

# 基于 XML 的空间元数据系统的思考

曹 卫

(厦门理工学院 计算机科学与技术系, 福建 厦门 361005)

**摘 要:**为提高元数据系统实现中算法结构的直观性、兼容性以及转换效率,通过分析比较 XML 的两种模式 XML Schema 和 XML DTD,整合关系数据库以及 XML 文档各自的优点,提出了在系统中将其两者相结合的方法。利用了 XML 进行各项操作,同时又采用关系数据库进行系统数据的存储,既保证了系统的高效性,也保证了数据存储的安全性。并通过树型结构图实现了 XML 文档与关系数据库数据之间的双向转换算法,极大地提高了元数据系统实现中算法的直观性、兼容性和转换效率。

**关键词:**空间元数据;XML;关系数据库

**中图分类号:**TP311

**文献标识码:**A

**文章编号:**1673-629X(2010)07-0032-04

## The Thinking of Spatial Metadata System Based on XML

CAO Wei

(Department of Science and Technology of Computer, Xiamen University of Technology, Xiamen 361005, China)

**Abstract:** To improve the intuitive and the compatibility and the conversion efficiency of the algorithm structure of meta-data system, propose the view of combining two kinds of pattern of XML Schema and XML DTD through analyzing and comparing the advantages and disadvantages of the relational database and the XML documents in system. Make use of XML to carry out various operation. At the same time carry out the system data adopting relation data. Then it ensures the security of the data stores and the high-effect of the system. At last give the bilateral switching algorithm of the XML documents and the relational database, use the tree structure drawing in the concrete algorithm. It has improved the intuitive and the compatibility and the conversion efficiency of the algorithm in meta-data system.

**Key words:** spatial metadata; XML; relation database

## 0 引言

随着数字地球概念在全球范围内的广泛传播,人们对于数字地球所需关键技术的认识已越来越深刻。而被戈列列为数字地球六大技术之一的 Metadata 则更加引人注目,成为数字地球建设能否走向成功和实用的关键<sup>[1]</sup>。目前,人们对于地理信息元数据的研究已广泛展开,并形成了多个元数据标准,建立了各种元数据系统。

## 1 XML 技术

与一般的系统相比,GIS 的数据具有数据量大、表达形式复杂的特点,各种空间操作计算量大,而且大多与空间对象的空间位置关系紧密相关,因此空间索引的有无及其性能的发挥直接影响到空间数据库和 GIS

系统的整体性能。XML 起初主要是为了增强应用程序对 Web 上获得的文档的解释和操作能力<sup>[2~4]</sup>。然而,从数据库的角度来看,XML 从另一个方面提出了一个令人兴奋的可能性:对于存储在 XML 文档中的数据,应该能够查询这些文档中的内容和结构<sup>[5]</sup>。

应用程序通过接口访问 XML 文档。目前较常用的 XML 数据访问接口有两种:DOM 和 SAX,DOM (Document Object Model)是 W3C 制定的接口标准,是一种规范,它具有平台无关性和语言无关性;DOM 接口定义了一系列对象来实现对 XML 文档数据的访问和修改,它将 XML 文档转换为树形文档结构,这棵树是 XML 文档内元素之间关系的反映,通过这棵树,可以访问和修改 XML 文档数据。SAX (Simple Application for XML)不是 W3C 的正式标准,但作为一种基于事件的 XML 编程接口,已经被各种 XML 团体广泛认可<sup>[6~8]</sup>。SAX 采用单向遍历文档的方法,它是由事件驱动的,当 XML 解析器遇到特定的事件,就调用相应的函数进行处理。DOM 接口能实现对文档元素的随机访问,但代价是将整个树形结构放入内存,当 XML

收稿日期:2009-10-27;修回日期:2010-01-29

基金项目:厦门市科技计划指导性项目(3502Z20077018)

作者简介:曹 卫(1980-),女,江苏靖江人,硕士,研究方向为信息管理与信息系统。

文档十分复杂和庞大时,要占用可观的资源,处理速度也会变得很慢。SAX 接口不需要将文档装入内存,但它只能实现文档的顺序访问,不能随机访问任意数据。

## 2 关系型数据库与空间元数据库 (XML 文档)之间的转换

### 2.1 Schema 和 DTD

目前,如何将关系数据表示为 XML 数据和如何实现基于关系数据库的 XML 数据存储、查询和更新已经成为研究的热点<sup>[9]</sup>。Deutsh, Florescu, Kossman, Shanmugasundaram 和 D W Lee 等人都在 XML 与关系数据的相互转化方面作了较深入的研究。然而由于 XML Schema 成为正式推荐标准的时间较晚,加之 XML DTD 语法相对简单,所以现在大部分的研究和应用都是基于 XML DTD 展开的<sup>[10]</sup>。但是,XML DTD 在对关系数据的描述方面明显存在着不足,例如 XML DTD 有限的数据类型根本无法完成对关系数据类型的一一映射,也无法实现大部分数据规则的描述。XML Schema 提供了更多的内建数据类型,并支持用户对数据类型的扩展,基本上满足了关系模式在数据描述上的需要,这一点可以作为 XML Schema 比 XML DTD 更适合描述关系数据的一个主要原因<sup>[11]</sup>。

### 2.2 空间元数据库(XML 文档)转换为关系型数据库

XML Schema 是一种 XML 文档,所以 DOM 技术支持对 XML Schema 的操作,也就是说 XML Schema 可以通过 DOM 技术转化为树型结构,不再存在 DTD 中会出现的嵌套定义问题,采用 XML Schema 进行定义之后,解决 XML 文档转换为关系型数据库中表之间的问题,所以可以将 XML Schema 利用 DOM 技术转换为树型结构,摆脱了传统的 DTD 定义模式的限制<sup>[12,13]</sup>。

对于 XML 文档与关系数据库的转换在文献[4, 5]中已经给出了一个十分完善的基于 DTD 的转换,文中在其基本思想的基础之上针对 XML Schema 对转换方式进行了一定的改进,并用 ASP 进行了实现。

下面给出本系统中关于公交情况的 XML Schema 文档如下所示,针对这个文档给出了经过 DOM 解析后得到的树型结构,如图 1 所示。

```
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:element name="buses" type="bustype"/>
</xsd:schema>
```

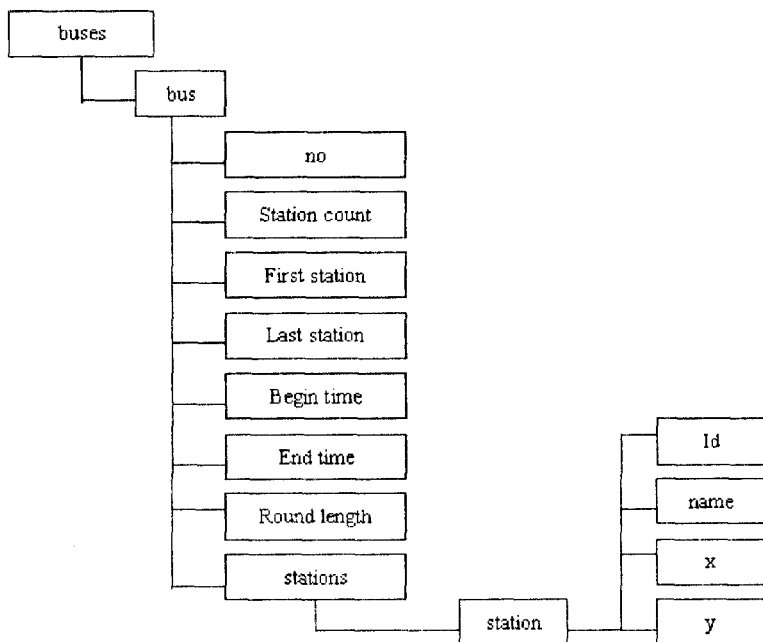


图1 对应于公交系统 XML Schema 文档的树型结构图

```
<xsd:complexType name="bustype">
  <xsd:sequence>
    <xsd:element name="bus" minOccurs="0" maxOccurs="unbounded">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="no" type="xsd:positiveInteger"/>
          <xsd:element name="stationcount" type="xsd:positiveInteger"/>
          <xsd:element name="firststation" type="xsd:string"/>
          <xsd:element name="laststation" type="xsd:string"/>
          <xsd:element name="begintime" type="xsd:time"/>
          <xsd:element name="endtime" type="xsd:time"/>
          <xsd:element name="roundlength" type="xsd:string"/>
          <xsd:element name="stations" type="station-type"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
```

```

<xsd:complexType name="stationtype">
<xsd:sequence>
<xsd:element name="station" minOccurs="0"
maxOccurs="unbounded">
<xsd:complexType>
<xsd:sequence>
<xsd:element name="id" type="xsd:positiveInteger"/>
<xsd:element name="name" type="xsd:string"/>
<xsd:element name="x" type="xsd:decimal"/>
<xsd:element name="y" type="xsd:decimal"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>

```

根据图 1 得到 XML 文档中各个元素的简略的关系如图 2 所示。根据图 2 的关系图可以看出对应于 XML Schema 的关系数据库的表以及表之间的相互关系。初步可以断定在该模型中有两个表:一个是 bus 表,另外一个表是 station 表。由图 1 可以看到表 station 应该是表 bus 的下属,那么只要找到表 station 中可以代表 station 中各个元素的属性也就是关键字,然后在表 bus 中相应添加一个属性即可。在这里 station 的关键字是 id,那么在 bus 表中添加属性 id 就可以了,这样关系数据库的整体结构就已经基本实现。

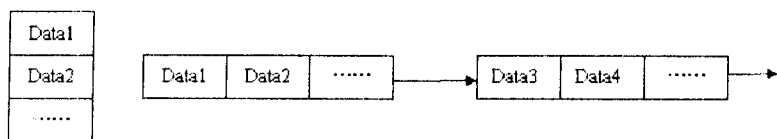


图 2 一些可用数据结构表示图

如果遇到更多层次的关系图,也可以按照这种方法先初步对关系数据库的表和表之间的关系进行设定,然后再看各个表之间的关系,如果是同事关系,那么表的属性就不用再进行添加,如果存在从属关系,则像图 1 一样找到低层表的关键字将其作为属性加入到高层表中。

根据以上所述,得到关系数据库结构之后就可以将相应记录转换为关系数据。

其中的 Data1, Data2 等分别表示车次、距离等信息,可以视需要增加其他数据。

### 2.3 关系型数据库转换为空间元数据库

在本系统中使用的是 ASP 语言,根据前面所述 XML 文档转换为关系型数据库的转换方法知道可以

根据关系数据库中表的关系得到与图 1 相似的关系图,然后再根据该图写出相应的 XML Schema 文档,再应用语言进行实现,具体的实现方法如下:

使用 ADO 生成 XML 记录集下面这个函数可以把一个数据库中数据直接转换成 XML 可读的文档(主要使用在 ASP 中),这个函数可以看作是 ADO 方法中的 GetString 方法(它主要用来将 ADO 的记录集转换成 HTML 格式)的补充。

```

<%
Function RecordsetToXML(oRec,sName)
Dim oFld
Dim sXML
Dim sSpaces
Dim sNames
sSpaces=Space(iLevel*2)
If Right(sName,1)="s" Then
sNames=sName
sName=Left(sName,Len(sName)-1)
Else
sNames=sName & "s"
End If
sXML=sSpaces & "<" & sNames & ">" &
vbCR
iLevel=iLevel+1
sSpaces=Space(iLevel*2)
oRec.MoveFirst
While Not oRec.EOF
sXML=sXML & sSpaces & "<" &
sName & ">" & vbCR
For Each oFld In oRec.Fields
If oFld.Type=adChapter Then
Set oChapter=oFld.Value
If Not oChapter.EOF Then
iLevel=iLevel+1
sXML=sXML & RecordsetToXML(oChapter,
oFld.Name)
iLevel=iLevel-1
End If
Else
sXML=sXML & sSpaces & "<" & oFld.
Name & ">" &
oFld.Value & "</" & oFld.Name & ">" &
vbCR
End If
Next

```

```

sXML = sXML & sSpaces & "</" & sName &
">" & vbCR
oRec.MoveNext
Wend
iLevel = iLevel - 1
sSpaces = Space(iLevel * 2)
sXML = sXML & sSpaces & "</" & sNames &
">" & vbCR
RecordsetToXML = sXML
End Function
%>

```

上面是完整的代码段,可移植性能是很不错的,只要在页面中包含该程序就可以使用这个技术。

#### 2.4 算法评价

在传统的数据库中,有关一个 Database 的操作,都必须建立在了解整个数据库的结构,也就是元数据的基础上,数据库创建、修正、备份、维护等应用程序都必须将数据库的元数据及程序逻辑合成在一起,即使能独立开来的,也是使用一套独有的、非开放的格式来描述其元数据,这就使得不同数据库应用程序之间交换数据、协同工作的难度大大增加。在极端情况下如果要完全实现  $N$  个独立数据库之间的转换,将需要设置  $N(N-1)/2$  个交换模块,其复杂度为  $O(N^2)$ ,由此可以看出,在传统的环境下,发生交换的系统数量级一般在  $10^0 \sim 10^1$ ,因此采用这种方式还是可以接受的,然而在网络飞速发展的今天,需要交换数据的数量级提升到了  $10^2 \sim 10^4$ ,因此在系统中采用了 XML 技术,使用 XML 来定义独立数据库的元数据,并将元数据和数据一起描述成 XML 进入交换体系,可以使交换代价大大减少。若要实现以上所说的  $N$  个数据库之间的完全图交换,则只需要构建  $N$  个转换模块,其复杂度为  $O(N)$ ,复杂度明显降低。

另外,以上给出的两个算法核心思想都是根据 DOM 技术支持对 XML Schema 的操作这一优点设计的,文中提出的方法由于可以直接通过 DOM 技术转换为树型结构,从而可以很容易地得到各元素之间的同事或者从属关系,因而转换过程就变得非常简单、直观,较之以往采用 DTD 格式构造的算法无论是从数据结构还是从转换途径上来说都有很大的进步。但值得一提的是虽然现在的趋势是使用 XML Schema 进行数据定义,但是以往的程序依然有很多是采用的 DTD 技术,因为采用 DTD 进行的数据定义中经常会出现嵌套结构,所以在这种情况下要么就采用文献[4,5]的算法,要么就利用 Schema 进行重新定义,一般在数据量较大的情况下为了防止进行重新定义会破坏数据的整

体设计则采用文献[4,5]的算法,但是从文献[4,5]可以看出各种的转换关系都是比较复杂的,而且直观性也比较差;如果数据结构比较简单,或者数据量相对较少,则才用重新定义,然后采用文中的算法,这样从结构的直观性以及兼容性等方面来说都有很大的好处。

### 3 结束语

文中结合 XML 文档和关系数据库各自的优势,利用了 XML 进行各项操作,同时又采用关系数据库进行系统数据的存储,这样既保证了系统的高效性,也保证了数据存储的安全性。文中实现了 XML 文档与关系数据库数据的双向转换算法,在具体的实现算法中采用了树型结构图进行实现,给出了算法的实现,并在最后给出了对于该算法的初步评价。

#### 参考文献:

- [1] Florescu D, Kossman D. Storing and Querying XML Data Using a RDBMS[J]. IEEE Data Engineering Bulletin, 1999, 22(3): 27-34.
- [2] 王建涛, 朱龙文. 基于 XML 元数据描述的空间数据共享管理平台的实现与应用[J]. 测试工程, 2007, 16(1): 12-15.
- [3] 郑建标. 一种基于 XML 和元数据的工具框架的研究[J]. 微计算机信息, 2007, 23(9): 197-199.
- [4] Ceri S, Comai S, Damiani E, et al. XML - GL a graphical language for querying and restructuring XML document[C]// Proceedings of www8. Toronto, Canada: [s. n.], 1999.
- [5] Czarnecki K, Eisenecker U. Generative Programming: Methods, Tools, and Applications[M]. [s. l.]: Addison - Wesley, 2000.
- [6] Bonifati A, Ceri S, Paraboschi S. Active rule for XML: A new paradigm for E-services[J]. The VLDB Journal, 2001(10): 99-101.
- [7] Bonifati A, Ceri S, Paraboschi S. Active rule for XML: A new paradigm for E-services[J]. The VLDB Journal, 2001(10): 114-116.
- [8] 张哲. 基于 XML 的元数据体系的数据交换[J]. 计算机工程与应用, 2003(10): 180-184.
- [9] 简广林. 基于元数据体系的分布式空间数据模型的研究及其在 WebGIS 中的应用[D]. 北京: 华北电力大学, 2003.
- [10] 庞芳, 李昭春. 基于 9210 工程的通用气象信息编辑与传输系统[J]. 广西气象, 2004(3): 47-52.
- [11] Goldfarb C F, Prescod P. XML 手册[M]. 第4版. 张晓晖, 王艳斌, 等译. 北京: 电子工业出版社, 2003: 147-323.
- [12] 李京, 庄成三. 利用 XML DOM 创建强大的 XML 应用[J]. 计算机应用研究, 2002, 19(2): 62-64.
- [13] 王津涛, 白乃侠. 基于 XML 元数据的研究与开发[J]. 计算机工程与设计, 2004, 25(7): 1086-1088.