

# 基于 GML 的空间数据库设计与数据存储研究

张立朝<sup>1</sup>,吕 蓬<sup>1,2</sup>,王建明<sup>1</sup>,申正宇<sup>3</sup>,李 强<sup>4</sup>,李遵强<sup>2</sup>

(1. 信息工程大学 测绘学院,河南 郑州 450052;2. 海军出版社,天津 300450;

3. 78145 部队,云南 昆明 650210;4. 69027 部队,新疆 乌鲁木齐 830001)

## Research on Data Storing and Design of Database Based on GML

ZHANG Li-chao, LÜ Peng, WANG Jian-ming, SHEN Zheng-yu, LI Qiang, LI Zun-qiang

**摘要:**长期以来,由于 GIS 数据模型的差异所导致的诸多问题一直得不到很好的解决,迫切需要新的数据模型来构建新的空间数据库和研究其存储管理问题。如何有效存储 GML 数据是当今 GIS 界研究的热门课题。首先分析现有 XML 数据存储技术,并根据 GML 的特性,提出 GML 数据存储策略以及实现技术;另一方面,针对传统数据库技术并不适合 GML 空间数据的存储管理的缺点,对 GML 空间数据库的设计进行研究。

**关键词:**GML 空间数据库;XML;空间数据存储

### 一、引言

随着地理标记语言 (Geography Markup Language, 简称 GML) 技术的日臻成熟, GML 技术在 GIS 领域得到广泛的应用, 很多 GIS 软件商开始在他们设计的软件产品中增加了对 GML 数据的支持。在 GML 技术给我们带来便利的同时, 也给我们带来了一个重大的问题, 就是面对不断涌现的 GML 数据我们如何去管理和操纵。空间数据管理一直是 GIS 研究的主题, 它包括空间数据模型和空间数据库两方面的内容。而 GML 就是空间数据建模标准规范, 为空间数据的描述提供了标准框架。GML 是开放地理信息系统协会 OGC (Open Geospatial Consortium) 制定的基于 XML 的中立于任何厂商、任何平台的地理信息编码标准, 用于地理信息的传输、存储和发布。OGC 先后推出了三个版本的 GML, 其应用和研究还处于起步阶段。目前绝大多数空间数据还是存储在传统的关系数据库或对象关系数据库中, 各 GIS 软件厂商只是用 GML 作为传输和交换的中间文件格式。在发送方需要专门的转换软件来将发送方的 GIS 文件格式转换为 GML 格式, 在接受方也需要专门的转换软件来将 GML 格式转换为接受方的 GIS 文件格式。GML 不仅可以作为一种有效的空间数据传输、交换的手段, 而且也是一种很好的空间数据存储格式。如果使用 GML 来存储管理空间数据, 即各 GIS 软件开发商都使用 GML 作为其数据模型

和文件格式, 那么就不需要任何转换, 所有 GIS 的空间数据都可以有效地集成与共享。关键是如何有效地存储管理 GML 空间数据。

由于 XML/GML 半结构化 (树状结构) 的数据模型与传统的关系模型之间的重大区别, 传统的关系数据库并不适合存储管理 XML/GML 数据。随着 XML 技术的快速发展, 已经涌现出了一批专门用来存储管理 XML 数据的数据库系统 (Native XML Database, NXD), 这些系统能够很好地实现 XML 数据的存储管理、查询检索等功能, 为 XML 数据的存储管理提供了有效的解决方案。由于 GML 数据是基于 XML 标准的空间数据, XML 的所有技术都可以应用于 GML, 人们很自然地想到能否使用 XML 数据库系统来存储管理 GML 空间数据? 若能用 XML 数据库系统来存储、传输 GML 空间数据, 那么就不需要专门的数据格式转换软件, 因为这种本原 XML 数据库存储、传输的就是 XML 数据, GML 数据也是一种 XML 数据。但是 GML 并不等同于 XML, GML 技术与 XML 技术的区别正如 GIS 技术与 MIS 技术的区别, GML 空间数据库与 XML 数据库的区别正如空间数据库与一般数据库的区别。所以需要进一步研究 GML 空间数据的存储管理问题。

### 二、GML 的研究现状

目前国内外对 GML 空间数据的存储管理研究较少。文献 [3] 对 GML 空间数据在三种关系数据

库中的存储和查询(SQL)性能进行了比较,提出了基于SQL的GML空间查询语言,由于XML数据模型与传统的关系模型之间的重大差别,扩展SQL以支持GML查询并不是GML查询系统的最有效方案。当用户需要进行GML查询或其他处理时,在GML与关系数据库之间进行来回转换要耗费相当多的处理时间,这会降低数据的处理速度。此外,这种处理方式与XML查询语言标准XQuery不相符。文献[4]比较了几种XML查询语言,并提出了对XQuery语言进行扩展以支持GML查询的设想,但文中并没有涉及如何实现等更深层次的问题。GML空间数据库的研究还有很多理论和技术问题需要解决。

目前,有的欧洲国家开始给用户发布GML数据,已不再直接提供传统的数据形式,这就迫使我们不得不研制一种能够具有传统数据库的功能,但又与传统数据库有区别的基于GML数据的新数据库。

### 三、GML数据存储技术

#### 1. GML简介

GML是一种用于建模、传输和存储地理及与地理相关信息的XML编码语言,它包括了地理要素(Feature)与层(Coverage)的空间与非空间特征。GML建立在W3C系列标准之上,以一种互联网上容易共享的方式来描述、表达地理信息,是第一个被GIS界广泛接受的元标记语言。在这之前开发的各种用于存储和交换地理信息的文件格式或标准,由于缺少相应的支持工具而没有被广泛接受。GML一个重要优势就是它能够让开发者或用户非常灵活地使用已被广泛接受的XML技术,它建立在XML,XML Schema,XLink和XPointer基础之上。GML数据能够很容易地与非空间数据集成使用。GML使用XML对地理数据进行编码,为开发商和用户提供了一种开放的、中立于任何厂商的地理数据建模框架,使地理信息能够在不同领域、不同部门进行语义共享。

GML采用要素(Feature)来描述地理实体和地理现象,GML要素是通过其属性(Property)来描述的,这些属性包括空间属性、时态属性及其他非空间时态属性。

OGC先后推出了三个版本的GML。GML1.0版本于2000年4月正式推出,GML2.0版本于2001年2月推出,GML2.12版本于2002年9月推出,2003年1月推出GML3.0版本。3.0版增加了对复

杂的几何实体、拓扑、空间参照系统、元数据、时间特征和动态要素等的支持,使其更加适合描述现实世界问题。GML作为一个空间数据编码规范,它并没有强制采用它的用户使用规定的XML标识,而是提供了一套基本的标签、公共的数据模型,以及用户构建应用模式(GML Application Schemas)的机制。

#### 2. GML空间数据的特征

和传统空间数据相比,GML空间数据具有如下特征:

1. GML遵循OGC所制定的地理抽象模型,该模型已得到了大多数GIS软件厂商及第三方软件厂商的支持,因此数据在转换成GML时不会有信息的损失。

2. GML是基于文本的地理信息表示,比较简单、直观,容易理解和编辑,使用一般的文字编辑软件或专门的XML编辑器即可阅读和编辑,并不依赖任何GIS软件。

3. GML空间数据的有效性控制,GML模式定义了GML文档的内容和结构,通过GML模式可以在数据编辑及传输时验证其规格是否有效。

4. GML容易与非空间数据集成,传统的二进制文件必须了解其数据结构,才能进行编辑修改,而且数据结构修改时,应用程序也必须跟着修改,故不易与其他系统的数据整合应用。而GML可通过XLink,XPointer或URI与其他XML数据链接。

5. 可将GML数据转换成任一向量数据格式(例如SVG,VML或VRML),即可在任意浏览器上显示,不需要安装特别的图形插件。

6. GML是基于XML的地理信息表示,各种XML技术如XML解析,XML Schema,XLink,XPointer,XSL,XML查询语言等都可应用于GML。

7. GML封装了地理信息及其属性,GML地理要素(Feature)包括一系列的属性和相应的几何信息、拓扑信息,一般来说属性由名字、类型和值组成,几何信息由基本元素如点、线、面、曲线、多边形等组成,拓扑信息由拓扑基元Node,Edge,Face,TopoSolid以及这些基元之间的关系来描述。GML允许构建相当复杂的要素,如要素间的嵌套。

8. GML封装了空间地理参考系统,空间地理参考系统是地理信息系统数据处理的基础。GML封装了空间地理参考系统、主要的投影关系等,保证了分布式处理的扩展性和灵活性。

9. GML可以实现空间数据的分布式存储,GML对空间数据的分发是非常方便的技术手段,但其作用不局限于此,同样可以成为空间数据分布式存储

的重要手段。

10. GML 不仅可以表达基于要素模型的矢量数据,还可以表达基于场模型的栅格数据。

### 3. 基于 XED 的空间数据存储

考虑到对象关系数据库存储空间数据的必要条件以及空间数据库的可操作性,如便于查询数据,本文设计了一种中粒度方法解决空间数据库的存储问题。空间实体有两个最基本的要素:空间要素和属性要素。实践证明:如果将空间要素和属性要素分开存储在两个表中,数据库在最终进行查询时将会耗费大量时间来进行表的连接,可操作性较差。由于一个 GML 文件可能包含多个图层信息,本文考虑将同一图层的数据都存储在同一表中,表分为两部分,分别表示空间要素与属性要素。每个 GML 文件都有一个对应的 XSD 文件用来解释此 GML 文件的结构,通过解析该 XSD 文件动态设计表结构。属性要素由分析 XSD 文件得来,空间要素包括两部分:一个是表示此空间要素的外接矩形,另一个是组成此空间要素的点集。

它的主要缺点在于:首先,由于 XML 数据是一种半结构化数据,而关系数据库管理系统都是面向结构化数据的,在两者之间进行数据的转化必然要丢失一定的信息;而且为了适应 XML 文档表示方式的随意性,必定要产生大量的冗余数据,这就破坏了表结构的紧凑性,也浪费了存储空间;其次,由于 XML 文档具有结构多变的特点,结构上的每一次变化都可能引起所对应数据库管理系统表结构的改变,这对现有数据库系统的性能产生很大的影响,不符合对结构化数据进行管理的特点;此外在对 XML 数据进行增删操作时,这种 XML 数据的管理方法会因数据的变化引起结构上的变化,使数据表结构要作相应的改变。当遇到大型或复杂文档时或当用户需要进行 XML 查询或其他处理时,在 XML 与数据库之间进行来回转换要耗费相当多的处理时间,数据库需要重新组合这些数据,这降低了数据处理的速度。

### 4. 基于 NXD 的空间数据存储

NXD 支持“文档集合”的概念,通常它提供两种模式,即有模式和无模式。前者是一个文档集合关联一种模式,当将 GML 文档加入到有模式的文档集合时,NXD 会对该 GML 文档进行模式检查,只有符合模式的文档才能添加到该集合中。当 NXD 使用“无模式”文档集合时,则更方便用户操作,灵活性大。本文假定所有要存储的空间数据均以 GML 格式进行编码,不存在用户存储格式不同的问题,故采用有模式的文档集合存储 GML 文件。

## 四、GML 空间数据库设计

### 1. XED 数据库模式

Table 表包含 ID,属性要素  $1 \cdots n, x_1, y_1, x_2, y_2$ , 空间要素点集。其中属性要素需要解析 XSD 文件后存入下面的数据结构中,待动态建表用。 $x_1, y_1, x_2, y_2$  四个字段表示空间要素外接矩形的左上角和右下角坐标,保证了在空间分析查询中系统的实时性和有效性。由于对象关系数据库中的记录存在无次序、无层次的问题,需定义一个 ID 标识每一条记录,显然对于初始记录指定为 1,依次自增。而空间要素点集通常采用 LOB (Large Object) 大对象存储空间对象的图形数据。LOB 以两种形式存储在数据库中,一般大型数据库都提供了相应的数据类型存放 LOB 数据。一种是以 BLOB 形式存储,比较典型的有 Access 数据库中 OLE 对象数据类型、SQL Server 数据库中的 Image 数据类型、Oracle 中的 LONG RAW 和 BLOB 数据类型等;另一种是以 CLOB 形式存储,比较典型的有 SQL Server 数据库中的 Text 和 Ntext 数据类型、Oracle 中的 CLOB 和 NCLOB 数据类型。

目前,对 XML 文档进行读、写等操作主要有三种方式:文档对象模型、Net Framework 提供的 PULL 模式、DOM 和 SAX。第一种模式相对于后两者优势明显,它按照只向前、无缓存的方式遍历 XML 文档,不但快捷、高效,且易于使用。图 1 给出了该方案的 GML 入库流程图。

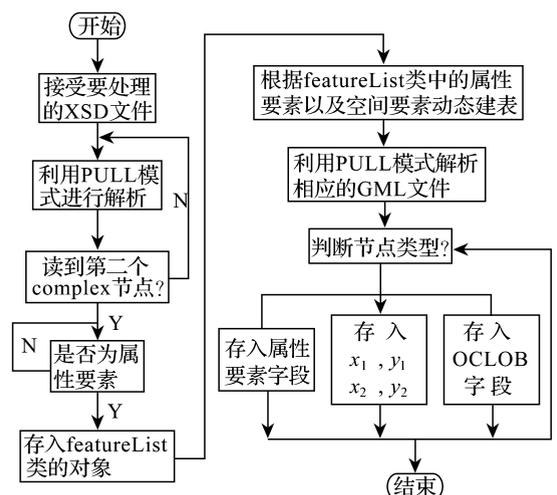


图 1 GML 入库流程图

### 2. NXD 数据库模式

本文以 Tamino 数据库为例,讨论 NXD 数据库的存储机理。Tamino 丰富的 SDK 工具为开发人员

提供了必需的开发接口,无论采用 Java 平台还是 .NET 平台都有相关支持。Tamino 所支持的开发语言主要有:Java, C/C++, JavaScript 和 C#等。本文采取 Tamino API for .net 将空间数据存入数据库。

Tamino 是纯 XML 数据库,对 XML 数据的格式要求不仅完全遵循 W3C 的规范,还对 Schema 文件有特殊的格式要求。Schema 文件格式要按照一定的格式才能被 XML Parser 识别。GML 文档入库分两步:GML Schema 入库,Schema 对应的 GML 文件入库。Schema 入库可以通过 Tamino 的 Manager 来进行,也可以通过 Tamino X - Plover 来进行,对应的 GML 文件入库流程图如图 2 所示。

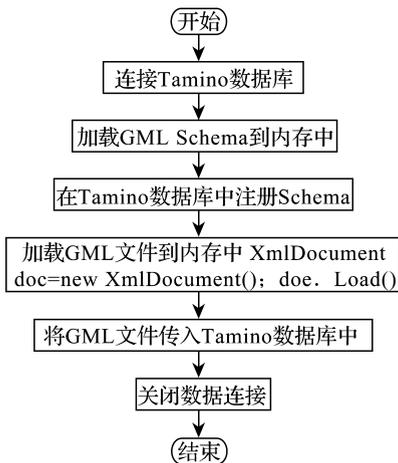


图2 GML 存入 Tamino 算法流程图

### 3. XED 和 NXD 性能比较测试

本文对两种数据库存入速度、查询速度和 GML 文件大小与复杂程度进行了测评。测试结果表明:存储结构简单、数据量小的 GML 文件,XED 和 NXD 的速度相近,但是随着 GML 文件增大以及结构复杂度提高,NXD 的存储优势明显提高。随着 GML 文件增大,利用空间数据库查询 GML 文件的速度是用

DOM 查询的 2~6 倍,而 XED 的查询速度是 NXD 的 3 倍。从常识来说,对数据库进行查询操作要比入库操作频繁,所以即便 NXD 的入库速度比 XED 快,二者仍各有优势。

## 五、结束语

本文从 XML 数据库的角度,对 GML 空间数据库系统和 GML 数据存储方法进行了全面、系统的研究,由于 GML 数据的存储研究还属于一个崭新的领域,目前还没有一个比较完美的技术实现方法。目前国内外有不少 GIS 界的学者正在致力于这方面的研究,相信在不久的将来将会出现具备更完善数据库功能的 GML 数据库系统。

## 参考文献:

- [1] BOURRETR. XML and Database [EB/OL]. <http://www.rpbourret.com/xml/XMLAndDatabases.htm>, 2005-05-06.
- [2] HOGAN M. XML: The Foundation for the Future [EB/OL]. [http://www.xml.org/xml/xml\\_foundation-future.shtml](http://www.xml.org/xml/xml_foundation-future.shtml), 2007-08-11.
- [3] 司功闪,王鸿谷,徐 婕. 以 XML 为核心的 Tamino 数据库的研究与分析[J]. 计算机工程, 2004, 30(16): 78-79.
- [4] 刘 源,姚淑珍. XML 的关系化存储及与关系数据库的数据转换[J]. 计算机工程与设计, 2004, 25(11): 2 039-2 043.
- [5] 刘 刚,喻 成. Native XML 数据库的研究与应用[J]. 微机发展, 2005, 15(8): 65-67.
- [6] 施伟斌,孙未未,施伯乐. XML 数据的结构化处理方法[J]. 计算机研究与发展, 2002, 39(7): 819.
- [7] 曹志月,刘 岳. 一种面向对象的时空数据模型[J]. 测绘学报, 2002, 31(1): 87-92.

## 欢迎订购《测绘通报》世纪光盘

《测绘通报》世纪光盘是集成化全文电子期刊,具有按年、期、篇名、关键词(任意词)、作者、机构、全文进行检索的功能。全文收录了《测绘通报》自创刊以来至 2001 年的期刊的全部文章。

《测绘通报》世纪光盘(2 张)定价 120 元/套。外埠加 20% 邮费。欢迎读者与编辑部联系订购,联系电话:010-68531192,联系人:金英。