIME.Year* are years of publication of the referenced articles whereas *TIME.Year* are the years of publication of *Au1*'s articles.

As another example, the query that provides the table displayed in Table 1 is:

SELECT(*FOCUS*(*TOP_KEYWORDS*(*ARTICLES.HS.Article*),((*ARTICLES.Reference.AUTHORS.HA*,<*Author*,*Institute*>),(*CONFERENCES.HConf*,<*Name*>))),*ARTICLES.References.AUTHORS.Institute*='*Inst1'*)

Where *ARTICLES.References.AUTHORS* are the authors of the articles cited in the conferences *CONFERENCES.Name* in the articles whose content is specified by *ARTICLES.Article*. Notice that hierarchies are specified only in the focus operation to allow drilling operations that follow the hierarchical structure of the parameters.

The links allow more flexibility when querying data sources that are interconnected together, as the links may be used to thoroughly explore and analyse datasets.

4 Conclusion and Future Works

In this paper we have defined an adapted multidimensional conceptual model for the analysis of text-rich documents. The model is based on a unique conceptual element: a dimension. It is associated to a set of manipulation operations to allow multidimensional OLAP analysis.

Contrarily to previous multidimensional models, this proposition has the advantage of preserving the document structure as well as the links within these structures. The usage of links allows thorough analysis of documents interconnected together such as articles that reference other articles. Moreover, these links simplify the expression of queries that would be very complex in other environments. The absence of factual entity does not restrain the analyst with predefined subjects of analysis that might produce analyses lacking sense on text-rich data sources. The associated manipulation operations allow easy switching of the focus of the analysis subject. Hence, the user may compensate the lack of accuracy in textual analysis by an increased flexibility within this OLAP framework. The preservation of the document structure allows analysts to use this structure in order to refine their analyses and perform fine tuning. Notice that facts may still be represented within this model by very simple dimensions, where each measure is a hierarchy with a unique parameter.

Due to lack of space we apologize for not having presented the logical level of this framework. We are currently extending a prototype: GraphicOLAPSQL [22]. This prototype is based on an Oracle 10*g* database, XML files for documents and a Java interface. In our implementation, in order to maintain performance, each dimension is linked to all other dimension instances allowing quick rotation around different subjects, i.e. in the Oracle R-OLAP environment this is physically implemented through VArrays and Nested Tables, depending on index sizes.

This conceptual model is the first step for a more complete framework. Throughout this paper, we have suggested the use of a simple aggregation function

(TOP_KEYWORDS). As future works, we consider the specification of a set of adapted aggregation functions such as AVG_KW [21] for document-centric document analysis. In parallel, as the goal of the conceptual model is to ease the process of analysis, we also intend to adapt and implement a graphical OLAP query language [22].

References

1. Abelló, A., Samos, J., Saltor, F.: Implementing operations to navigate semantic star schemas. 6th ACM int. workshop on Data Warehousing and OLAP (DOLAP), ACM, pp.56–62, 2003.
2. Agrawal, R., Gupta, A., Sarawagi, S.: Modeling Multidimensional Databases. Int. Conf. on Data Engineering (ICDE), pp.232–243, 1997.
3. Boussaid, O., Messaoud, R.B., Choquet, R., Anthoard, S.: 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, pp.39–54, 2006.
4. Cabibbo, L., Torlone, R.: A Systematic Approach to Multidimensional Databases. 5th Italian Symposium on Advanced Database Systems (SEBD), pp.361–377, 1997.
5. Fuhr, N., Großjohann, K.: XIRQL: A Query Language for Information Retrieval in XML Documents. 24th int. ACM SIGIR conf. on Research and development in information retrieval, pp.172–180, 2001.
6. Golfarelli, M., Rizzi, S., Saltarelli, E.: WAND: A CASE Tool for Workload-Based Design of a Data Mart. 10th Italian Symposium on Advanced Database Systems (SEBD), pp.422–426, 2002.
7. Gray, J., Bosworth, A., Layman, A., Pirahesh, H.: Data Cube: A Relational Aggregation Operator Generalizing Group-By, Cross-Tab, and Sub-Total. 12th Int. Conf. on Data Engineering (ICDE), pp.152–159, 1996.
8. Gyssen, M., Lakshmanan, L.V.S.: A Foundation for Multi-Dimensional Databases. 23rd Int. Conf. on Very Large Data Bases (VLDB), pp.106–115, 1997.
9. Jensen, M.R., Møller, T.H., Pedersen, T.B.: Specifying OLAP Cubes On XML Data. In 13th Int. Conf. on Scientific and Statistical Database Management (SSDBM), IEEE Computer Society, pp.101–112, 2001.
10. Keith, S., Kaser, O., Lemire, D.: Analyzing Large Collections of Electronic Text Using OLAP. APICS 29th Conf. in Mathematics, Statistics and Computer Science, pp.17–26, 2005.
11. Kimball, R.: The data warehouse toolkit. John Wiley and Sons, 1996, 2nd ed. 2003.
12. Khrouf, K., Soulé-Dupuy, C. : A Textual Warehouse Approach: A Web Data Repository. Intelligent Agents for Data Mining and Information Retrieval, Masoud Mohammadian (Eds.), Idea Publishing Group, pp. 101–124, 2004.
13. Malinowski, E., Zimányi, E.: Hierarchies in a multidimensional model: From conceptual modeling to logical representation. J. of Data & Knowledge Engineering (DKE), vol.59(2), Elsevier, pp.348–377, 2006.
14. Mass, Y., Mandelbrod, M.: Component Ranking and Automatic Query Refinement for XML Retireval. Advances in XML Information Retrieval, 3rd Int. Workshop of the Initiative for the Evaluation of XML Retrieval (INEX), LNCS 3493, Springer, pp.73–84, 2004.
15. McCabe, C., Lee, J., Chowdhury, A., Grossman D. A., Frieder O.: On the design and evaluation of a multi-dimensional approach to information retrieval. 23rd Int. ACM SIGIR Conf. on Research and Development in Information Retrieval, ACM, pp.363–365, 2000.
16. Mothe, J., Chrisment, C., Dousset, B., Alau, J.: DocCube: Multi-dimensional visualisation and exploration of large document sets. J. of the American Society for Information Science and Technology (JASIST), vol.54(7), pp. 650–659, 2003.

17. Nassis, V., Rajugan, R., Dillon, T.S., Wenny Rahayu, J.: Conceptual Design of XML Document Warehouses. 6th Int. Conf. *on* Data Warehousing and Knowledge Discovery (DaWaK), LNCS 3181, Springer, pp.1–14, 2004.
18. Park, B.K., Han, H., Song, I.Y.: XML-OLAP: A Multidimensional Analysis Framework for XML Warehouses. 7th Int. Conf. *on* Data Warehousing and Knowledge Discovery (DaWaK), LNCS 3589, Springer, pp.32–42, 2005.
19. Pérez, J.M., Berlanga-Llavori, R., Aramburu-Cabo M.J., Pedersen, T.B.: Contextualizing data warehouses with documents. Decision Support Systems (DSS), Elsevier, available online doi:10.1016/j.dss.2006.12.005, 2007 (in press).
20. Rafanelli, M.: Operators for Multidimensional Aggregate Data. Chapter 5 of Multidimensional Databases: Problems and Solutions, M. Rafanelli (ed.), Idea Group Inc., pp. 116–165, 2003.
21. Ravat, F., Teste, O., Tournier, R.: OLAP Aggregation Function for Textual Data Warehouse. 9th Int. Conf. on Enterprise Information Systems (ICEIS), INSTICC Press, pp. 151–156, June 2007.
22. Ravat, F., Teste, O., Tournier, R., Zurfluh, G.: Algebraic and graphic languages for OLAP manipulations. Int. j. of Data Warehousing and Mining (DWM), IDEA Group Publishing, 2007 (to appear).
23. Rizzi, S., Abelló, A., Lechtenbörger, J., Trujillo, J.: Research in data warehouse modeling and design: dead or alive? 9th ACM Int. Workshop on Data Warehousing and OLAP (DOLAP), ACM, pp. 3–10, 2006.
24. Sullivan, D.: Document Warehousing and Text Mining. Wiley John & Sons, 2001.
25. Torlone, R.: Conceptual Multidimensional Models. Chapter 3 in Multidimensional Databases: Problems and Solutions, M. Rafanelli (ed.), Idea Group Inc., pp. 69–90, 2003.
26. Tseng, F.S.C: Design of a multi-dimensional query expression for document warehouses. Information Sciences, vol.174(1-2), Elsevier, pp. 55–79, 2005.
27. Tseng, F.S.C., Chou, A.Y.H.: The concept of document warehousing for multi-dimensional modeling of textual-based business intelligence. J. of Decision Support Systems (DSS), vol.42(2), Elsevier, pp. 727–744, 2006.
28. Vrdoljak, B., Banek, M., Skočir, Z.: Integrating XML Sources into a Data Warehouse. 2nd Int. Workshop on Data Engineering Issues in E-Commerce and Services (DEEC), LNCS 4055, Springer, pp. 133–142, 2006.
29. Yin, X., Pedersen T.B.: Evaluating XML-extended OLAP queries based on a physical algebra. 7th Int. Workshop on Data Warehousing and OLAP (DOLAP), ACM, pp. 73–82, 2004.