SEMI-STRUCTURED INFORMATION WAREHOUSES
Requirements and Definition

Juan Manuel Pérez, Rafael Berlanga and María José Aramburu
Jaume I University, Castellón, Spain

Keywords: XML applications, semi-structured data analysis, document warehouses.

Abstract: During the last decade, data warehouse and OLAP techniques have helped companies to gather, organize and analyze the *structured* data they produce. Simultaneously, digital libraries have applied Information Retrieval mechanisms to query their repositories of *unstructured* documents. In this context, the emergence of XML means the convergence of these two approaches, making possible the development of warehouses for *semi-structured* information. Although there exist several extensions of traditional data warehouse technology to manage semi-structured information, none of them are based on an underlying document model able to exploit this kind of information. Along this paper we expose our vision of what a semi-structured information warehouse should be, by identifying a set of requirements throughout an example scenario.

1 INTRODUCTION

During the last decade, data warehouse (Kimball, 2002) and OLAP (Codd et al, 1993) techniques have helped companies to gather, organize and analyze their *structured* data (usually stored in their own enterprise's databases) to support decisions at various levels. These organizations also produce huge amounts of *unstructured* documents such as emails, spread sheets or word processing documents. At the same time, the Web has become the largest source of companies external information. Unfortunately, although all these documents contain highly valuable information, current data warehouse technology cannot be applied to them.

The ever increasing amount of information published on the Internet has provided us with new services like digital libraries. All these applications require of novel techniques to store and manage huge amounts of *unstructured* information. Most solutions to query these repositories are based on Information Retrieval (IR) (Baeza-Yates and Ribeiro-Neto, 1999) techniques. More recently, the efforts are focused on the definition of architectures for the integration of distributed and heterogeneous documents sources.

From our point of view, XML is a means of convergence for the warehouse and document retrieval research areas, and opens a novel and interesting range of possibilities to exploit semi-structured information.

The acceptance of XML as the standard for semi-structured data exchange over the Web, points out to a close future when information on the Internet will be published as XML documents, and exportation tools from most proprietary systems to XML-like formats will be available. Furthermore, the current demand of new architectures for the integration of distributed information, together with the already proven qualities of the data warehouse and OLAP techniques for the analysis and exploitation of large data repositories, make very attractive the idea of extending data warehouses with more flexible data models, able to incorporate XML documents: *semi-structured information warehouses*.

In the recent literature some proposals start analyzing the problem of extending current data warehouse technology to manage semi-structured information. Section 2 presents some works done in the field of semi-structured data models and XML warehouses. However, to date none of these approaches entirely exploit all the properties of these documents. At this work we present our particular interpretation of what a warehouse for semi-structured information should be. In Section 3 we present an example scenario for the development of semi-structured information warehouses and identify a set of new requirements for this novel technology. Finally, we expose conclusions and future research at Section 4.

Manuel Pérez J., Berlanga R. and José Aramburu M. (2004).
SEMI-STRUCTURED INFORMATION WAREHOUSES - Requirements and Definition.
In *Proceedings of the Sixth International Conference on Enterprise Information Systems*, pages 579-582
DOI: 10.5220/0002633705790582

2 RELATED WORK

Several models have been proposed to store and retrieve XML documents, like (Navarro and Baeza-Yates, 1997) and (Xyleme, 2001). These approaches combine IR techniques with mechanisms for the evaluation of conditions over the structure of the documents. In the same line, TOODOR (Aramburu and Berlanga, 2001) is a storage and retrieval model for structured documents which additionally considers their temporal dimensions.

More recently, in the scope of data warehouses, some works which start managing semi-structured information have been presented. They can be classified as follows.

A first group of works (Binh et al, 2001) (Mangisengi et al, 2001) are oriented towards the integration of data warehouses. These systems use XML languages to represent metadata over the data sources, or as canonical languages when transferring data between their components.

A second group of works (Xyleme, 2001) (Ishikawa et al, 1999) are focused on the definition of architectures for document or semi-structured data warehouses. Although, these proposals specify techniques for the collection and massive storage of semi-structured data, they do not include any high level analysis tools able to exploit this information.

Finally, a quite different approach is (Pedersen et al, 2002). There, a new query language based on SQL and XPath allows the execution of OLAP operations that involve data contained in external XML documents. In this case, the semantics of aggregation operations is meticulously revised, but the highly structured philosophy of the traditional data warehouses remains in the model. That is, OLAP operations include data coming from semi-structured documents, but this information is managed as structured data once inside the warehouse.

Summarizing, there exist different proposals to enrich current data warehouse technology with XML information. However, in our opinion, to date none of these systems is able to entirely exploit the semi-structured nature of these kind of documents. In next section we expose our interpretation of semi-structured information warehouses, by pointing out a set of requirements throughout an example scenario.

3 AN EXAMPLE SCENARIO

In this section an example scenario and a set of analysis queries is used to explain the special requirements of a warehouse for semi-structured information. The language used at the example queries is an extension of TDRL (Aramburu and Berlanga, 2001) with XPath expressions and the OLAP operators of SQL-99.

We will consider that our warehouse stores a collection of XML digital news extracted from various Internet sources. Figure 1 presents a document of this repository with a news item about the disasters caused by a storm. The objective of our analysis is to study the weather conditions that caused the natural disasters described at the relevant documents.

Traditional data warehouses operate over a set of highly structured facts. These tuples contain attributes with the measures of study and the dimensions to analyze. However, in a warehouse for semi-structured information the facts are not so highly structured as they take part of the textual contents of XML documents. Thus, traditional data warehouse techniques cannot be directly applied to them.

As shown in Figure 1, the labels **LOCATION**, **DATE**, **RAINFALL** and **TEMPERATURE** contain values that can be considered as either measures of analysis, or values of the corresponding dimensions.

```
<NEWSPAPER NAME=”El País” PUBLICATION_DATE=”Tuesday, 2nd July 2002”> ...
<LOCAL_SECTION>
<NEWS_ITEM> <AUTHOR>Carlos García</AUTHOR>
<TITLE>
<LOCATION XW:VALUE=”/Europe/Spain/Valencia”>Valencia</LOCATION> suffers the biggest storm of July of the last
41 years
</TITLE>
<SUBTITLE>
Two bathers die drowned at the beaches of <LOCATION XW:VALUE=”/Europe/Spain/Menorca”>Menorca</LOCATION> and
<LOCATION XW:VALUE=”/Europe/Spain/Formentera”>Formentera</LOCATION>
</SUBTITLE>
<PARAGRAPH>
The biggest storm of July of the last 41 years fell<DATE XW:VALUE=”/2002/07/01”>yesterday</DATE> night over
the city of <LOCATION XW:VALUE=”/Europe/Spain/Valencia/Valencia”> Valencia</LOCATION>. The
<RAINFALL XW:VALUE=”128” XW:UNIT=”l/m2”>128 liters per square meter</RAINFALL> rained in only 24 hours made
firemen had to go on rescue more than 100 times. At <LOCATION XW:VALUE=”/Europe/Spain/Valencia/Burjassot”>
Burjassot</LOCATION> fell <RAINFALL XW:VALUE=”132” XW:UNIT=”l/m2”> 132 liters per square meter</RAINFALL>
throwing down a building. The strong rain on the East of Iberian Peninsula caused disasters in the regions of
<LOCATION XV:VALUE=”/Europe/Spain/Murcia”>Murcia</LOCATION> and <LOCATION XW:VALUE=”/Europe/Spain/Baleares”>
Baleares</LOCATION>.
</PARAGRAPH> ...
</NEWS_ITEM> ...
</LOCAL_SECTION> ...
```

Figure 1: A piece of a document of the warehouse.

These labels could appear in the original documents or, alternatively, they can be inserted by applying information extraction or shallow parsing techniques.

3.1 Documents structure and conceptual analysis schema

As XML documents are self-describing, part of the conceptual analysis schema (dimensions and measure values) is implicitly represented in their own structure (or in their associated XML Schemas). Notice how the elements in bold of the example document of Figure 1 can be used as a dimension or as a measure. In this way, when managing semi-structured information, path expressions are a natural way of specifying the dimensions and measures involved in analysis queries. Thus, it could be possible to design warehouse architectures where the analysis schema would only define the dimension hierarchies, and where the measures and dimensions to study could be directly specified in the analysis queries.

The example query below shows how XPath expressions can be used to indicate the dimension and measures under analysis:

```
SELECT Avg(Paragraph/Rainfall)
FROM //Local_Section//News_Item
GROUP BY CUBE (Paragraph/Location)
```

3.2 Fact relevance

The previous query computes the average of the values of the measure `Rainfall` for the dimension value `Location`, that is, it returns the average of the amount of water collected per location. The facts considered by this query are those contained in the paragraphs of the news items stored in the warehouse. Notice that in the document of Figure 1 there exist paragraphs which describe more than one fact. Conversely, there could also occur different paragraphs of the same news item describing the same `Rainfall-Location` fact. The reason for these repetitions is the high relevance of the fact with respect to the news main subject.

Like in IR systems, where query results are ranked with a relevance index, a measure of fact relevance must be introduced in semi-structured information warehouse models. In this way, the most relevant facts of news items could receive more consideration in the evaluation process than the other less relevant facts. Notice that the levels of a dimension hierarchy will affect to the relevance of the facts for a given news item. For example, considering the levels of the `Location` dimension, facts that are different at lower levels of the hierarchy could became the same fact at higher levels.

It is concluded that the semantics of the aggregation operations in semi-structured information warehouses must be carefully revised, not only to manage the facts relevance, but also to consider those facts that appear without some of their measures or dimensions.

3.3 The structure as an implicit dimension

In this section we explain how the document structure can be considered as an implicit dimension when analyzing semi-structured information. For example, next query builds a cube for the average of temperature values but without selecting any dimension.

```
SELECT Avg(//Temperature)
FROM //Local_section//News_Item
GROUP BY CUBE
```

Like in the example of Figure 1, each `News_Item` element in the warehouse describes a set of facts. Thus, this query would return a temperature average for each news item of the `Local_section` elements stored in the warehouse. By going up a level in the document structure hierarchy we would obtain the same measure for each local section. This process could be repeated several times until considering complete documents.

Consequently, by using the structure of documents as an implicit dimension, it is possible to construct OLAP cubes to analyze the facts at different levels of detail. Notice that the elements at higher levels of the structure use to group more occurrences of the same fact, implying that the relevance of the facts is also affected by the structure dimension.

3.4 OLAP queries with IR conditions

In order to complete our analysis of the general requirements of a warehouse for semi-structured information, in this section we explain the importance of specifying IR conditions in OLAP queries. Notice that although our objective is to study the weather conditions that caused the natural disasters, previous example queries involved all the news items stored in the warehouse.

The next query shows how to restrict our analysis to those news items that in their title contain the term “storm” with a relevance index greater than 50%. When evaluating an aggregation, the degree of contribution of the facts described in a news item must be proportional to the relevance of this news item for the studied topic.

```
SELECT Avg(Paragraph/Rainfall)
FROM //Local_Section//News_Item
WHERE Title contains ‘storm’ > 0.5
GROUP BY CUBE (Paragraph/Location)
```

3.5 IR terms as a dimension

Finally, next query shows how the terms specified at IR expressions can be used as an additional analysis dimension. In this context, thesaurus and ontologies would allow defining classification hierarchies over this new dimension. In this way, the query below provides a new dimension to study the aggregation of those news that contain the term “tidal wave”, or the aggregation for the term “flood”. Note that these two aggregations can be joined at higher levels of the term hierarchy (e.g. “natural disaster”).

```
SELECT Avg(Paragraph/Rainfall)
FROM //Local_Section//News_Item
WHERE Title contains
      ‘tidal wave| flood’ > 0.5
GROUP BY CUBE
     (‘tidal wave| flood’,
       Paragraph/Location)
```

4 CONCLUSIONS

In this paper we have explained how the data warehouses and digital libraries communities, each from its particular point of view (*structured* vs. *unstructured* information, respectively), can mutually take advantage from *semi-structured* information. Recently, some proposals start extending the traditional data warehouse technology towards semi-structured information. However, to date none of these approaches entirely exploit all the properties of these documents. Along this work we have identified a set of requirements for this novel technology, the semi-structured information warehouses technology.

In our opinion, the development of such systems must be based on an underlying document model able to exploit the nature of this kind of information. We are currently working on a semi-structured model which combines IR and evaluation of structural conditions techniques to query an XML documents collection, and where the facts described at the selected documents are ranked by relevance.

For the future, we plan to design a semi-structured warehouse model built over this document model. In order to involve the facts relevance in the warehouse model, the semantics of the aggregation operations will have to be revised. In this context, we find interesting some works which study the management of imprecise information (Pedersen et al, 1999) (Rundensteiner et al, 1992).

REFERENCES

Kimball, R., 2002. The Data Warehouse toolkit. John Wiley & Sons.

Codd, E. F.; Codd, S. B. and Salley, C.T., 1993. Providing OLAP to user-analysts: An IT mandate. Technical Report, E.F. Codd & Associates.

Baeza-Yates, R. and Ribeiro-Neto, B., 1999. Modern Information Retrieval. Addison-Wesley.

Navarro, G. and Baeza-Yates, R., 1997. Proximal Nodes: A Model to Query Document Databases by Contents and Structure. *ACM Trans. on Information Systems*.

Xyleme, L., 2001. A dynamic warehouse for XML data of the Web. *IEEE Data Engineering Bulletin* 24(2).

Aramburu, M. J. and Berlanga, R., 2001. A Temporal Object-Oriented Model for Digital Libraries of Documents. *Concurrency: Practice and Experience* 13 (11), John Wiley.

Binh, N. T.; Tjoa, A. M. and Mangisengi, O., 2001. Meta Cube-X: An XML Metadata Foundation for Interoperability Search among Web Warehouses. *Proc. Intl. Workshop on Design and Management of Data Warehouses*.

Mangisengi, O.; Huber, J.; Hawel, C. and Essmayr, W., 2001. A Framework for Supporting Interoperability of Data Warehouse Islands Using XML. *Proc. of the 3rd Intl. Conference on Data Warehousing and Knowledge Discovery*. LNCS 2114.

Ishikawa, H. et al, 1999. Document Warehousing Based on a Multimedia Database System. *Proc. IEEE 15th Intl. Conference on Data Engineering*, pp. 168-173.

Pedersen, D.; Riis, K. and Pedersen, T. B., 2002. XML-Extended OLAP Querying. Technical Report, Department of Conputer Science, Aalborg University.

Pedersen, T. B.; Jensen, C. S. and Dyreson, C. E., 1999. Supporting Imprecision in Multidimensional Databases Using Granularities. *Proc. of the Eleventh International Conference on Scientific and Statistical Database Management*, pp. 90–101.

Rundensteiner, E. and Bic., L., 1992 Evaluating Aggregates in Possibilistic Relational Databases. DKE, 7(3):239–267.