Journal of Southeast University (English Edition) Dec. 2001 Vol.17 No.2 ISSN 1003—7985

XML Based Data Cube and X-OLAP

Wang Xiaoling* Dong Yisheng

(Department of Computer Science and Engineering, Southeast University, Nanjing 210096, China)

Abstract: Data warehouse provides storage and management for mass data, but data schema evolves with time on. When data schema is changed, added or deleted, the data in data warehouse must comply with the changed data schema, so data warehouse must be re-organized or re-constructed, but this process is exhausting and wasteful. In order to cope with these problems, this paper develops an approach to model data cube with XML, which emerges as a universal format for data exchange on the Web and which can make data warehouse flexible and scalable. This paper also extends OLAP algebra for XML based data cube, which is called X-OLAP.

Key words: data warehouse, data cube, XML, X-OLAP, semi-structured data

Data warehousing[1] technologies are emerging as key technologies to improve data analysis, decision support activities and automatic extraction of knowledge from data. Much progress in these new fields has been witnessed[2]. However, most data from application systems or web is irregular and incomplete, called semi-structure data[3]. The term semi-structured data has been used to refer to data which is irregular or which exhibits type and structural heterogeneity semi-structured, since it may not conform to a rigid, predefined schema. Because there is no uniform schema for those semi-structured data, how to clean and integrate heterogeneous data and how to organize semi-structured data are challenging[4]. In traditional database systems, it is exhausting to modify the schema, which requires data to be migrated wholly to the new structure and requires application code to be rewritten. This paper presents a method to cope with this evolution data, and this method can avoid data warehouse re-organization or re-construction.

The extensible markup language (XML) is believed to become a universal format for data exchange on the Web and in the near future we will find vast amounts of documents in XML format on the Web. Because XML can be used to model many data models, such as tree model and graph model, in this paper, we combine XML graph data model and OLAP operations in data warehouse in order to make the data warehouse model more efficient and effective. We present an approach to map from semi-structure data model to snowflake schema for data in data warehouse, in order to help users to locate and store evolving semi-structured data in data warehouse.

The rest sections of this paper are organized as follows. Section 1 introduces related work by others in this field; Section 2 gives the system architecture; Section 3 presents mapping method and X-OLAP operations on XML-based data cube in data warehouse. Conclusions are presented in the last section.

1 Related Work

In recent years, some researchers have done much research in these fields, which includes how to model and query semi-structured data and how to extract and integrate information from the heterogeneous data sources into data warehouse. Many prototypes and commercial systems for data mining, information integration, and data warehousing have been developed, including Lore[3] and WebBase[5]. Most of those web models are graph-based or tree-based, among which OEM[3] is the most typical, which is schema-less, self-description model, presented by Stanford University. The vertices in the OEM graph are objects, each object has a unique object identifier (OID). The edges are attributes of the objects. Though there are too much data models for semi-structured data, it is no standard for exchange among them, so it is difficult to communicate between these systems. With the emerging of XML as a standard for data exchange and representation in the web, there are more semi-structured data modeling with XML. XML is a more generic language for modeling semi-structured data than OEM model. This paper takes advantage of XML for data model in data warehouse.

Received 2001 - 08 - 23.

* Born in 1975, female, graduate.

2 XML Based Data Warehouse

Traditional data warehouse is a subject-oriented, integrated, time-variant and nonvolatile data base. The point is that many data are evolving and we must be very smart in managing these changing data, i.e., semi-structured data. XML based data warehouse also includes six main modules as the traditional data warehouse: data storage, meta data, wrapper, data integrator, data modeling and application tools[6].

- Data warehouse storage: Two main physical models for data warehouse are relational database and multi-array data storage. How to map logical model to physical model isn't focused in this paper.
- Meta data: It stores all meta information of data in the data warehouse. Those meta data include data source descriptions, data format, data model, data location, the structure of tables, data cleaning rules, and so on. The main distinguishing features for XML based data warehouse is that these are XML schema related to XML data in data warehouse. Following these XML schemas, OLAP on XML based data cube is executed.
- Wrappers: A uniform interface for data integrator is provided.
- Data integrator: It is responsible for putting data from wrappers into the data warehouse.
- Data warehouse modeling tool: It provides a series of tools for concept modeling, logical modeling and physical modeling. Next section will focus on data model in data warehouse.
- Application tools: There are OLAP tools, mining tools, report tools and other analysis tools for making decisions and knowledge discovery[7].

In the field of data warehouse, former work on data model in data warehouse has two main logical models: star model and snowflake model. In this paper, data cube is represented by snowflake model and we present an approach to map between snowflake model and XML. Online analysis and process (OLAP) operations on data cube are defined on these models. These OLAP operations are used for analysis and query on the data cubes for data warehouse users.

3 XML Based Data Cube

This section reveals the similarity between semi-structured data in XML and data cube, and then presents X-OLAP operations on semi-structured data in XML. There is a natural link between data cube and XML graph, and they are all multi-dimensional and hierarchy structure. The more detailed, the lower granularity data is; the more summarized, the higher granularity data is.

3.1 Data cube in data warehouse

Because data warehouse focuses on OLAP operations, such as roll-up, drill-down, slice and rotation, how to represent data with data cube is the main task for modeling data warehouse. There are two kinds of logical model: snowflake model and star model, both of which are composed of fact table and dimensional tables. To some extent, the fact table is the center of this entity, and the dimensional tables are some attributes or level of this entity. In this paper, we use snowflake schema to model data cube, and by the dimension identifications to build a relation between fact table and dimension table, or between dimension tables. An example about data cube on sale analysis is shown in Fig.1.

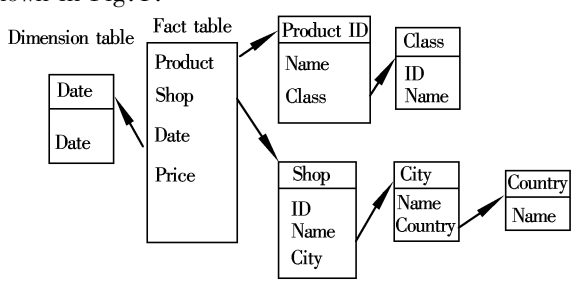

Fig.1 Snowflake model in data warehouse

For every topic, in data warehouse, there is a corresponding data cube, which reveals a level structure about this topic, such as, sale → product → class, sale → region → city → country (from detail to summary).

3.2 XML data

XML is a textual language, which is very suitable for web application, including more structures, more semantics, extensibility, simplicity, self-describing, validation, separation between data and its representation, and so on. XML is presented by W3C, and it is a standard language for data transformation and exchange on the web. There are more and more data represented by XML on the web or in some applicable systems.

When modeling data in XML, data representation can be viewed as a tree and query can be treated as traversing tree or graph. Nested, tagged elements are the building blocks of XML. Each tagged elements has a sequence of zero or more attribute/value pairs, and a sequence of zero or more sub-elements. XML data has

been modeled as a directed acyclic graph (in most cases trees)[8]. An example about data in XML is shown in Fig.2.

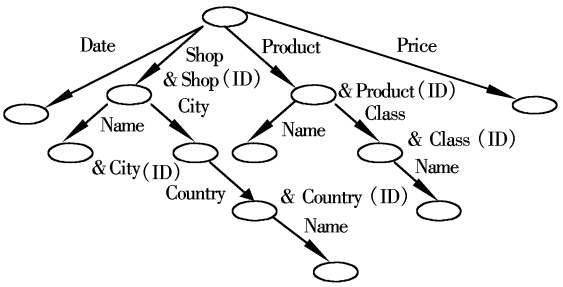

Fig.2 XML data model

XML schema is emerged as a standard to describe the schema of XML data. It describes XML data type and XML data structure. For example, XML schema about "product" in Fig.2 is as follows.

```
〈complexType name = productType 〉
  〈sequence〉
      〈element name = "ID" type = "int"/ 〉
      〈element name = "name" type = "string"/〉
      〈 element name = "class" type = "classType"/ 〉
〈/sequence〉
〈/complex Type〉
〈complexType name = classType 〉
  〈sequence〉
      〈element name = "ID" type = "int"/ 〉
      〈element name = "name" type = "string"/〉
〈/sequence〉
〈/complex Type〉
```

One XML schema can be correlative to one or more XML data, that is, some data documents has a schema. At the same time, every XML data document can be interpreted by different XML schema files, so there are different semantics for one XML data file when it attached with different XML schema files. The following is a part of XML data related XML schema in Fig.2.

```
〈data〉 2001-5-1 〈/data〉
〈product ID = "A1"〉
      〈name〉 book 〈/name〉
      〈class ID = "1-002"〉
          〈name〉 novel 〈/name〉
      〈/class〉
〈/product〉
```

3.3 XML based data cube

XML has become a uniform language to model data from web and other data sources into the data warehouse. Because of its good features, we combine XML tree data model and OLAP operations in data warehouse in order to provide more efficient and effective models. Data in XML is represented as a graph-based data model, and there are path operations on it, in which there is a unique root for a model, and there are some hierarchical level structures. Thus, there is a unique entry for data cube and XML data model, and they all have hierarchical structures to describe the entity content.

Except the same structure above, there are some similar operations on them. For data cube, there are some OLAP (on-line analysis and process) operations on data cube, including slice, rotation, roll-up, drill-down and so on. By these operations, users can get any aspect of this entity. And, for graph operations, there are path operations and navigate operations, by invoking these operation, users can get any nodes in the graphs. By the approach we present in this paper, all OLAP operations on data cube can be gotten by X-OLAP operations on XML based data cube.

So, from the structure and the operations, we can see that the similarities between the snowflake schema and XML graph. Our work includus: firstly, we use graph model to represent data in data warehouses; secondly, use OLAP and navigate operations to process data; thirdly, describe meta data or data schema with XML schema.

3.4 OLAP operations on XML based data cube

Most decision analysis and query process in data warehouse are implemented by OLAP. OLAP defines some statistical operations on data cube, such as roll-up, drill-down, slice, rotation and so on. These operations get useful data from SQL aggregation functions: SUM, COUNT, MAX, MIN, AVG. In this section, we present OLAP operations on XML data cube, and we also show that this operation calculus is completeness and closure[5].

Definition 1 XML document is an XML instance of an XML schema. An XML data can be modeled as a grammar represented by a triple t = (node, label, operation), where node is an object with OID, label is attributes of objects and operations are accessing or navigating operations on the XML tree or graph.

Definition 2 XML based data warehouse is a finite set of documents (or data cubes) conforming to one of the XML schema definitions in meta data.

OLAP operations on XML objects are called X-OLAP algebra. This paper shows that OLAP operations on XML-based data cube are equal to traditional OLAP operations on data cube, and that algebra is completeness and closure, we will prove it in this section.

Definition 3 In data cube, dependent relation is

defined: $\langle a_1, a_2, \cdots, a_n \rangle \leqslant \langle b_1, b_2, \cdots, b_n \rangle$ iff $a_i \leqslant b_i, 1 \leqslant i \leqslant n$.

For example, there is a dependent in Fig.2: country $\leqslant$ city; $\langle$class, country$\rangle \leqslant \langle$product, city$\rangle$.

Definition 4 In XML graph, dependent relation is defined: $\langle a_1, a_2, \cdots, a_n \rangle \leqslant \langle b_1, b_2, \cdots, b_n \rangle$ iff node a_i is the offspring of node b_i in graph, $1 \leqslant i \leqslant n$.

For example, node "country" is a child of node "city" and node "class" is a child of node "product", so $\langle$class, country$\rangle \leqslant \langle$product, city$\rangle$.

Rule 1 Any dependent relation in snowflake schema can be mapped to dependent relation in XML graph model.

Proof Because of the structural mapping from snowflake schema to graph model, there are the same hierarchical structure among them. So, it is obvious that dependent relation in snowflake schema is equal to ancient-offspring in XML graph.

Definition 5 In OLAP, distributive aggregate function is defined as follows: S is a set of value of any variable. $S = S_1 \bigcup S_2 \bigcup \cdots \bigcup S_n$ and $S_1 \bigcap S_2 \bigcap \cdots \bigcap S_n = \varnothing$. If $\Psi(S) = \Psi(\Psi(S_1) + \Psi(S_2) + \cdots + \Psi(S_n))$, then Ψ is distributive aggregate function.

There are four aggregation functions, such as SUM, MAX, MIN and COUNT, which are all conform to the above definitions.

Definition 6 In OLAP, algebraic aggregate function is defined: the aggregate function results from any other aggregate function by algebraic operation. For example, AVG(S) = SUM(S)/COUNT(S).

Rule 2 Aggregate functions in OLAP can be mapped into functions on XML graph model.

Proof Any aggregate $\Psi(S)$, where Ψ is distributive aggregate function, can be got by SUM, MIN, MAX or COUNT, those operations can be got by operation on the attribute S according to the definition 5 and 6. So, aggregate functions of X-OLAP in XML graph model can be got by algebraic functions in OLAP.

Rule 3 Any operations in OLAP, such as roll-up and drill-down, can be transformed into operations in X-OLAP.

Proof X-OLAP is for graph model, so we need to prove that OLAP functions can be got by operations on XML graph. Roll-up operation can be regarded as path travel from root node to leaves node. Drill-down can be regarded as from leaves nodes to root nodes. Slice is an operation to get a sub-tree of a node in XML tree; rotation is a change of order among sub-trees about a node in the tree or graph.

From above, there is a mapping from snowflake schema to XML tree or graph. With the help of XML schema, OLAP operations can be executed on this data model. Except traditional OLAP operations, path operations and navigate operations on DAG are also executed on XML-based data cube.

Theorem 1 Completeness of X-OLAP algebra.

Proof All traditional OLAP operations can be mapped into XML tree operations as we show in this section.

Theorem 2 Closure of X-OLAP algebra.

Proof Closure means that XML is to be used as the input as well as the output for queries. Closure is important because it allows nesting of queries and also simplifies the implementation of the input-output handling of this language. As we show above, all operations are closed under this model, in the sense that they all take valid XML documents as input and valid XML as output.

According to the above definitions, OLAP operations can be executed on XML-based data cube. Because any XML data has an XML schema, which is helpful to describe the XML data and which is a directive to execute OLAP operations, XML based data cube is feasible to cope with semi-structured and evolving data.

4 Conclusion

In recent years much progress about XML applications and data warehouse solutions are viewed. This paper develops an approach to map from data in data cube to data in XML, and which is important for web applications and evolution data in data warehouse. We present how to model data with XML and give X-OLAP operations on XML based data cube. XML is used to describe any data from structure data to semi-structured data, and there is much advantage to use XML to model data in data warehouse. It is easier to put data to data warehouse and it is feasible to cope evolution data warehouse with XML. At the same time, it is a feasible method to get deduced schemas from the data schema in data warehouse, so it is easier to get different views for specific users.

Now we focus on how to implement operations of X-OLAP on the XML based data model presented in this paper. In the future work, we will do some research on indexing, data mining on XML based data cube and query language on the XML based data

warehouse and X-OLAP.

References

1 T. Palpanas, Knowledge Discovery in data warehouses, *SIGMOD Record*, vol.29, no.3, pp.88 – 100, 2000

2 P.Bernstein, M. Brodie, S. Ceri, and J. Ullman, et al. *The asilomar report on database research*, http://cs-tr.cs.cornell.edu:80/Dienst/UI/1.0/Display/xxx.cs.DB/9811013, September 1998

3 J. McHugh, S. Abiteboul, R. Goldman, D. Quass, and J. Widom, Lore: A database management system for semistructured data, *SIGMOD Record*, vol.26, no.3, pp.54 – 66, September 1997

4 R.D. Hackathorn, *Web farming for the data warehouse*, Morgan Kaufmann Publishers, Inc. San Francisco, California, 1999

5 J. Hirai, S. Raghavan, H. Garcia-Molina, and A. Paepcke, WebBase: *A repository of web pages*, www9, Amsterdam, 2000

6 X. L. Wang, and Y. S. Dong, RDB-based data warehouse system design and implement, *AMSMA*'2000, GuangZhou, China, October 2000

7 N. Sundaresan, and J. Yi, *Mining the Web for Relations*. www9, Amsterdam, 2000

8 N. B. Wang, *Database management system*(in Chinese), Publishing House of electronics Industry, 2000

基于 XML 数据立方的面向对象扩展

王晓玲　　董逸生

(东南大学计算机科学与工程系, 南京 210096)

摘　要　数据仓库为海量数据提供了一种有效的存储管理和查询分析的数据平台,随着应用需要和万维网的发展应用,数据仓库中的数据模式会随着时间而发生变化,需要对数据仓库重组或者重建,会耗费大量的人力物力.本文提出了基于 XML 的数据立方数据模型.通过对数据仓库技术、面向对象技术和 XML 技术的结合的探讨,提出了基于 XML 的数据立方以及定义在这种数据模型上的 OLAP 操作,从而为基于 Web 数据仓库的应用提供了一种新的表示和实现方法,解决了数据仓库中模式演化所带来的重组问题,保证了数据仓库系统的稳定性、灵活性和可扩展性,适应了新一代 Web 应用的需要.

关键词　数据仓库, 数据立方, XML, X-OLAP, 半结构化数据

中图分类号　TP39