

XML 数据库技术研究^{*}

秦杰¹ 杨树强 龚文华

(国防科技大学计算机学院 长沙 410073)

XML Database Researching

QIN Jie YANG Shu-Qiang DOU Wen-Hua

(Computer School, National University of Defence Technology, Changsha China, 410073)

Abstract XML database is a new database technology. This paper gives an overview about the last research about XML database and its related productions, then comparing XML database with traditional RDBMS. We analyze and point out some weaknesses about XML database, and give out our viewpoint.

Keywords NXD, XEDB, Semi-structure data

随着 Web 应用的不断扩展,基于 HTML 的 Web 信息表达方式已经不能适应人们进行信息查询和对 Web 数据进行管理的需要^[1]。由万维网协会(W3C)提出和设计的可扩展标记语言 XML(eXtensible Markup Language)正在逐步成为新一代 Web 数据描述和数据交换标准。XML 是一种自描述的半结构化语言,不仅能描述数据的外观,还可以表达数据本身的含义。在兼容原有 Web 应用的同时,XML 还可以更好地实现 Web 中的信息共享与交换。XML 的扩展性和灵活性允许它描述不同种类应用软件中的数据,从 Web 页面信息到数据记录,XML 能够使不同来源的结构化的数据很容易地结合在一起。各种基于 XML 的应用不断涌现,产生了对 XML 文档进行统一管理的需求。另外,传统的数据库在基于 Web 的应用中遇到了诸多的障碍,传统的关系数据库和面向对象数据库无法对基于 Web 的半结构化数据进行有效的管理,在这种情况下 XML 数据库应运而生。

目前,XML 数据库主要有两种类型:XML 本源数据库 NXD(Native XML Database)、支持 XML 的数据库 XEDB(XML-enable database)。NXD 是专门对 XML 文档格式的数据进行存取管理和数据查询的数据库。XEDB 是在传统数据库的基础上,通过增加对 XML 文档中数据的映射功能,从而实现了对 XML 文档中数据的存取管理^[3]。

目前,XML 数据库的主要用途可以概括为两方面:对 XML 文档进行有效的管理、对基于 Web 的各种数据源进行集成,提供数据查询功能。

XML 数据库作为一种新型的数据库技术得到了数据库研究人员、数据库厂家以及用户的普遍关注,本文结合 XML 数据库研究的最新进展和 XML 数据库最新产品的实际性能对 XML 数据库进行总体分析。内容安排如下:XML 数据库系统的组成;XML 的文档类型;XML 产品举例及其与传统数据库的比较;关于 XML 数据库需要解决的若干问题的思考。

1 XML 数据库系统的组成

当然,单纯的 XML 仅仅意味着 XML 文档,XML 本身并不是数据库。然而,当 XML 文档被用于数据存取和数据查找

时,XML 和它相关的技术结合就组成一个数据库管理系统,已经具备了数据库的基本特性,图 1 是 XML 数据库系统构成示意图。

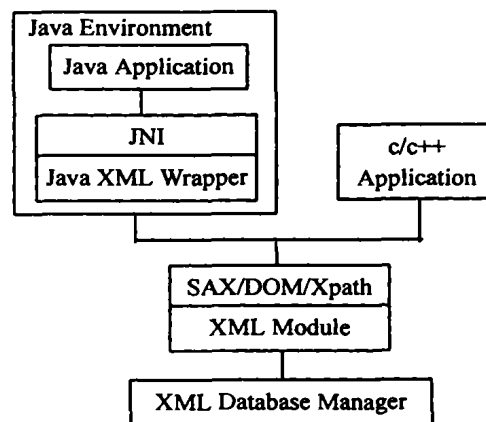


图 1 XML 数据库系统构成示意图

1.1 XML 数据库数据模型

DTD(文档类型定义)或者 Schemas 是 XML 数据库数据模型。对 XML 数据库内容的存取访问都依据该模型。该模型是树型结构^[4],模型中规定了元素(elements)、属性(attributes)、PCDATA 以及文档内容的顺序。XML 数据模型可以支持任意层次的数据嵌套,对半结构化数据提供了良好的支持。

在 DTD 中可以说明 XML 文档中的数据类型(datatypes),通过 datatypes 可以进行基于数据类型的查询操作,DTD 是对 XML 数据建立索引的一种主要手段。

Schemas 是 W3C 推荐的 XML 数据模型标准,它比 DTD 更加严格规范。目前大量使用的是 DTD,未来 XML Schema 有可能取而代之。

下面是一个关于学生记录的 XML 文档片段:

```
<gatech-student 编号="gt000x">
  <name>岗山</name>
  <age>21</age>
</gatech-student>
```

其对应的 DTD 如下:

^{*} 本文受国家自然科学基金资助(项目编号 60003001)。秦杰 讲师,博士生,主要研究方向:Web 数据库、XML 数据库技术。杨树强 博士,副教授,研究方向:数据库技术。龚文华 教授,博士生导师。

```

<! ELEMENT gatech--student (name,age)>
<! ATTLIST gatech--student 编号 CDATA>
<! ELEMENT name (#PCDATA)>
<! ELEMENT age (#PCDATA)>

```

1.2 XML 数据库的数据区以及存储机制

XML 文档是 XML 数据库的数据区,XML 数据库中的数据直接以 XML 文档格式存储。一个 XML 文档就是一个基本的存储单元,相当于关系数据库中的一个表格。

1.3 XML 数据库查询和操作语言

Xquery、XPath、XQL、XML-QL 都是数据库查询语言。

Xquery 是 W3C 提出的对 XML 文档进行信息查找的查询语言规范。其具体标准由 XML Query 工作组制订。Query 规范还在不断完善,其目标是提供一种灵活的基于 XML 的查询语言,使得访问 XML 文档可以像访问传统数据库那样方便。

XPath 是 W3C 提出的在 XML 文档内描述数据元素在文档中的位置的规范。W3C 在 2002 年 4 月 30 日推出的 XPath 2.0 标准中,容纳了 Xquery1.0 的基本要求。因此 XPath 不仅用于确定数据在文档中的位置,还支持数据查询。目前的大多数 XML 数据库都采用它实现数据查询。

XSLT 是对 XML 数据进行操作的语言。在对 XML 文档操作时,XPath 通常是与 XSLT 协同工作的。

1.4 XML 数据库处理工具

DOM、SAX、JDOM 是 XML 数据库处理工具。提供对 XML 文档的编辑、管理功能,以及与其它语言的编程接口。

文档对象模型(Document Object Model,DOM)是 W3C 推荐的对 XML 数据进行组织管理的标准和编程接口规范。JDOM 是采用 JAVA 语言实现的 DOM。

SAX(Simple API for XML)是目前大多数 XML 数据库实际使用的开发标准,几乎所有的 XML 解析器都支持它。与 DOM 比较而言,SAX 是一种轻量型的方法。

2 XML 文档类型

XML 文档可以分到两大类:以数据为中心、以文档为中心。

以数据为中心的文档:这种文档结构规范、数据粒度好(最小的独立数据单元是 PCDATA、elements、或者 attributes)、很少或者没有混合内容。文档中同层次的 elements、PCDATA 之间的顺序不作严格要求,类似于关系表。以数据为中心的文档主要用于数据存储或交换,便于机器理解和处理。

下面的销售记录就是一个以数据为中心的文档:

```

<SalesOrder 销售编号="00001">
  <Customer 客户编号="053">
    <CustName>ABC 公司</CustName>
    <Street>东风路23号</Street>
    <City>长沙</City>
    <PostCode>410073</PostCode>
  </Customer>
  <OrderDate>020615</OrderDate>
  <Item ItemNumber="1">
    <Part PartNumber="123">
      <Description>办公桌:木制,120 * 60mm,一年质量保证</p></Description>
      <Price>230</Price>
    </Part>
    <Quantity>15</Quantity>
  </Item>
  <Item ItemNumber="2">
    <Part PartNumber="456">
      <Description>椅子:皮椅,一年质量保证.</p></Description>
      <Price>110</Price>
    </Part>
    <Quantity>20</Quantity>
  </Item>

```

```
</SalesOrder>
```

以文档为中心的文档:这种文档内部数据结构不规则,内容是混合的,其内部包含的信息自身的物理结构很重要。数据的粒度比较大。以文档为中心的文档便于人们的阅读,是典型的半结构化的文档,具有不规则的内容,如书本、电子邮件、广告等等。

下面是一个以文档为中心的文档示例:

```

<Product>
  <Intro>
    The <ProductName>Turkey Wrench</ProductName> from <Developer>Full Fabrication Labs, Inc.</Developer> is <Summary>like a monkey wrench, but not as big.</Summary>
  </Intro>
  <Description>
    <Para>The turkey wrench, which comes in <i>both right- and left-handed versions (skyhook optional)</i>, is made of the <b>finest stainless steel</b>. The Read-i-grip rubberized handle quickly adapts to your hands, even in the greasiest situations. Adjustment is possible through a variety of custom dials.</Para>
    <Para>You can:</Para>
    <List>
      <Item><Link URL="Order.html">Order your own turkey wrench</Link></Item>
      <Item><Link URL="Wrenches.htm">Read more about wrenches</Link></Item>
      <Item><Link URL="Catalog.zip">Download the catalog</Link></Item>
    </List>
    <Para>The turkey wrench costs <b>just $19.99</b> and, if you order now, comes with a <b>hand-crafted shrimp hammer</b> as a bonus gift.</Para>
  </Description>
</Product>

```

3 XML 数据库产品简评及其与传统数据库的比较

3.1 NXD 产品概述

截止2002年8月,共有32种 NXD 产品,这些 NXD 产品基本上是在2000年以后面世的^[1]。比较有代表性的 NXD 产品有 Tamino、dbXML、eXcelon 和 XHive/DB,分别由美国的 Software AG 公司、dbXML Group LLC 公司、eXcelon 公司和荷兰的 The Connection Factory 公司研制。

Tamino 是第一个商用 native-XML 数据库,2002年5月发布的最新版为 2.3.1,系统提供与其它数据源的调用接口,可以对关系型数据进行存取管理,其管理界面基于 Web。与主流的关系数据库管理系统相似,包括与数据库交互和对数据模型进行编辑(editting schemas)的 GUI 工具。其 Schemas 基于 DTD,查询语言基于 XPath,可实现对任意数据元的检索。根据 Gartner Group 的研究报告,其查询功能在某些方面甚至超过 SQL。

3.2 NXD 与传统数据库的比较

与传统数据库相比,NXD 具有以下特点:

(1)对半结构化数据进行有效的存取和管理。传统的关系数据库或者面向对象数据库都无法对半结构化数据进行有效的管理,而 XML 本身就是一种半结构化的数据模型,因此专门用于对 XML 文档进行存取管理的 NXD 实际上就是一种半结构化数据库。

(2)提供对标签(Tag)和路径的操作。传统数据库语言不能对元素名称操作,半结构化数据库提供了对标签名称的操作,还包括了对路径的操作。当数据本身具有层次结构时,由于 XML 数据格式能够清晰表达数据的层次特征,因此 XML 数据库便于对层次化的数据进行操作。

(3)顺序性。在传统的数据库表中,表项(field)之间的顺序是可以互换的。以文档为中心的 XML 文档与传统的数据表不同,其内容是有顺序的,不允许随便调整 elements、at-

tributes、PCDATA 的顺序。当然,顺序性使得对 XML 文档的查询、修改比传统数据库的相应操作复杂。

(4)与各种数据的交换能力.XML 本身是一种标准的数据交换格式,因此 NXD 能够存储和查询各种不同结构类型的文档,对异构环境的信息存取提供了良好的支持,为异构环境下的数据集成提供了一种新的方法,这一特性对于 Web 数据管理来说是十分重要的。

当然,传统的数据库具有完备的理论基础、简洁的数据模型、透明的查询语言和方便的操作方法等优点,传统的关系型和对象关系数据库适合对结构化数据的管理。但是由于它们本身并没有针对网络的特点和要求进行设计,因此并不适用于网络环境下大量存在的半结构化数据的管理。

3.3 XEDB 产品概述及其与传统数据库的关系

目前已知的 XEDB 产品有 Access 2002、SQL Server 2000、IBM 的 DB2 XML Extender、FileMaker、FoxPro、Informix、Objectivity/DB、Oracle 8i,9i 等。它们都是使用 DTD 实现与关系表格的转换(mapping),对 XML 文档的查询符合 W3C 推荐的 XPath 规范,其中,微软、IBM 和 Oracle 的 XEDB 核心都采用 XQuery 标准执行查询。

传统的关系数据库中的数据表(Table)类似于以数据为中心的文档,因此由 XEDB 管理的以数据为中心的 XML 文档可以方便地存储到关系数据库中,关系数据库中的数据表也可以转换成 XML 文档由 XEDB 管理(当然,使用中间件,如 Microsoft's ADO.NET、IBM's Database DOM 等也可以实现 XML 文档与关系数据库之间的转换、数据查询等功能,这不在本文讨论范围之内)。

实际上,XEDB 只是在传统数据库基础上增加了对 XML 数据的映射机制,只能对以数据为中心的、结构化程度较高的 XML 文档进行管理,能够实现这类 XML 文档与传统数据库之间的映射。对于 XEDB 来说,数据库内部仍然采用关系表的格式存储数据,当用户进行基于 XML 的查询或处理时,XEDB 可以根据需要重新组合这些数据。当 XML 文档结构较为复杂时,XEDB 无法发挥作用。因为如果把一个以文档为中心的文档存储到关系表中,XML 文档的内部结构完全被破坏,通常无法再从关系表复原回原先的 XML 文档,会导致 XML 文档意义的改变。

从本质上看,XEDB 就是一个具有对以数据为中心的、结构化程度比较好的 XML 文档进行存取管理功能的关系数据库或是面向对象数据库。

XEDB 只可以对以数据为中心的文档进行管理,而 NXD 不仅可以管理以文档为中心的文档,也可以管理以数据为中心的文档。如果要处理的文档是以数据为中心时,可以使用 XEDB;如果要对以文档为中心的文档进行管理,应当选择 NXD。如果 XML 文档同时具有以文档为中心和以数据为中心的特征,则应当采用 NXD。

4 XML 数据库亟待解决的若干问题

XML 数据库作为一种新型的数据库技术,目前还存在许多需要改进的方面。

4.1 聚合问题

聚合能力是数据库的重要功能之一,是进行联机分析处理(OLAP)的关键。现阶段 NXD 产品在执行聚合任务时表现不佳。

由于 XML 文档中的所有内容都被统一作为字符处理,

不支持复杂的数据类型定义;再加上 XML 数据是以层次化模型进行管理的,当 XML 文档中数据层次较多(如大于7层)时,对处于不同层次的相关信息汇集起来进行求和、求平均数等算术等运算时,就必须增加许多附加的代码,计算过程比较繁琐,远远没有在执行算术操作那样简单方便。

4.2 数据查询与更新能力有待提高

Xquery 查询语法规则不支持组、排列和摘要数据等功能,也不支持数据更新、插入和删除等功能^[10]。

由于 XML 文档自身的结构特点,目前还无法建立与关系数据库功能性能相当的索引机制。当需要对 XML 数据库进行查询或内容更新时,一般的办法是对整个数据区从头到尾进行查找。如果对一个信息量巨大的 XML 文档进行查询或内容更新时,这显然是不能容忍的。

常规的解决方法是选择指定的 element、或者 attribute 建立索引,如同在关系数据库里选择特定的 field 用于索引一样。当对 XML 文档建立索引时,必须把需要索引的 element、或 attribute 所在文档中的位置(Path)同时保存在索引文件中,因此,XML 文档的索引文件占用的存储空间比关系数据库的索引文件要大许多;如果要对多个 element、或 attribute 建立索引,将占用大量的存储空间。加上 XML 数据库主要是针对关系复杂的半结构化数据进行管理的,其索引机制比关系数据库复杂得多,目前实现的基于 XML 文档的索引机制远没有关系数据库的索引机制灵活。

下面是 DB2 的 XML EXTENDER 表采用 XPath 方式查询的示例。查询结果是返回所有邮箱的收件箱中标题内容含有“XML”的用户名称。

```
TABLE mail-user
user-name VARCHAR(20)NOT NULL PRIMARY KEY
passwd VARCHAR(10)
mailbox XMLVARCHAR
SELECT user-name FROM mail-user WHERE
extractVarchar ( mailbox,"/Mailbox/Inbox/Email/Subject ") LIKE
"%XML%"
```

为了提高查询速度就需要对邮箱(XMLVARCHAR)建立索引。显然这种索引机制不适用于较大的 XML 文档,目前的 NXD 系统只适合于对 1M 以下的 XML 文档进行管理^[6],当管理的文档过大时,整个系统的开销(占用的存储空间,查询时间等)将是巨大的。

对 XML 文档建立类似于关系数据库那样索引机制的各种探索,也是目前 XML 数据库的一个研究热点。

4.3 标准化问题

根据 Ronald Bourre^[3]2002年8月的最新统计,XEDB 产品有10种,NXD 产品有32种,由于这些产品在开发之前还没有具体的统一规范,不同厂家的数据库产品之间存在差别,它们之间的互操作能力值得怀疑。尽管 W3C 在 XML 数据标准化方面做了大量工作;XML:DB 作为一个致力于 XML 数据库标准化和推广 XML 数据库产品的国际组织,也在努力工作,不断推出基于 XML 数据库的开发规范,但其规范常常比 XML 数据库产品出现的时间还晚,在实际的 XML 数据库技术发展过程中只能起到一定的指导作用。再加上 XML 数据库作为一种新兴的数据库技术,研究者众多,不断有研究者和 XML 数据库厂商对 XML 数据库的功能进行扩展,提出新的技术方案^[11~13],估计在近两三年内很难制订出一个比较全面的得到统一支持的数据库开发标准。

结论 XML 数据库是一种基于 Web 的新型的半结构化数据库.XML 作为一种新型的数据库格式具有一些特别的优

势:它是自描述的,使用 Unicode 编码方式,简单通用,可以描述树型或图型结构的数据,适合于对半结构化数据的管理。这些特点不论是 DBMS 还是 OODB 都是无法取代的。另外,XML 数据库的另一优势还在于数据的移植能力、以及对现有的异构数据库系统的集成能力。NXD 适合管理复杂数据结构的数据集,如果已经以 XML 格式存储信息,则采用 NXD 利于文档的存取和检索,能够提供高质量的全文搜索引擎,特别适合对半结构化数据的管理。

当然,传统的关系型和对象关系数据库也有自己的应用领域,如果要对结构化的数据进行管理,传统的关系数据库还是首选。

对 XML 数据库研究的重点应当放在 NXD 上面,现阶段,NXD 的数据更新以及数据查询、修改能力需要进一步提高,在并发性方面,NXD 还缺乏有效的管理机制,缺乏多重管理能力,协同工作能力、规划能力不强。这些不足之处正是 XML 数据库近期和未来的研究热点所在。

参考文献

- 1 Seligman L, Rosenthal A. XML's Impact on Databases and Data Sharing. IEEE Computer, June 2001. 59~66
- 2 XML: DB, XML: DB Initiative for XML Databases. [http://](http://www.xmldb.org/index.html)

- 3 Bourre R. XML Database Products, 14 Aug. 2002. <http://www.rpbouret.com/xml/XMLDatabaseProds.htm>
- 4 Ipedo XML Database. <http://www.ipedo.com/html/products-xml-dat.html>
- 5 Obasanjo D. AN EXPLORATION OF XML IN DATABASE MANAGEMENT SYSTEMS. <http://www.25hoursaday.com/StoringAndQueryingXML.html>
- 6 Clark, James DeRose S. XML Path Language (XPath) 2.0, W3C Working Draft 30 April 2002. <http://www.w3.org/TR/xpath20/>
- 7 Bourett, Ronald. XML and Databases. June 2002. <http://www.rpbouret.com/xml/XMLAndDatabases.htm>
- 8 Chamberlin Don, et al. XQuery 1.0: An XML Query Language (Working Draft). June 2001. <http://www.w3.org/TR/2001/WD-xquery-20010607>
- 9 Cheng, Josephine, Xu, Jane. IBM DB2 Extender. ICDE '00 Conference, San Diego. Feb. 2000
- 10 W3C Working Draft. XML Query Use Cases, 30. April, 2002. <http://www.w3.org/XML/Query>
- 11 Tatarinov I, Ives Z G, Halevt A Y. Updating XML. SIGMOD, 2001
- 12 Jaenicke C. Native XML Databases and Relational Database. <http://www.exln.com/products/whitepapers>
- 13 University of Washington Database Group. Management of XML Data. <http://www.data.cs.washington.edu/xml>
- 14 W3C Technical Reports and Publications. <http://www.w3.org/tr/>
- 15 Airi S, Frank W T. System Desiderata for XML Databases. In: Proc. of the 27th VLDB Conf., 2001

(上接第5页)

环保等对高性能计算网格即信息网格的需求将非常巨大。目前来看,网格计算在我国主要可以提供以下服务:

•资源数据:国家在过去几十年已经花费了大量的财力收集各种地下、地面、大气的资源数据,采用的手段包括地下勘探、地面人工测绘和监测、航空遥感、卫星遥感等。但由于目前这些数据零碎地分散在不同的地方,共享困难,利用率低,将它们放在网络上共享将带来很多好处。

•高性能计算:计算网格的建立有利于各计算中心实现资源共享,充分利用硬件和软件资源,节约成本。它能在基础研究、汽车、大型水电工程、石油勘探、气象气候、航空、交通、金融、医疗等领域发挥空前的作用。

•生物信息:生物科学的信息化和全球化已成为大势所趋,我国的生命科学界对网格也有强烈呼声。以北京大学生命学院为龙头的一些生物科研机构已经加入了相关的国际组织,并在网络上分享现代生物信息资源以提高工作效率。

•环境保护:我国已经建立了若干环境数据收集网和计算机网,但由于没有实现统一的网上共享和管理,使用极不方便。网格建立后,可以解决这一问题,并且可以提供新闻媒体的缓存服务、目录服务、数据挖掘和分类服务等多种新的应用形式。

参考文献

- 1 <http://www.863.org.cn/863-105/active/200205280194.html>
- 2 肖连兵,黄林鹏. 网格计算综述. 计算机工程, 2002, 28(3): 1~3
- 3 Foster I, Kesselman C, et al. The Grid: Blueprint for a New Computing Infrastructure. Morgan Kaufmann, San Francisco, 1999
- 4 <http://www.yesky.com/20020702/1618643.shtml>
- 5 Foster I, Kesselman C, Tuecke S. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. Int'l J. High-Performance

- Computing Applications, 2001, 15 (3): 200 ~ 222. <http://www.globus.org/research/papers/anatomy.pdf> (current June 2002)
- 6 <http://www.st121.com.cn/text/d/210.htm>
- 7 Foster I, Kesselman C, Nick J M, Tuecke S. Grid Services for Distributed System Integration. 15. <http://www.globus.org/research/paper>
- 8 Baker A, Buyya R, Laforenza D. The Grid: A survey on global effort in grid computing. <http://www.byte.com>
- 9 都志辉, 李三立, 等. 网格计算及其原型实现研究. 计算机科学, 2002, 29(8)
- 10 <http://www.computer.org/proceedings/hpdc/0287/02870034a-bs.htm> (current June 2002)
- 11 Foster I, et al. The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration; [Tech. report]. Glous Project. <http://www.globus.org/research/papers/ogsa.pdf> (current June 2002)
- 12 Christensen E, et al. Web Services Description Language (WSDL) 1.1. W3C Note, 15 Mar. 2001. <http://www.w3.org/TR/wsdl> (current June 2002)
- 13 Catlett C, Smarr L. Metacomputing. Comm. ACM, June 1992. 44~52
- 14 Paton N W, et al. Database Access and Integration Services on the Grid; [Tech. report UKeS-2002-3]. National e-Science Centre. <http://www.nesc.ac.uk>, 2002
- 15 <http://www.globus.org>
- 16 <http://www.gridforum.org>
- 17 Decusatis C. Grid computing: the next (really, really) big thing. <http://www.byte.com>
- 18 Foster I, Kesselman C. Computational Grid. The Grid: Blueprint for a future computing infrastructure. Morgan Kaufmann Publishers, 1998
- 19 Waananen A, Ellert M, Konstantinov A, et al. An overview of grid architecture for scientific computing. <http://arxiv.org/PS-cache/cs/pdf/0205/0205021.pdf>
- 20 Vazhkudai S, Tuecke S, Foster I. Replica Selection in the Globus Data Grid. <http://cn.arxiv.org/abs/cs.DC/0104002>