第 28 卷 第 7 期
2005 年 7 月
计 算 机 学 报
CHINESE JOURNAL OF COMPUTERS
Vol. 28 No. 7
July 2005

基于 XML 的地理信息元数据系统

罗英伟 汪小林 许卓群
（北京大学计算机科学技术系 北京 100871）

摘 要 首先对地理信息元数据的内容从不同角度进行了研究：地理信息元数据内容由描述数据自身的元数据信息和描述数据之间关系的元数据信息构成；根据地理信息元数据的规范程度对地理信息元数据进行分类，随规范程度递减分为标准元数据、共识元数据和用户元数据；同时还分析了利用 XML 来表示地理信息元数据方法. 最后，设计了一个基于 XML 的地理信息元数据系统 GeoMeta，对其体系结构和各个部分的功能进行了分析，并详细介绍了基于 XML 的地理信息元数据存储和查询策略.

关键词 地理信息元数据；XML；Z39.50；地理信息元数据系统；GeoMeta
中图法分类号 TP391

XML-Based Geospatial Metadata System

LUO Ying-Wei WANG Xiao-Lin XU Zhuo-Qun
(*Department of Computer Science and Technology, Peking University, Beijing* 100871)

Abstract The content of geospatial metadata is studied from different aspects: geospatial metadata consist of information that describe both a data set itself and the relation among different data sets; geospatial metadata are classified as standard metadata, common metadata and user-defined metadata according to their normative degree; and the method of representing geospatial metadata by XML is analyzed. At last, an XML-based geospatial metadata system GeoMeta is designed. The architecture and the function of different parts of GeoMeta are explored first, and then XML-based storage strategies and query strategies of geospatial metadata are introduced in details.

Keywords geospatial metadata; XML; Z39.50; geospatial metadata system; GeoMeta

1 引 言

随着信息技术的广泛应用，社会各领域对空间信息的需求也越来越大[1]. 特别是自从数字地球的概念提出后，空间信息技术的发展有了很大的发展. 作为数字地球的重要支撑技术之一的地理信息元数据技术也有了很大的发展. 随着网络的发展，以前所谓的单一数据中心的概念不复存在，数据可能分布在网络的任何节点，如何通过地理信息元数据技术使用户在浩如烟海的信息海洋中发现和使用需要的数据变得十分迫切；还有，从历史的角度看，地理信息系统已经积累了大量的不同格式的数据，如何利用地理信息元数据技术有效地管理数据并且帮助实现地理信息数据共享也是一个很重要的问题.

目前，人们对于地理信息元数据的研究已广泛开展，并形成了多个标准，其中主要有美国联邦数据委员会（Federal Geographical Data Committee,

收稿日期：2003-06-30；修改稿收到日期：2005-02-26. 本课题得到国家自然科学基金（60203002）、国家“八六三”高技术研究发展计划项目基金（2002AA134030）、高校博士点基金（20020001015）和北京市自然科学基金（4012007）资助. 罗英伟，男，1971 年生，博士，副教授，主要研究方向为分布式计算、软件 Agent 及 GIS. E-mail：lyw@pku.edu.cn. 汪小林，男，1972 年生，博士，副教授，主要研究方向为构件技术、软件 Agent 及 GIS. 许卓群，男，1937 年生，教授，博士生导师，主要研究方向为科学计算可视化、并行计算、人工智能及 GIS.

FGDC)①的地理信息元数据标准和国际标准化组织地理信息技术委员会(ISO Technical Committee of Geographic Information /Geomatics, ISO/TC211)②的地理信息元数据标准,这两个标准都从不同侧面对地理信息数据进行了比较详细的描述.而我国的地理信息元数据标准的制定工作也正在积极展开,目前已有"中国可持续发展信息共享元数据标准"、"NREDIS 信息共享元数据内容标准草案"等.然而正如 FGDC 的用户手册所说:地理信息元数据标准只是描述了为了实现数据维护、数据共享、数据查询和数据传输等功能,系统需要有什么内容的元数据,而如何在计算机上组织和管理这些元数据信息,如何在数据传输的过程中组织数据,如何将数据提交、传输或呈现给用户是系统实现时需要考虑的问题.

XML 技术是由 W3C(The World Wide Web Consortium)③提出的 Internet 上最新的软件技术之一,它以其结构化、可扩展性及灵活性的特点引起了业界的普遍关注.XML 语言不仅可以用于数据的精确表达,而且可以为异构信息系统之间的数据交流提供有效的方法.

本文首先对地理信息元数据的内容从不同角度进行了研究,然后设计了一个基于 XML 的地理信息元数据系统 GeoMeta.

2 地理信息元数据内容及其 XML 表示

2.1 地理信息元数据内容

我们认为,地理信息元数据信息应该由描述数据自身的元数据信息和描述数据之间关系的元数据信息构成.数据自身的元数据信息主要是指那些只与数据本身相关的一些信息,比如:数据标识信息、数据的生成时间、数据质量信息和数据发布信息等.而关系元数据则描述了各个数据是通过什么样的关系构成了一个有机的整体,比如最基本的组成关系.就像汽车工业一样,信息产业也有自己的零件加工部门(提供基础数据的部门)、各级的装配部门(提供数据增值服务的地方).就一般的消费者而言,他们看到的、关心的主要是加工后的产品:汽车(具体的应用系统中的数据),而具体的部件及其组成信息(一些基础数据)是他们所不关心也没有能力了解的.但是对于汽车工程师或修理工而言,可能每个部件及部件间关系的详细信息(数据的自身信息和数据间的关系信息)都是需要详细了解的.这就需要汽车行业在各个层次上都建立产品说明标准,才能满足他们不同层次的需要(这也意味着系统需要不同层次的元数据).

如图 1 所示,一个开放式的数据系统不仅应该描述数据自身的信息,还应该描述数据之间的关系信息.这样才能使用户更好地理解系统,从而才能更有效地使用数据.如果只是孤立地说明各个数据的信息,系统的可用性可能会大大降低.

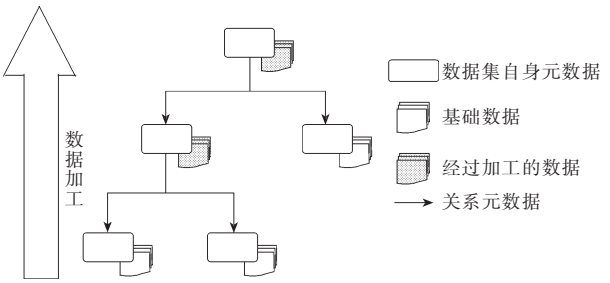


图 1 开放式数据系统中的元数据

由于元数据的内容涉及到了各行各业的各个方面,而且发展标准是一项非常艰难和得不到回报的工作[2],加之随着时代的发展,瞬息万变的信息社会给标准的制定添加了许多新的色彩,所以元数据的标准的制定是一件很困难的事情.可以肯定,不可能出现一个能让大家都遵循的元数据标准.即使在某个行业内有了这样的标准,由于大家在具体组织数据时有着不同的应用背景,在最终的实现系统中元数据内容上并不一定和标准相一致.特别是如果要描述上面说的数据间的关系信息,要定义标准是很困难的.基于这种情况,我们认为,可以根据元数据的规范程度(也就是共识程度)对元数据进行分类,随规范程度递减分为标准元数据、共识元数据和用户元数据(它们之间并没有明显的分界),如图 2 所示.

(1)标准元数据.它是按一些权威的标准化组织

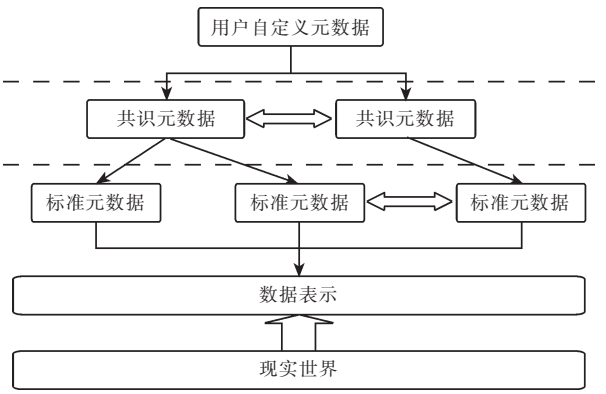


图 2 元数据层次示意图

① http://www.fgdc.gov
② http://www.isotc211.org/
③ http://www.w3.org/

制定的元数据标准(如：FGDC，ISO/TC211 等)得到的元数据. 由于经过多年的研究与实践，花费了很多的人力和物力，这些标准都比较完备和规范. 他们一般都是基于特定的行业背景提出，然后根据实际需要不断地修改和扩充，目前在自己的应用领域中都取得了广泛的承认与支持. 现在元数据标准的发展趋势是为了让自己的标准更加全面，在制定更新的标准时既得跟上时代的发展又要互相参考借鉴，结果最后虽然不可能大家标准都统一，但肯定可以在某些特定的行业上很大程度地取得一致，以致大家的元数据记录能够方便地互相转换.

(2)共识元数据. 虽然对有的信息大家在行业中已经有了相当的共识，但在上面的元数据标准中并没有提到. 比如说在目前的大多数 GIS 产品中都有图层和地图(或相当于地图)的概念，而在实际应用中，FGDC 定义的元数据标准主要是针对图层的特征来定义的，故而元数据的查找结果也是以图层为单位的. 但一般用户更熟悉的是地图的概念，他们中大部分人关心的只是几个图层经过一定的配置后得到的地图是什么样的，并不关心组成地图的具体图层的信息. 而我们就可以在 FGDC 的基础上定义地图的元数据信息，虽然由于没有相关的标准，大家在语法层次上的要求可能有较大的差异，但在语义层次上是基本一致的，故而两者之间的相互转换也是可能的.

(3)用户自定义元数据. 在实际应用中，用户可能会根据具体的需求将一系列的基础数据按某种方式组织起来或者再加入一些其它信息得到新的数据. 在描述这种数据的元数据时，除了作为数据的一些固有的属性(如数据生成时间、摘要信息、关键字等)可以用某种元数据标准来描述，其它属性都是很随意的，不可能有大家都承认的标准.

2.3　元数据的 XML 表示

图 3 是利用 XML 来表示各类元数据的示意图. 对于标准元数据，常见的元数据标准大部分都有基于 SGML(ISO8879：1985)的 DTD 格式. 由于 XML 是 SGML 的一个子集，因此将 SGML 的 DTD 转换为 XML 的 DTD 或 XML Schema 是非常方便的. 而对于共识元数据和用户自定义元数据，只要清楚元数据各个数据集之间的关系信息以及数据集自己的信息，就可以用 DTD 或 XML Schema 表示出来. 具体如何表达，可以参看 W3C 的相关文档，关于这方面的资料也很多，这里就不具体介绍了.

RDF 也是 W3C 提出的一种用来描述网上资源的元数据描述框架，而 RDF Schema 可以定义 RDF 所须的标签词汇. 通过 subClassOf 和 subPropertyOf 等概念，RDF Schema 不仅可以描述对象类之间的派生关系，也可以描述属性成员上的约束关系. RDF Schema 可以非常灵活地描述数据集间的关系，所以比较适合用来描述不易规范的用户自定义元数据.

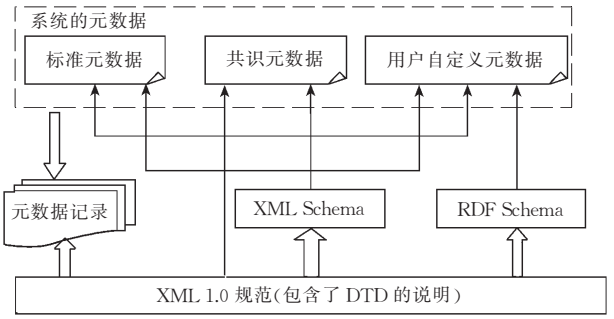


图 3　用 XML 表示元数据的示意图

3　地理信息元数据系统的设计与实现

3.1　元数据系统的体系结构

在参考了 FGDC 推荐的用于建设 ClearingHouse(空间信息交换中心)的 I-Site 系列软件①和由 Blue Angel Technologies 公司开发的 MetaStar 系列产品②后，我们设计了一个基于 XML 的元数据系统——GeoMeta 系统，其体系结构如图 4 所示. 它主要包括服务器端的 GeoMeta 服务器、元数据管理器、查询处理器和系统配置工具、客户端的元数据查询工具和元数据管理工具. 为了方便用户使用，服务器还提供一个网关使得用户可以通过 WWW 网在浏览器中对系统进行查询和管理. 一个元数据系统要真正实用还必须支持 Z39.50 协议③，因为目前大部分的元数据系统都以 Z39.50 协议作为信息查找和获取的协议. 所以 GeoMeta 还利用 I-site 提供的 API 开发了 Z39.50 查询处理器处理来自 Z39.50 客户端的请求.

3.1.1　服务器端

(1)GeoMeta 服务器. 它是元数据服务器对外的接口. 主要负责接收来自客户端的按照 GeoMeta 通信协议组织的消息，经过解析后调用相应的功能模块，然后将结果组织好后返回.

---

① http://www.fgdc.gov/clearinghouse/clearinghouse.html
② http://www.fgdc.gov/metadata/toollist/metastar.html
③ http://www.niso.org/z39.50/z3950.html

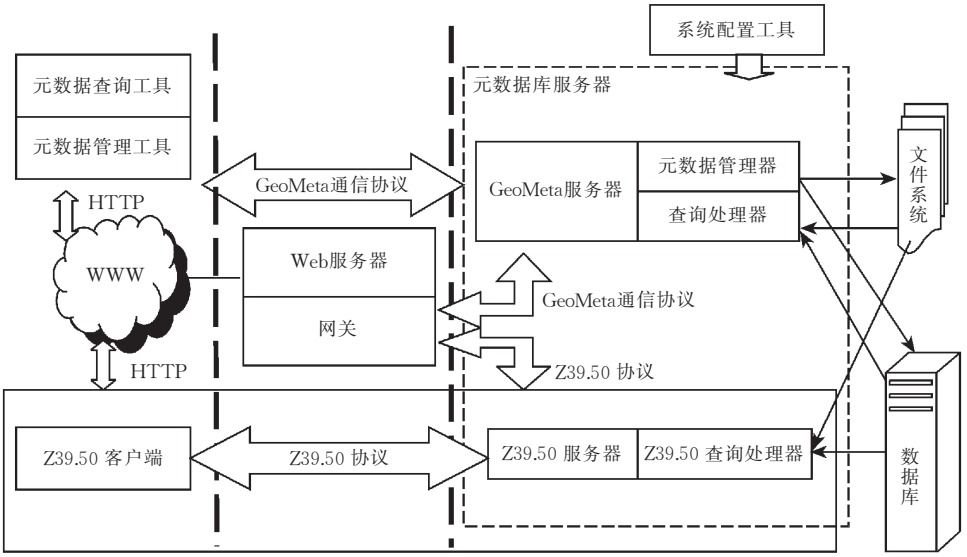


图 4 GeoMeta 系统体系结构图

(2)元数据管理器. 它负责管理各种元数据(这里是泛指各种标准元数据、基于共识的元数据和用户自定义元数据)的模式信息文件(DTD 或 XML Schema)以及它们的元数据记录,还有一些映射文件. 用户可以通过元数据管理工具来编辑元数据的 DTD 和 XML Schema,然后加入到系统中来. 还可以通过元数据管理工具来建立一些元数据之间的映射关系以及各种数据的输入输出格式与系统元数据之间的映射关系,这些关系可以用 XSL 文件来表示. 有了这些映射关系元数据管理器就可以很方便地导入和导出不同格式的元数据文件. 当然它还负责元数据记录的添加、删除和修改.

(3)查询处理器. 负责处理由 GeoMeta 服务器传过来的各种查询要求,根据用户的需要可以支持最简单的关键词查询,也可以支持结构化或半结构化的带限制的 XQuery 查询语句. 通过配置还可以让查询处理器调用其他的搜索引擎(目前只局限于关键词查询).

(4)Z39.50 服务器. 它负责接收来自 Z39.50 客户端的查询请求. 将请求重新组织后交给 Z39.50 查询处理器,得到结果后将其按用户要求组织,返回给客户端.

(5)Z39.50 查询处理器. 它根据用户的查询请求,参照系统的检索属性映射表和系统的实际存储策略来构造查询语句,得到结果后返回给 Z39.50 服务器.

(6)系统配置工具. 它负责配置元数据库服务器的系统参数,比如 GeoMeta 服务器的监听端口、文件管理目录等等. 还负责维护为支持 Z39.50 协议而保存的检索属性映射表和属性与元数据元素的对应关系.

3.1.2 客户端

(1)元数据查询工具. 查询工具由用户界面模块和协议处理传输模块构成,可以是基于 GeoMeta 通信协议的应用程序、嵌入式构件(COM)或 JavaApplet. 由于系统所支持的查询语言不是一般用户能书写的,所以要由用户界面模块来负责与用户交互,输入查询条件和呈现查询结果. 而协议处理传输模块负责将用户界面模块收集到的查询参数组织成查询语句,再根据 GeoMeta 协议产生系统消息,通过 TCP/IP 协议发送给 GeoMeta 服务器. 接收到返回结果后拆包,将查询结果交用户界面模块显示. 如果是在浏览器上做查询界面,只需在 HTML 的表单中收集到查询的参数,发送给 WEB 服务器就可以取得以 HTML 格式返回的数据了.

(2)元数据管理工具. 在结构上与元数据查询工具相同,主要是要提供给用户一个友好的界面来管理元数据模式信息、元数据记录以及各种映射关系,包括增、删、改和显示. 它也是既可以走 GeoMeta 通信协议也可以通过浏览器来处理. 不同的是它需要先登陆,才能进行权限内的操作,还需要对前面的操作有所记录,以便做日志或支持取消操作.

(3)Z39.50 客户端. 它帮助用户构造基于 Z39.50 协议的查询并呈现返回结果. 目前已经有很多这样的免费工具.

3.1.3 协议部分

(1)HTTP 协议. 为了让用户能通过浏览器来进行各种操作,系统必须支持 HTTP 协议.

(2)GeoMeta 通信协议. 这是我们自己设计的基于 XML 的通信协议. 协议用一个 XML Schema 文件描述了不同种类的服务以及服务参数和返回结果. 这样就可以由 GeoMeta 服务器统一接收消息，经过解释器处理以后发送到相应的功能模块. 这样系统服务的添加和修改就显得比较简单，只需双方都有该服务的那部分 Schema 就可以通信了，而且由于消息是纯文本格式，还避免了不同平台的编码问题.

(3)Z39.50 协议. 它是一种在客户/服务器环境下计算机与计算机之间进行数据库检索的通信协议. 它的出版及使用解决了不同系统间的数据交流的问题，克服了信息检索网络化的障碍. 需要提及的是，在 1992 年以后的 Z39.50 标准中增加了对空间数据查询的支持，使得空间信息的查询可以通过 Z39.50 协议进行①.

3.1.4　网关

包括 GeoMeta 网关和 Z39.50 网关. 它在某个意义上相当于客户端的协议处理传输模块的工作，主要功能是将用户通过 HTTP 协议传过来的一系列参数转化为满足协议的系统消息，然后发送到服务器，最后将结果以 HTML 格式的形式返回给浏览器. 我们使用 CGI 程序来做 GeoMeta 网关，而 Z39.50 网关已有现成的免费软件.

3.2　元数据的存储策略

目前在系统中用 XML 来描述数据记录传输和表达的格式是大家一致接受的. 但是应该如何来有效地存储 XML 数据以方便查询却众说纷纭，这也是当前的一个热点话题.

有一点需要明确：虽然 XML 文件有诸多优势，但并没有表现在存储上. 尽管我们希望数据存储和表达能有比较好的对应，但在真正有效的 XML 数据库出现以前，任何存储策略只需通过接口服务程序使得在用户看来系统的数据是按 XML 的模式来存储的，并且可以将满足这些模式的 XML 数据文件存入系统，同时可以要求结果按某种 XML 模式返回就可以了(当然能支持 XQuery 更好). 以下是几种比较典型的存储策略.

3.2.1　无结构的存储

将元数据文件直接存到文件系统或数据库的长二进制域段中. 在 Oracle8i 以上版本中还可以用 CLOB(character LOBs)、BLOB(binary LOBs)或者 BFILE(binary files)来保存文件. 这样做对于数据文件的存取十分简单有效，但对查询的支持比较有限. 如果文件很多的话，基本上只能通过关键字检索. 在查询要求不高或数据本来就无结构时此方法比较有效. 目前有很多支持文件全文检索的搜索引擎，而且数据库对全文检索也有了比较好的支持. 在 SQL Server7.5 以后，就有了全文检索(full-text index)的支持. 用户在表域段上建立索引后，就可以在 SQL 语句中做关键字查询了. 而 Oracle 数据库通过 interMedia text 工具也可以在 BLOB 域段上建索引. 值得一提的是，它还支持部分的检索. 比如说通过如下查询语句“Select * from purchaseXMLTab where Contains(po_XML,“street WITHIN addr”)>=1”，它可以查到〈addr〉〈/addr〉两个标签之间“street”出现次数大于等于 1 的记录项. 这样，如果用户能确定关键字的标签，查询就更准确快捷了.

3.2.2　基于线性域段结构的存储

一般来说，元数据从逻辑结构上讲都是树状的. 如图 5 所示，如果没有可重复多次的节点，可以利用节点标签的唯一性(如果不唯一的话还可以通过该节点的 XPath 来做标记)将整个树展开来存储在一个关系表中，取数据的时候再恢复过来. 但如果 XML 模式中有选择结构的话，可能会造成空间的浪费(因为关系表必须为每个可能的节点分配空间). 在实际应用中，一般只需将用户检索时用到的节点提出来存放到关系表中并且建立索引，把源文件直接按无结构的存储策略存放就可以了. 从 3.3.2 节中 Z39.50 查询服务的实现中，可以了解到，大部分的 Z39.50 服务器在处理存放在数据库中的元数据时采取的都是这种策略. 一般来说，如果需要返回的部分是整个记录或者记录的大部分内容，并且可能出现在查询条件中的元素没有重复结构时，这种存储策略还是比较有效的.

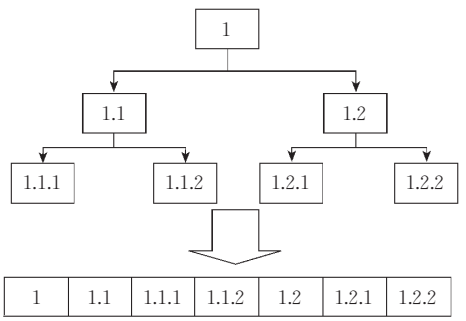


图 5　树形结构在关系表中的存放

① Use of Z39.50 to Search and Retrieve Geospatial Data. http://www.fgdc.gov/publications/documents/clearinghouse/dlipaper395.html

3.2.3 基于结构的存储

基于结构的存储大致有两种方式：

(1)先周游整个 Schema 结构，找出所有可能的 XPath，然后将数据文件的元素按不同的 XPath 存放它们的数据，必要时可将它们再组合起来. 它对系统的要求不高，用文件系统或关系数据库都可以，而且可以比较好地支持 XQuery 查询，就是实现起来比较麻烦.

(2)利用对象关系数据库来存储 XML 数据. 随着 XML 技术的广泛使用，越来越多的数据通过它来传输和表达，利用对象关系数据库来直接存储 XML 数据将是一种自然的选择，因为 XML Schema 规范和对象关系数据库的表示方法有很多可以对应的地方. 各大数据库厂商都紧密关注 XML 的发展，纷纷推出帮助用户使用和管理(不单单是存储) XML 数据的产品. 因为面向对象关系数据提供了复杂类型和面向对象的一些支持，比如：对象类型、对象类型的集合、对象类型的引用、继承. 从而使我们可以在一定程度上结构化地存储 XML 数据文件.

3.3 元数据的查询

在元数据系统中存储策略的选择往往是由实际要求来决定的，而其中最关键的因素是对查询的支持.

3.3.1 非结构化查询

直接利用数据库的全文检索功能，在保存元数据文件的域段上建立 interMedia 文本索引，通过“Contains(XML 文件所在域段，‘关键字’)”来查找就可以了. 如果用户对返回的格式有所要求，可以根据某个映射文件将其转换后再返回.

3.3.2 结构化查询

结构化查询是系统中的重点. 查询可能是 Z39.50 客户端传过来的，也可能是用 XQuery 表达的.

对于 Z39.50 客户端的查询，可以直接基于I-Site 软件提供的 API 接口来实现. 需要做的工作主要是根据接口传入的一个二叉查询树结构来查询数据库中的元数据记录. 一般的元数据库系统用的是基于线性域段结构的存储，根据查找相关的注册属性集中用户属性和相关操作对应的编码以及所用的属性表，就可以构造一条带有选择字段、查询的表名和条件子句的完整的 SELECT 语句. 将它提交给数据库，就可以得到查询结果了. 由于系统是利用对象关系数据库来存储元数据记录，就需要将原来的形如“4,1016 Title”的属性编码对应加上该属性在对象关系表中的位置“4,1016 Title metadata. section1. Title”，这样就可以构造基于对象关系模型的 SQL 语句了. 查出数据以后，将其转换为客户端定制的格式，就可由 I-Site 服务器自己打包发回了.

而对于结构化的 XQuery 查询(不好说一个 XQuery 查询是结构化还是半结构化的，我们姑且认为能被转换成基于对象关系模型的 SQL 语句的查询为结构化的查询)，系统的策略是将其用 SQL 语言来表达.

可以看出对于 XQuery 所要求的形如图 6 所示的 XML Query Data Model，对象关系数据库的对象关系模型与它有很好的对应：其中对象类型或表对应于元素节点，其域段对应于属性节点，而文本节点对应的是以文本方式存储的域段(可能是一个 XML 文件).

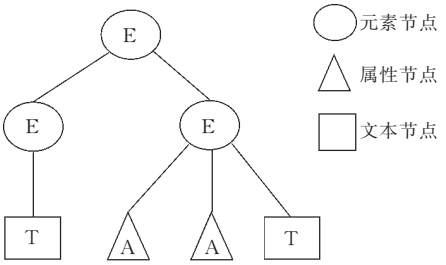


图 6 一个数据查询模型的实例

由于 XQuery 是一种基于表达式的查询语言，下面将描述几种基本表达式是如何对应的.

(1)XPath 表达式. XPath 一开始是在 XSLT Script 和 Xpointer 中做导航用的，在 XQuery 中一个 XPath 表达式代表了原文件的 Model(注：本节中将用 Model 来代表 XML Query Data Model)中的一个或多个有序的节点(包含它们各自的子节点). 显然这些节点自己也是一个 Model 的实例. XPath 具有很强的表达能力，很多情况下一个 XPath 表达式就可以代表一个查询. 如一个 XPath：document(“zoo. XML”)/chapter[2]//figure[caption＝“Tree Frogs”][①]，它就包含了 SQL 查询中的 3 个重要部分：①域段选择：Chapter[2]. figure；②查询表：zoo. xml；③查询条件：figure 的属性 caption＝‘Tree Frogs’. 相应的 SQL 语句为“select zoo. chapter[2]. figure from zoo where zoo. chapter[2]. figure. caption＝‘Tree Frogs’”.

需要指出的是，chapter//feature 指的是 chapter 的后代节点 feature，所以 feature 不一定是 chapter 的子女，也许要用 zoo. chapter. paragraph. figure 才能表达 feature. 一般情况下，只要“//”后面的标签能用 a. b 的形式表示，就可以用 SQL 来对应，否则

① XQuery. http://www.w3.org/TR/2001/WD-xquery-20010215. 本文图 7 也出自于此.

就只能按半结构化查询来处理.

目前 Oracle9i 对 XPath 表达式有一定支持，可以通过两个操作 extractNode 和 existNode 来使用 XPath. 通过 extractNode 可以从 XML 文档中通过指定 XPath 来提取对应节点：select extractNode (e. po_XML_column，'chapter[2]//figure[caption="Tree Frogs"']). 而 existNode 可以用来判断 XPath 所表示的节点是否存在. 但两个操作只能针对以无结构的形式存入的 XML 文件，虽然它们已经在很大程度上提供了实现 XML 查询的策略（因为 XPath 表达式能力很强），但由于 GeoMeta 中数据是按对象关系模型存储的，且还想更好地支持 XQuery，所以并没有仅满足于此.

(2)元素构造表达式. 通过元素构造表达式可以基于原来的 XML 文件构造出新的 XML 文件. 它由开始标签、结束标签和由其它表达式得到的内容组成. 它不好直接对应到 SQL 语句，但是通过其它表达式得到的结果不难构造出该元素的实例.

(3)FLWR 表达式. FLWR(FOR，LET，WHERE，RETURN)表达式，图 7 描述了它的数据流，可以看出其结构很像 SQL 的 SELECT 语句，因而可以很自然地建立它们的对应关系. 其 FOR 部分相当于 SQL 中的 From 部分，WHERE 部分相当于 SQL 中的 Where 部分，RETURN 部分是一个可不带标签的元素构造表达式，相当于 SQL 中 SELECT 后面的结果组织部分. 虽然 SQL 中没有与 LET 的直接对应，但"LET $ a＝XPath 表达式"相当于将 XPath 表达式对应节点与变量 a 绑定，我们在 SELECT 语句中直接用 XPath 表达式对应的节点就可以了.

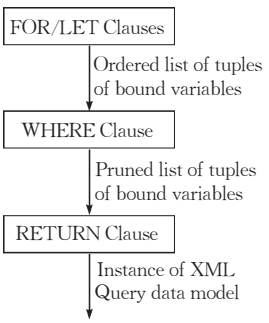


图 7　FLWR 表达式的数据流

(4)表达式的操作. 像 UNION、INTERSECT 和 EXCEPT 在 SQL 中就有直接的对应，但是 BEFORE、AFTER 就不好对应，这属于半结构化查询.

3.3.3　半结构化查询

定义 node(a)为标签为 a 的元素节点的实例. 半结构化的查询大致可以分为两类：如果查询中任何两个节点 node(x)、node(y)在判断条件中都没有涉及到它们之间的结构关系，比如 node(x)在 node(y)前出现，node(x)是 node(y)的后代等等，就属于第一类情况. 这时可以基于一定策略先找到所有这种的节点，分别判断，然后再根据它们之间的 and，or，not 关系，组合起来作为条件判断的依据. 如前面举的"要查 1950 年以后出版的一本老舍写的书"的例子，就可以用"node(author) contains("老舍") and node(pub_year)＞1950"作为查询的判断条件. 第二类情况涉及了节点间结构上的关系，一般需要周游 Model 树来判断，这就得先将 XML 文件读入内存，通过解释器将其构造成 Model 实例树，但这样做在数据量很大时效率很差.

4　结束语

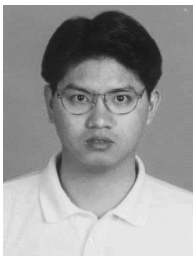
已有的一些元数据系统也已经开始支持 XML，但目前只是考虑用 XML 表示和传输元数据. 我们认为要考虑有关元数据的查询问题，必然会引起存储策略、相关协议乃至体系结构的变化. 基于这种情况，我们在设计时尽可能多地考虑在使用了 XML 来表示和传输元数据的基础上如何存储和查询这些数据. 由于 XML 技术的发展太快，不容易很快得到合适的开发工具包，同时为了简化系统忽略了一些情况（比如系统安全性，如何在客户端帮助用户方便地构造 XQuery 语句等等），所以 GeoMeta 还有很多不足之处.

GeoMeta 进一步主要的工作是：一方面是加强系统对 XQuery 的支持，特别是对半结构化查询的处理；另一方面是从系统的实用性出发，做进一步的完善工作. 从长远考虑，我们认为元数据系统将会溶入到更多的信息系统中，有一套完备的信息发布、查询、获取、使用以及付费、安全管理等体系.

参　考　文　献

1　Cheng Ji-Cheng，Li Qi，Yi Shan-Zhen. NSII and Digital Earth. Beijing：Tsinghua University Press，1999(in Chinese)
（承继成，李　琦，易善桢. 国家空间信息基础设施与数字地球. 北京：清华大学出版社，1999）

2　The Administrative Center for China's Agenda21. A Study on the Spatial Metadata Standards of China. Beijing：Science Press，1999(in Chinese)
（中国 21 世纪议程管理中心. 中国地理信息元数据标准研究. 北京：科学出版社，1999）



LUO Ying-Wei, born in 1971, Ph.D., associate professor. His research interests are distributed computing, software agent and GIS.

WANG Xiao-Lin, born in 1972, Ph. D., associate professor. His research interests are component technology, software agent and GIS.

XU Zhuo-Qun, born in 1937, professor, Ph. D. supervisor. His research interests are computer visualization, parallel computing, AI and GIS.

Background

This work is supported by the National Research Foundation for the Doctoral Program of Higher Education of China under grant No. 20020001015; the National Natural Science Foundation of China under grant No. 60203002; the National High Technology Research and Development Program (863 Program) under grant No. 2002AA134030; the Natural Science Foundation of Beijing of China under grant No. 4012007. All those projects are studied around spatial information processing, sharing and integrating, so as to realize an integrating platform for distributed and heterogeneous spatial information that possesses authors' own copyright.

The group has focused on research and development of GIS for over ten years. They studied some key technologies in spatial information science, and developed several GIS platform and tools. Also, they applied their platform and tools in many application fields, such as digital city, joint emergency response, forest, land use and so on.

This paper is aiming to study and expand the content of geospatial metadata from different aspects. Geospatial metadata are classified as standard metadata, common metadata and user-defined metadata, and the method of representing geospatial metadata by XML is analyzed. Also, an XML-based geospatial metadata system GeoMeta is designed.

国际计算机创新大会(Computer Innovation 6016)

http://www.ict.ac.cn/ci6016/

2005 年 11 月 30 日～12 月 3 日

中国·北京国际会议中心

在中国科协批准并指导下,由美国 IEEECS、ACM、IFIP 与中国计算机学会联合主办、中国科学院计算技术研究所承办的 2005 年"国际计算机创新大会(简称 CI6016)"将于 2005 年 11 月 30 日～2005 年 12 月 3 日在中国北京国际会议中心隆重召开.本次活动将向社会展示在过去的 60 年中,世界计算机硬件与软件发展的轨迹、信息化进程中的重大事件与里程碑式的创新成果,以及对未来 16 年世界计算机领域的发展趋势的预测和研讨.大会的展览和会议内容将涵盖计算机、网络、通信的应用与融合技术、网络与信息安全、未来市场需求趋势以及中国信息化教育等领域的内容.

本次大会不仅是国际范围的计算机领域的创新纪念与展望大会,同时还囊括了目前在计算机领域发展中备受业界关注的多个国际性技术会议和产业会议的内容,其中包括 HPC-Asia 2005、GCC 2005、IFIP NPC 2005 以及"第二届下一代网络发展论坛"等等国际、国内的交流活动.

大会活动内容及形式:

"国际计算机创新大会 CI6016"将以特邀演讲、圆桌会议、产业论坛、公关/新闻活动、技术会议和展览活动为特色.

·大会特邀报告

"国际计算机创新大会 CI6016"将从学术界和产业界邀请国际级专家,分享他们对计算机科技创新的历史和未来发展的真知灼见以及对计算机教育、市场和社会影响的看法.

大会安排特邀报告共 12 场,与会人数将超过 1000 人,国际会议中心第一会议室.

·展览展示活动

商业应用与研究成果展示活动将是"国际计算机创新大会 CI6016"的亮点之一.届时将大会将设置 1500～2000 平方米的展览展示空间,为国内、外软硬件供应商、最终用户、计算机技术的研究人员提供相互交流与探寻合作的场所.

·大会产业论坛

大会将安排计算机领域的产业论坛,为参与大会的企业提供展示自身最新应用技术、最新理念、最新产品的场合,并与行业用户建立起直接交流的机会.

1. IT 创新技术与应用论坛.将安排参会人数 300 人;
2. 第 2 届下一代网络发展论坛.将安排参会人数 300 人;
3. 3C 融合论坛.将安排参会人数 300 人.

·媒体招待会

媒体/公关活动作为扩大社会影响的宣传活动,大会将主持一定数量的新闻和公关活动以将计算机行业与我们的社会相连接.国内和国际媒体将被邀请参加这次活动.

将邀请超过 20 家国内外纸面、网站和电视媒体参与报道.

·技术会议

1. 第 8 届国际亚太高性能计算大会(HPC-Asia 2005) http://www.ict.ac.cn/hpcasia2005;
2. 第 4 届国际网格和合作计算会议(GCC 2005) http://www.ict.ac.cn/gcc2005;
3. 第 2 届 IFIP 国际网络和并行计算大会(IFIP NPC 2005) http://grid.hust.edu.cn/npc05/.

·圆桌会议

"国际计算机创新大会 CI6016"将主持 3 个高层会议,内容包括:

1. 计算机创新趋势会议.将安排参会人数 100 人;
2. 计算机教育会议.将安排参会人数 100 人;
3. 大众信息技术会议.将安排参会人数 100 人.