

应用对象关系型数据库存储 GIS 数据

罗忠文

(中国地质大学信息工程学院,湖北武汉 430074)

摘要: 在比较目前主要 GIS 存储方式的基础上,认为基于对象关系型数据库存储 GIS 数据是一种良好的选择. 讨论了对象关系型数据库系统 Postgre-SQL 的特点及其适合于存储 GIS 数据的一些优异特性,主要包括基本的空间数据类型及内置空间操作符及可扩展机制等;并讨论了基于 OpenGIS 标准扩展该数据库以实现地理信息数据存储的方法,主要包括空间数据类型的扩展及空间操作扩展;最后讨论了三维地理信息系统的扩展,在三维数据结构方面采用单纯复型作为基本数据模型,在空间的几何属性中引入几何体的包围盒,并通过 R-树等索引函数进一步提高速度. 另外还实现了空间数据操作的两类重要函数,即空间位置判断和并、交、分、割、裁剪等物体运算函数. 所有的内核函数基于 C 及 C++ 实现,而在客户端通过 JAVA 实现服务器数据的访问.

关键词: 对象关系型数据库;地理信息系统;开放数据库标准;Postgre-SQL.

中图分类号: TP311.132 **文献标识码:** A

文章编号: 1000-2383(2002)03-0267-04

作者简介: 罗忠文(1963—),男,副教授,1985年毕业于武汉大学,1990年毕业于中国科学院获硕士学位,现从事地理信息系统、虚拟现实的教学科研.

如何有效存储 GIS 数据一直是 GIS 研究的重要课题. 早期的 GIS 系统多采用文件方式存放图形数据,而用关系数据库存放属性数据,造成数据的分离,给数据管理及操作带来很大的困难. 同时用文件系统管理数据量很大的图形库也异常困难. 基于以上问题,人们提出利用关系数据库来存储图形数据,但由于关系数据库自身的特点,不能很好表达图形数据,特别是不能直接在服务器端实现对图形数据的空间查询等操作,从而不是一种好的解决方案. 近年来,随着计算机硬件技术的发展及面向对象的数据库系统的日趋成熟,使得利用其来存储图形数据成为可能.

1 面向对象数据库的类型

面向对象数据库系统的类型主要有两种:其一是对某种成熟的面向对象语言进行持久化(persistence)扩展而产生出来的,其二是在关系型数据库的基础上进行面向对象的扩展. 前者由于直接基于面

向对象的语言,在速度上更快,在功能扩展也较容易,但由于语言本身的强大功能,对后台数据稳定及一致性等方面可能造成严重破坏. 由于后者数据库内核是封闭的,在安全性及稳定性方面更好一些,但效率没有前者高. 前者的主要代表产品有: Object-store^[1]和 Jasmine. 后者的主要代表产品有加州大学 Berkely 分校最初开始研制,后由世界各地自由软件开发人员共同维护的 postgres^[2],该软件提供了分布式 C/S 结构与面向对象,完善的安全机制,支持多种客户端开发环境等功能. 目前,在成熟的二维 GIS 商业软件系统方面,还主要用关系型数据库或文件系统存储图形数据. 但在三维 GIS 理论研究方面^[3],国内外基本达成共识,即应基于面向对象的数据库来构建属性与图形数据一体化管理的三维 GIS 系统^[4,5].

2 Postgre-SQL 对象关系型数据库系统

2.1 Postgre-SQL 系统模型

基于 C/S 结构的系统模型如图 1 所示. 在以上

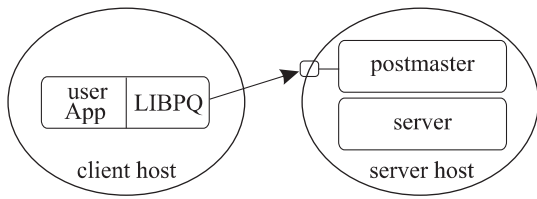


图 1 C/S 结构的系统模型

Fig. 1 Relation model of C/S structure

模型中,使用的是简单的“每用户一进程”的 C/S 模型。一个 postgres 会话由下面一些 UNIX 进程组成:(1)一个监控的守护进程(postmaster);(2)用户的前端应用(如 psql 及 pgaccess 程序);(3)一个或者多个后端数据库服务器(postgres 进程本身)。

一个 postmaster 管理某台主机上的一定的数据库集合,这个集合叫做一次安装或节点。如果某个前端想访问某个节点中的某一数据库,它就会进行库函数调用。库把用户的请求通过网络发给 postmaster, postmaster 接着启动一个新的后端服务进程,并将前端进程和这个新的服务进程联接起来。之后,前端进程和后端服务不再通过 postmaster,而是直接进行通讯。因而,postmaster 总是在运行等待请求,而前后端进程则是起起停停。在这种体系结构中,postmaster 和后端服务器总在同一台机器上(数据库服务器)运行,而前端应用程序可以在任何地方运行。

2.2 Postgre-SQL 系统特点

(1)内置的丰富的空间数据类型:点(point)、线(line)、线段(lseg)、长方形(box)、闭合及开放路径(path)、多边形(polygon)、圆(circle)。(2)内置的空间操作符有平移、伸缩、旋转、求交、求多边形顶点数、求最近点、(左、右)重叠判断、求距离、位置关系判断(左、右、上、下、包含、在同一水平(铅直)线上、平行、垂直、相同、在中心)。内置的空间操作函数有求面积、求交、求中心、圆的直径(半径)、长方形的高(宽)、开环还是闭环、对象长度、点数、封闭路径及各种几何转换函数。值得指出的是 Postgre-SQL 已内置了 R 树及 GIST 等空间索引方式,由于其扩展性还包括一些新的索引方式。(3)完善的安全机制:提供基于用户及用户组的权限控制、基于单个 IP 地址及子网的权限控制,以及对每个数据库赋予不同访问权限的机制。在具体的认证中,提供了不用密码(trust)、明码密码(password)、密文密码(crypt)、Kerberos V4 密码认证、Kerberos V5 密码认证等不同加密强度的认证方式。(4)丰富的开发语言及支持

多种操作系统平台:postgres 服务器软件可在几乎所有 UNIX 平台及 WinNT/2000 上运行,而其客户端则可在各种 UNIX 及 WINDOWS 平台下运行。(5)可扩展机制:由于 Postgre-SQL 提供了面向对象的特性,通过类型定义及表的继承,能方便地实现新的用户数据类型及函数定义,并能通过继承,方便地实现数据重用,从而为进行遵循 OpenGIS 标准的扩展及三维 GIS 数据模型扩展提供了方便。

3 Postgre-SQL 遵循 OpenGIS 标准的功能扩展

尽管 Postgre-SQL 提供了部分空间数据结构和操作函数,但仍然很不完善。目前主要的工作有 Dave Blasby 的 POSTGIS,主要是基于二维数据进行扩展的。

3.1 POSTGIS 系统的对象模型

POSTGIS^[6]基于 OpenGIS 对原来的 Postgre-SQL 进行了扩展。在扩展中遵循了 OpenGIS 标准^[7],其对象结构模型如图 2 所示。

在以上的类层次模型设计中,抽象的几何(Geometry)对象被所有其他对象所继承,任何具体对象都应定义为抽象的几何对象的一个实例^[8]。具体对象须提供对象的适当表示方式及独立的几何函数实现。举例来说,在抽象对象中定义的求包围盒函数,在具体对象(如线、曲面等)中详细实现其特点。

3.2 POSTGIS 系统的功能嵌入

在 POSTGIS 中,主要实现了 OpenGIS 所定义

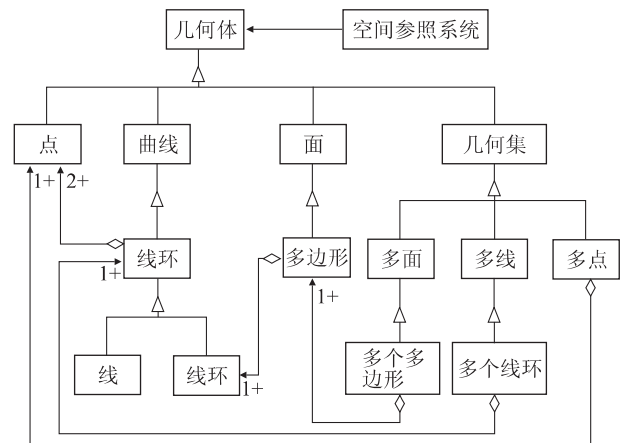


图 2 POSTGIS 对象结构模型

Fig. 2 Object structure model of POSTGIS

的一些函数,具体如下:在表中添加(删除)几何列的函数: AddGeometryColumn、DropGeometryColumn. 以 WKB (WKT)的形式返回几何: AsBinary (AsText);测试维数函数: Dimension;求包围盒的函数: Envelope;求几何类型的函数: GeometryType (geometry);求几何体中的首点的 $X[Y, Z]$ 坐标值的函数: $X[Y, Z]$ (geometry);求几何中第一条线点的个数: NumPoints(geometry);求几何第一条线段的第 N 个点: PointN (geometry, integer);求几何中首多边形的外环: ExteriorRing (geometry);求几何中首多边形的内环的编号: NumInteriorRings (geometry);求几何中首多边形的第 N 个内环: InteriorRingN (geometry, integer);判断几何的起点与终点是否重合: IsClosed (geometry);求几何集合体中几何体的个数: NumGeometries (geometry);求几何集合体的第 N 个几何体: GeometryN (geometry);求两个几何体间的距离: Distance (geometry, geometry);求(设置)几何的空间参考系统参数: SRID/SetSRID (geometry);将几何的 WKT 表示转换成几何对象: GeometryFromText (varchar, integer);求几何中的终点(起点): EndPoint[StartPoint] (geometry);求中心标识的函数: Centroid (geometry).

4 针对三维 GIS 的扩展

4.1 三维 GIS 模型的数据结构

对于三维物体笔者采用 Breunig^[9]提出的单纯复型作为三维 GIS 的基本数据模型. 基于单纯复型的模型,笔者用对象层次模型来存储三维 GIS 数据.

在以上的结构中,三维几何体(Solid)继承抽象对象 geometry 而生成,其由具体的三维单纯复型构成,三维单纯复型由多个单纯形组成. 在现实世界,笔者一般根据物体的形状来表示,但作为一种内部形式,将复杂的空间物体统一表示成单纯形的集合体较方便,原因有 2 点:(1)单纯形较简单,有统一的数学表示形式;(2)所用的单纯形相对较少,而用其他形体(如长方形及四方体)表示复杂物体则可能要更多.

在空间几何的属性中,笔者引入了几何体(集合)的包围盒. 由于在笛卡尔坐标系下,四方体的操作及判断非常快. 作为一种预判各种空间关系的方法,包围盒的引入能大大加快空间查询及运算的速度. 而基于包围盒定义空间索引,可进一步加快检索

速度,故在包围盒对象上,笔者还定义了 R-树等索引函数,以实现有效查询.

4.2 空间数据上的操作函数的实现

在三维空间函数的设计上,笔者主要定义及实现具有通用性的函数,而对一些与应用领域有关的函数,可针对具体专业应用领域的需要进行扩展产生. 具体包括 2 类:一类是判断空间位置,返回 bool 型的函数,包括:(1)包含测试;(2)相等测试. 另一类是 2 个物体进行运算,并返回 1 组物体的函数,包括:(1)三维物体的并;(2)三维物体的交;(3)用平面分割三维物体;(4)包围盒裁剪.

在三维空间运算中,运算结果可能不再属于原来的对象,例如 2 个三角形的交可能是四边形. 对此,可将其分解成 2 个三角形以保证运算的封闭性.

4.3 外部接口的实现及功能扩展

对于数据库服务器端的对象属性扩展及函数定义,是基于 C 及 C++ 实现的. 同时,也提供了面向客户端用户的接口. 这里,主要通过 JDBC,来实现客户端对数据库服务器的访问. 这样做的原因有以下 3 点:(1)JAVA 的跨平台特性,使接口能方便地提供对不同操作系统及硬件平台的支持,从而保证了软件的通用性;(2)JAVA 在 Internet 网络中的扩展功能极强;(3)后端的三维应用软件采用 JAVA 3D 的三维可视化引擎.

5 结论

本文在比较目前主要 GIS 存储方式的基础上,提出基于对象关系型数据库存储 GIS 数据,并讨论了基于 Postgre-SQL 及 OpenGIS 实现空间数据存储的方法. 通过以上途径,成功地解决了属性数据与空间数据分离的问题. 通过在服务器端扩展 SQL 语句,实现了基于数据库的空间查询. 由于 Postgre-SQL 数据库的 C/S 特性,使笔者的存储方式具备网络存储访问、完善的控制机制、分布式数据存储等一系列功能. 另外,在前人工作的基础上,还重点实现了三维 GIS 的面向对象数据结构及数据存储.

参考文献:

- [1] Excelon Corporation. Technical specification [EB/OL]. <http://www.objectdesign.com>, 2000.
- [2] Momjian B. PostgreSQL: introduction and concepts [M]. [s. l.]: Addison-Wesley, 2000.

- [3] 罗忠文, 谢忠. 三维 GIS 系统框架模型[J]. 计算机工程与应用, 2002(待刊).
LUO Z W, XIE Z. 3D-GIS structure mode [J]. Computer Engineering and Application, 2002 (in Press).
- [4] 吴信才. 地理信息系统的基本技术与发展动态[J]. 地球科学——中国地质大学学报, 1998, 23(4): 329—333.
WU X C. Basic technology and development of geographic information system [J]. Earth Science—Journal of China University of Geosciences, 1998, 23(4): 329—333.
- [5] 牛雪峰, 孙运生, 杨国东. 三维地理信息系统面向对象数据结构[J]. 世界地质, 1999, 18(3): 72—75.
LIU X F, SUN Y S, YANG G D. 3D GIS object orient data structure [J]. World Geoscience, 1999, 18(3): 72—75.
- [6] Ramsey P. Postgis manual [EB/OL]. Reflection Research Corporation. <http://postgis.reflections.org>, 2001.
- [7] OpenGIS Consortium Inc. OpenGIS simple feature specification for SQL [EB/OL]. <http://www.opengis.org/techno/specs.htm>, 1999.
- [8] 袁艳斌, 吴冲龙, 李伟忠. 面向野外地质填图的空间实体对象表达[J]. 地球科学——中国地质大学学报, 2001, 26(2): 192—196.
YUAN Y B, WU C L, LI W Z. Object styles of spatial entity for field geological mapping based on object-oriented method [J]. Earth Science—Journal of China University of Geosciences, 2001, 26(2): 192—196.
- [9] Breunig M. An approach to the integration of spatial data and systems for a 3D geo-information system [J]. Computers and Geosciences, 1999, 25(1): 39—48.

Using ORDBMS to Store GIS Data

LUO Zhong-wen

(Faculty of Information Engineering, China University of Geosciences, Wuhan 430074, China)

Abstract: Based on the comparison of the main storage method used in current GIS, we think that the ORDBMS is a good choice for storing the GIS data. We have discussed the characteristics of a special ORDBMS Postgre-SQL, which include predefined space data type and operation and extensibility for data and function, for the storing and processing of GIS data. We have further discussed the extension of the Postgre-SQL following the standard of OpenGIS, which involves a more complicated space data structure and more functions. Moreover, we have extended to the discussion on the store of 3D GIS data applicable in the real world. At last, we discuss the extension for 3D GIS. We use simple complexes as the basic data model, define bounding box in the geometry feather and using the R-tree structure to improve searching proficiency. We also implement two kinds of important function in space manipulation, one is the determiner of the spatial relation between objects, the other is manipulation between two 3D bodies such as union, intersection and clipping. All the function is designed for the extension in server side using C or C++, and in the client side, we use JAVA language to get the space data from server.

Key words: object-relation database management system (ORDBMS); GIS; OpenGIS standard; Postgre-SQL.