XML 数据库管理系统研究*)

Research on XML DataBase Management System

王照岳　孙建伶　董金祥
(浙江大学人工智能研究所　杭州310027)

Abstract XML is not only an intermedia between the WEB and DBMS. W3C has defined many database characters for XML, such as schema, algebra, etc, so we believe that XML will become a new data model of database, and it is necessary to apply traditional DBMS techniques to build a XML DBMS. In this paper we show the necessity of building a XML DBMS and present the architecture of it, based on natural XML data model. Then we describe some key techniques of XML DBMS, including data storage, query optimization and visual user interface.

Keywords XML DBMS, XML Storage, Query Optimization, Visual User Interface

1 引　言

XML 是连接互联网和数据库的桥梁[1]，但 XML 不仅仅是互联网和数据库之间的中介。首先，XML 数据是典型的半结构化数据[2]，其表达能力要强于关系模型和对象模型；其次，XML 数据有自己的特点，简单的应用传统的数据库技术不能完全体现其特点；第三，XML 主要应用在互联网上的信息交换，而传统的数据库技术对这类应用并不擅长；第四，W3C 在 XML 工作文本[3,4]中为其定义了很多数据库特征，这使得 XML 很有可能成为和关系模型、对象模型并列的新的数据模型。因此，有必要将传统的数据库技术移植到 XML 上，建立一个基于 XML 的数据库管理系统(XML DBMS)。

由于 XML 与关系模型、对象模型的差异，要在 XML 中应用传统数据库技术并不是一件简单的事，还有很多问题有待解决[5]。这其中，最核心的是如何表达 XML 数据。目前这方面已经有了很多研究，也出现了很多 XML 数据库产品和原型系统，如 Lore[6]，Tamino[7]，POET[8]，eXcelon[9] 和 STORED[10] 等。W3C 在 XML Schema 推荐方案[3]中定义 XML 的模式，这不仅使得 XML 数据模型到其他模型的转换更加简单，而且更重要的是 XML 数据的结构性大大增强了。XML 查询语言也是一个研究的热点，目前已经出现了很多的原型语言，W3C 最近也推出了 XML 查询工作草案[4,11,12]，对 XML 查询语言的各方面都做了详尽的规定。

本文的余下部分从实现的角度出发，综述了 XML DBMS 的体系结构和一些关键技术，包括数据存储、查询处理和可视化用户界面。

2 XML DBMS 体系结构

XML DBMS 主要用于互联网上的信息发布和数据交换，其中最主要的数据访问方式就是查询，因此要求查询处理效率高；事务管理有效，能应付短时间内的大量只读事务。在互联网上，用户都是通过浏览器访问数据，这就要求 XML DBMS 有一个友好的可视化用户界面。对 XML 数据的访问主要是计算机程序发出的，因此需要一个表达能力强而且便于程序自动生成的 XML 查询语言。另外，Web 上的数据一般是异构的，因此 XML DBMS 应支持对异构数据源的透明访问。这些都和传统 DBMS 有所不同。

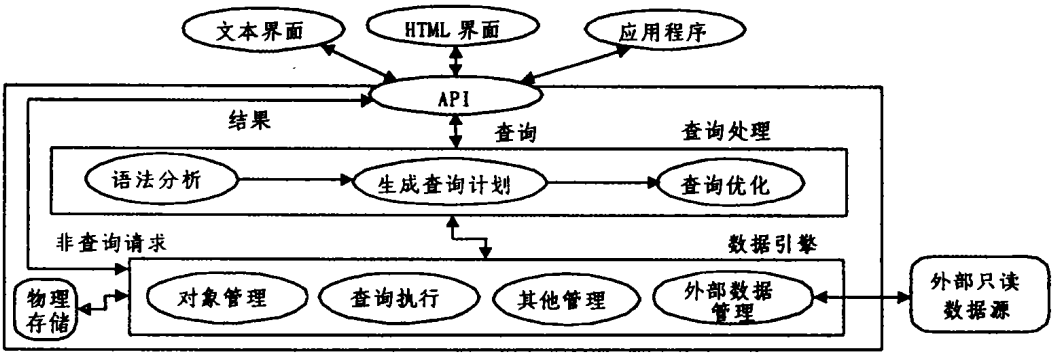

图1

基于传统数据模型的 XML DBMS 的体系结构比较简单，就是在传统 DBMS 上增加一个数据模型和查询语言转换层，而直接使用它的底层。这里主要介绍基于 XML 数据模型的体系结构。这个体系结构参照了 Lore 系统[6]。从图1中可知，XML DBMS 分为4层：访问界面、查询处理、数据引擎和物理存储。

*)航天工业总公司国防预研基金(项目编号：45.6.1-015)资助项目。王照岳　硕士研究生，主要研究方向为 Web 数据管理。孙建伶　副教授，主要研究方向为对象数据库，Web 数据管理等。董金祥　教授，博导，主要研究方向为数据库、CAD、CAM、CIMS 等。

XML DBMS 支持三种访问界面：用户可以通过文本或浏览器浏览和查询 XML 数据；应用程序通过调用 API 函数来访问 XML DBMS。所有这三种界面都是通过调用系统提供的 API 接口来实现的。

XML DBMS 的查询处理过程与传统数据库类似，首先对查询语句进行语法分析，再根据语法分析结果生成查询计划，根据查询计划进行优化，得到最优的执行计划，最后执行计划，并将结果返回给请求者。查询执行由于要直接操作 XML 数据，因此划分到数据引擎中。

数据引擎执行系统的数据管理功能。对象管理负责所有数据（XML 数据、DTD、索引等）的管理。对象管理提供事务级的并发控制和恢复功能。查询执行提供了查询的物理操作功能。外部数据管理提供了对外部数据源的透明访问。数据引擎的其他管理功能包括索引管理，查询辅助信息管理等。

XML 数据和索引，DTD 等都以文件的形式存储在外存。出于优化性能考虑，这些数据还可能要进行聚簇。

3 XML 数据的存储

3.1 XML 数据的存储方式

存储 XML 数据的方式很多[13]：1）直接存储 XML 文档，即将 XML 文档作为大对象存储在数据库中；2）将 XML 数据模型映射到其他模型，如关系模型、面向对象模型和 OEM 模型等，每项 XML 数据都作为一个记录存储；3）同时采用两种方法，即存储 XML 文档的同时，将数据从文档中抽取出来，按记录存储；4）综合方式，即对 XML 数据按一定的方法分成比较大的块，以块为单位存储数据。第一种方法最简单，但隐藏了 XML 数据，操作数据之前要对文档进行预处理。第二种方法充分利用传统 DBMS 存储管理技术，但重新生成 XML 文档的代价很大，一些模型映射算法十分复杂[14]，甚至需要消除重复数据[10]。第三种方法兼有前两者的优点，但更新操作比较复杂，需要保持两种存储方式之间的数据一致性。第四种方法简化了数据存储，同时又不需要复杂的数据模型转化算法，但不能有效利用块内数据的结构信息。现在 W3C 的工作文本中已经为 XML 定义了多种数据模型，如 DOM 模型[15]、查询数据模型[12]等。基于这些模型的持久化存储 XML 数据，不但表达上自然，而且能方便地转化到 DOM 模型或查询数据模型，这对 XML 数据的处理十分有利。

3.2 XML 文档的 IMPORT/EXPORT

在互联网上，数据交换都是以 XML 文档的形式进行的，因此 XML DBMS 必须提供 XML 文档的 IMPORT/EXPORT 功能。如果直接存储 XML 文档，就不存在 IMPORT/EXPORT 的问题。将 XML 数据转化为其他模型存储可以充分利用现有 DBMS 成熟的存储管理技术，所以 XML 文档的 IMPORT/EXPORT 的关键就在于模型的转化上。文[14]提出了一种利用 DTD 把 XML 数据转化到关系模型的方法。STORED[10]的方法有所不同，它先为半结构化数据生成一个关系模式，利用这个关系模式来存储半结构化数据，有一些半结构化数据无法映射，就保留在一个溢出图（overflow graph）中。STORED 同样适用于 XML 数据。XML 具有和对象模型类似的层次结构，因此转化到对象模型要比转化到关系模型直观，而且能自然地表达引用和路径表达式。XML 数据是典型的半结构化数据，因此将其转化到 OEM 模型相比之下要简单得多，文[2]介绍了具体的转化方法。

以上几种方法都存在问题：关系模型很难表达 XML 数据的语义，且模型映射算法复杂；将 XML 数据转化到关系模型或对象模型会产生大量的小关系或小对象，EXPORT XML 文档的代价很大[13]；而 OEM 模型和关系模型、对象模型还有一个共同的缺点，它们很难表达 XML 数据间的次序。

XML 文档 IMPORT 操作过程如下：首先，分析文档，从中抽取出数据；然后进行模型转化，最后存储数据。EXPORT 的过程正好与此相反。

3.3 XML 文档的细粒度操作

对 XML 文档的细粒度操作即元素级的更新操作，分为值更新操作和结构更新操作。其中值更新操作是对元素或属性的值的更新，结构更新操作是对 XML 文档结构的改动，比如增加一个元素，删除一个属性等。如果直接存储 XML 文档，无论是值更新还是结构更新，即使只做了很小的改动，都要重新存储整个文档。关系模型表达 XML 数据的能力很有限，一个值更新操作可能会影响到很多表中的值，而结构更新操作则可能给关系模式带来毁灭性的破坏。对于对象模型，值更新操作只影响到一个确定的对象，而结构更新则需要修改相应对象的模式定义，这样的代价可能比重新 IMPORT 的代价还大。OEM 模型和 XML 数据模型之间的映射比较简单，即使是结构更新，对整个 OEM 图的影响也就局限在对应的节点和边上。

XML 文档的 DTD 包含着 XML 数据的结构信息，如果结构更新操作在 DTD 的允许范围之内，在进行模型映射时考虑到可能的变化，就可以大大减少结构更新带来的影响。

4 XML 查询语言及处理

4.1 查询语言与模式

目前比较著名的 XML 查询语言有 Lorel[16]，XML-QL[17]，XML-GL[18]和 Quilt[19]等。这些查询语言各有优缺点[20]：XML-QL 和 XML 文档的集成性比较好，Lorel 的功能比较强，而 XML-GL 在图形化界面方面做得比较好。Quilt 综合很多语言的优点[19]：它的路径表达式参考了 XPath，变量绑定借鉴了 XML-QL，FLWR 表达式类似于 SQL，而且它还支持用户自定义函数。Quilt 不仅可以查询 XML 数据，也可以方便地查询关系数据，因此它的表达能力是很强的。正因如此，W3C 推荐的查询语言 XQuery[11]在很大的程度上是改进的 Quilt。XQuery 主要是在 Quilt 的基础上增加了与 XML 模式、XML 查询代数的结合。

W3C 为 XML 定义了模式，它和 XML DTD 有很大的不同：

1. XML Schema 本身是一个 XML 文档，而 DTD 有自己的语法。

2. XML Schema 利用名域（namespace）将类型定义和特定的模式相联系，一个 XML 文档可以对应多个模式，但只能对应一个 DTD。

3. DTD 主要定义内容模式（content model），只支持字符串类型；而 XML Schema 不仅定义了内容模式，而且提供丰富的数据类型，还允许用户自定义类型，甚至支持类似于继承的类型重定义，因此表达能力更强。

4. XML Schema 还支持类似于关系数据库的 unique，key，reference 和 null 值，这使得 XML 看起来更像数据库而不只是文档；DTD 对这些都不支持。

5. XML DTD 源于 SGML DTD，得到了广泛的支持；而 XML Schema 尚在制订之中，应用还有待时日。

4.2 XML 索引技术

在 XML 查询中，索引的作用至关重要。XML 数据是树结构，单纯的值索引远不够用，需要新的索引类型。Lore 系统在这方面做了很多研究[21,22]。这里参考 Lore，介绍了几种相应的索引。

·值索引 Vindex：可以根据元素名或属性名和值上的谓词条件找到满足条件的元素或属性。

·父子索引 Lindex：可以根据元素名找到所有由该元素名连接的父子元素对。

·路径索引 Pindex：可以根据路径找到所有该路径可达的元素。路径索引还可以记录统计信息，并作为可视化界面的一部分提供给用户。XML 文档的 DTD 规定了可能存在的路径表达式，因此可以利用 DTD 来方便建立路径索引。

对于传统数据库来说，索引是一种内部数据，用户不必知道它的存在，而且同样的数据在不同的数据库中的索引都是不一样的。因此，将数据从一个数据库转移到另一个数据库中时，需要重新建立索引。而在 XML DBMS 中，完全可以利用 XML 来传输索引数据。这样，在传送 XML 文档(数据)时，可以把文档相关的索引数据也转化为 XML 文档，和数据文档一起传送。数据库接收 XML 数据的同时也接收了索引数据，就可以马上利用这些信息来加速查询处理。

4.3 查询优化

查询优化有3个关键：搜索空间、代价估算、计划枚举。查询的搜索空间依赖于代数等价规则、物理操作集和查询执行策略。W3C 的 XML 查询工作草案中已初步提出一些代数等价规则。应用等价规则，同一个查询可以由多种代数表达，而每一种代数操作都可以用多种物理操作实现，而且 XML 查询可以有多种执行策略[24]。因此，查询的搜索空间往往很大，可以应用启发式规则来裁剪掉无用的搜索空间。XML 文档的 DTD 决定了 XML 数据树的结构，可以利用 DTD 减少路径表达式查询的开销[24]。

XML 查询的代价估算一般只考虑 CPU 代价和 I/O 代价。I/O 代价的估算在很大程度上依赖于 XML 数据的统计信息[23]。这些信息都和路径有关，因此可以直接记录在路径索引上。

查询的搜索空间往往很大，文[23]用一种贪心算法来加速枚举：每个逻辑计划节点都做一次局部的优化决策，生成当前节点的最优物理子计划。这样生成的物理计划可能不是最优的，但缩减了搜索空间，而且从 Lore 的实践看，结果还是比较好的[23]。

5 可视化用户界面

互联网上浏览和查询 XML 数据的用户绝大部分都是没有数据库知识的非专业人员，而 XML 查询语言本身又比较复杂，因此需要一个可视化的用户界面，用形象的方式表达 XML 数据，甚至 DTD，同时能让用户方便地输入查询条件。关系数据库中的 QBE 语言以表格的形式表示查询语言，让用户以举例的方式进行查询。在 XML 中也可以采取类似的方法。

XML-GL[18]就是一个图形化的 XML 查询语言，它定义了图形化的 XML 数据模型 XML-GDM，用户可以通过拖-放 XML-GDM 中对象来“写”查询语句和构造查询的输出。Lore 系统中的 DataGuide[21]是一个精确概括半结构化数据库结构的工具，同时也是一个简单的可视化用户查询界面。用户可以通过 DataGuide 查看一个复杂的 OEM 对象的结构，帮助构造 Lorel 查询语句。不懂 Lorel 的最终用户也可以通过 DataGuide 界面，以“by example”的方式查询数据库。

一个友好的用户界面应该包括下列的功能：

·为用户提供一个工具，让用户方便地了解 XML 数据的结构，在4.4节中已经提到可以让路径索引兼有这个功能；

·用户可以自由地输入查询的条件，对于任意的 XML 数据，查询条件的输入方式应该是统一的，而不需要为每一种 XML 数据定义一组查询条件，这个可以参考 QBE；

·用户在查询时，可能事先不知道数据的书写格式，因此应该有参考的例子来帮助用户书写正确的查询条件，这些例子应是从数据库中选出的；

·用户可以通过浏览器浏览 XML 数据，也可以查询，浏览本身也可以是某种条件下的查询，对于查询的结果可以再进行查询；

·用户甚至可以象浏览普通 XML 数据一样直接查看 XML 数据的索引和其他的统计信息，了解这些信息将有助于用户建立高效的查询语句。

总结与展望 XML 的出现使得互联网和数据库的联系更加密切，建立 XML DBMS 正是适应了这种趋势。由于 XML 的新特性，要在其上应用传统数据库的各种技术并不容易，还有很多问题有待研究。今后将就上述的几个方面作进一步研究，并着力于实现一个 XML DBMS，它将包括文中提出的数据存储、查询处理和可视化用户界面等。

参考文献

1 Abiteboul S, Buneman P, Suciu D. Data on the Web. Morgan Kaufmann Publishers, 2000

2 Goldman R, McHugh J, Widom J. From Semistructured Data to XML: Migrating the Lore Data Model and Query Language, WebDB99, Pennsylvania, June 1999. 25～30

3 Fallside D C. XML Schema Part 0: Primer. W3C Candidate Recommendation, October 2000

4 Fankhauser P, et al. The XML Query Algebra. W3C Working Draft, Feb. 2001

5 Widom J. Data Management for XML: Research Directions. Bulletin of the Technical Committee on Data Engineering. IEEE Computer Society, 1999, 22(3): 44～52

6 McHugh J S, et al. Lore: A Database Management System for Semistructured Data. SIGMOD Record, 1997, 26(3): 54～66

7 Software AG, Tamino. http://www.softwareag.com/tamino/

8 POET Content Management Suite. http://www.poet.com/CMSsdk.htm

9 OjbectDesign Inc. Excelon, the ebusiness information server. http://www.exceloncorp.com/

10 Deutsch A, Fernandez M F, Suciu D. Storing semistructured data with STORED. In: Proc. of the 1999 SIGMOD Conf. 1999

11 Chamberlin D, et al. XQuery: A Query Language for XML. W3C Working Draft, Feb. 2001

12 Fernandez M, et al. XML Query Data Model. W3C Working Draft, Feb. 2001

13 Kanne C-C, Moerkotte G. Efficient storage of XML data. In: Proc. of the 16th Intl. Conf. on Data Engineering, San Diego, Feb. 2000

14 Shanmugasundaram J, et al. Relational Databases for Querying XML Documents: Limitations and Opportunities. In: Proc. of the 25th VLDB Conf., 1999. 302～314

15 Hors A L, et al. Document Object Model (DOM) Level 2 Core Specification, W3C Recommendation, Sep. 2000

16 Abiteboul S, et al. The Lorel query language for semistructured data. International Journal on Digital Libraries, 1997, 1(1): 68～88

17 Deutsch A, et al. A Query Language for XML. Computer Networks, 1999, 31(11-16): 1155～1169

18 Ceri S, et al. XML-GL: a Graphical Language for Querying and Restructuring WWW Data. In: Proc. of 8th Intl. WWW conf. May 1999

一种检测多语言文本相似重复记录的综合方法

A Synthetical Approach for Detecting Approximately Duplicate Database Records of Multi-Language Data

俞荣华　田增平　周傲英

（复旦大学计算机系　上海200433）

Abstract Detecting approximate duplicate records in database is a key problem related to data quality. In this paper, we present a synthetical approach for recognizing clusters of approximately duplicate records of multi-language data. The key ideas are: (1) an efficient algorithm for sorting multi-language data; (2)an efficient edit-distance based pairwise comparison method for multi-language data; (3)using a priority queue of duplicates clusters and representative records strategy to respond adaptively to the data scale.

Keywords Approximate duplicates records, Clustering, Pairwise comparison, Priority queue

1. 前言

随着信息技术的广泛应用，如何有效利用不断激增的数据成为企业的迫切问题。数据仓库和数据挖掘技术为企业从浩瀚的数据海洋中获取有用的知识提供了一种有效的手段。然而，现实世界中的数据往往存在着大量的质量问题，从简单的数据输入错误到相对较复杂的数据间的语义不一致性。如果数据的质量达不到要求，那么数据挖掘这类技术产生的结果也不会理想，甚至产生错误的分析结果，从而误导决策。可见提高数据质量的重要性。

信息重复问题是影响数据质量的关键问题之一。理想情况下，对于现实世界中的一个实体，数据库或数据仓库中应该只有一条对应的记录。根据这样的理解，联邦数据库中的实体标识问题和面向对象数据库中的对象标识问题都可以被看作是信息重复问题。从比较狭隘的观点来看，如果两条记录在某些对应的字段(Field)上的值相等或者足够相近，那么可以认为这两条记录互为重复。我们讨论的相似重复记录检测也以这样的观点为基础。由于这些记录不完全一样，我们称之为相似重复记录。在不致引起混淆的情况下，为了方便，文中有些地方直接称之为重复记录。

检测和去除数据库中的重复记录并非一个新的问题，排序-合并方法是检测数据库中完全重复记录的标准方法[1]。它的基本思想是，先对数据集排序，然后检测相邻记录是否为重复记录。目前已有的检测相似重复记录的方法也大多以此思想为基础，不同之处在于排序对象的选取和记录比较的方法。文[2]中将整条记录视为一个长字符串进行排序，通过计算字符串编辑距离进行记录比较；文[3]以自定义的一个应用相关的健作为排序键，通过一组规则定义的相等理论判定记录是否重复；文[4]以记录的 NGram 值作为排序键，通过计算编辑距离进行记录比较。

上述方法的处理对象都是纯西文表示的数据，不能直接用来处理多语言数据。然而，在使用东方语言的国家里，数据经常是含有多语言的文本。比如下面这样一条描述个人信息的记录：

马明/浙江杭州市凤起路162号 B 座301室/ UT 斯达康通信公司/maming@263.net/6805201.

其中这个记录的有些字段全是中文，有些全是西文，有些既有中文又有西文。

与纯英文数据记录相比，多语言数据记录的排序和记录比较方法都会有所不同，本文中我们以中西文混合情形下的数据记录为例来介绍和讨论我们的方法。本文的主要贡献有：通过序值文件的方式解决了多语言文本的排序问题，并给出了一个有效的排序算法；针对多语言数据，给出了一种合理、高效的记录比较算法，其复杂度为 O(M * N)；采用以聚类为元素的优先队列和代表记录相结合的策略，大大减少了记录比较次数。

本文以中西文混合数据为例介绍了多语言文本排序问题，并给出了有效的排序算法。介绍了多语言数据的记录比较算法，整个检测算法的总体思想以及优先队列等关键策略。给出了实验结果评价和我们的工作。

2. 排序一初步聚类

初步聚类的目标是将潜在的重复记录调整到相邻位置，一般采用排序的方法。对于西文字符而言，排序也就是按字符在字符集中的次序排列(即字典序)。对于汉字而言，存在多种排序方式。在国家标准 GB2312-80中共收集汉字6763个，分成两级。一级字库包括汉字3755个，按拼音字母排序；二级字库包括3008个汉字，按部首排序。由此可见汉字本身的编码不满

俞荣华　硕士研究生，主要研究方向为数据质量和数据清洗。田增平　副教授。周傲英　教授，博士生导师。

19 Chamberlin D, Robie J, Florescu D. Quilt: An XML Query Language for Heterogeneous Data Source. WebDB2000, May 2000

20 Bonifati A, Ceri S. Comparative Analysis of Five XML Query Languages. SIGMOD Record, 2000, 29(1): 68～79

21 Goldman R, Widom J. DataGuides: Enabling Query Formulation and Optimization in Semistructrued Databases. In: Proc. of the 23rd VLDB Conf. Athens Greece, 1997

22 McHugh J, et al. Indexing Semistructured Data: [Technical report]. Stanford University Database Group, 1998

23 McHugh J, Widom J. Query Optimization for XML. In: Proc. of the 25th VLDB Conf. 1999

24 姚建中. XML 文档查询处理与数据库存储的研究:[浙江大学硕士学位论文]. 2000