

基于对象代理机制的 GIS 空间数据模型研究

陈良臣¹, 王云龙², 谢卫平²

(1. 中国科学院高能物理研究所 计算中心, 北京 100049;

2. 武汉大学 计算机学院, 湖北 武汉 430037)

摘 要: 为了设计出一种基于对象代理机制的 GIS 空间数据模型, 分析了地理信息系统、空间数据和空间数据库的特点, 归纳了混合数据库模型、扩展结构模型、全关系模型、面向对象模型和对象关系模型等传统空间数据模型的优缺点, 在对象建模技术的基础上, 引入地理信息系统对象代理机制, 提出了基于对象代理机制的 GIS 空间数据模型。把对象代理机制引入到空间数据库的建模以及地理信息的组织和显示中, 完成了地理信息系统中的空间数据存储和管理的新形式, 更大程度地满足了用户的操作需求。

关键词: 对象代理机制; 空间数据模型; 地理信息系统; 空间数据库

中图分类号: TP391; P208

文献标识码: A

文章编号: 1673-629X(2010)07-0175-05

Geographic Information Modeling and Storage Management Based on Object Deputy Model

CHEN Liang-chen¹, WANG Yun-long², XIE Wei-ping²

(1. Computing Center, Institute of High Energy Physics, Chinese Academy of Sciences, Beijing 100049, China;

2. Computer Department, Wuhan University, Wuhan 430037, China)

Abstract: In order to design a geographic information system modeling based on object deputy model, it analyzes the features of geographic information system, spatial data and spatial database, and it also introduces modeling problems and deficiencies of hybrid model, extended model, integrated model, object oriented model and object relational model in the process of GIS modeling. As a result, object-deputy spatial model is put forward based on object deputy mechanism. The object deputy mechanism is introduced into the modeling of spatial databases and geographical information, it completed a new forms of geographic information system spatial data storage and management. And it also greater degree to meet the user's operational needs.

Key words: object deputy mechanism; spatial data model; GIS; spatial database

0 引言

地理信息系统的核心是空间数据, 而空间数据的关键在于如何存储和管理。随着数据库技术的发展, 对于地理信息系统的建模先后诞生了多种空间数据模型: 混合数据库模型、扩展结构模型、全关系模型、面向对象模型和对象关系模型。这些模型在一定程度上或在某一方面都解决了相应的问题, 但是它们彼此都有自身的缺陷: 有的在空间数据的定义上缺乏独立性, 有的缺乏强大的空间 SQL 的支持, 还有的由于数据模型的对象独立性特征, 使得空间叠加操作的效率低下

等^[1]。另外, 基于这些模型的地理信息都采用分层的结构, 特别是地理对象的显示使用的是图层的方式, 难以去除数据的冗余, 不能很好地满足用户的需求和相应的用户在电子地图上的操作。

针对上面的问题, 建立基于对象的数据模型是一种可行的解决途径。A. Tang、K. Borges、G. Kusters 使用并且扩展对象建模技术, 实现了对要素、要素属性以及空间关系的面向对象的表达方法。在此基础上, 文中基于 OpenGIS 规范, 引入了对象代理机制, 使用基于对象的方式来存储、管理并显示地理对象, 实现上层 GIS 的对象代理建模。

1 地理信息概述

1.1 地理信息系统

地理信息系统是在计算机软硬件支持下, 采集、存

收稿日期: 2009-09-29; 修回日期: 2010-03-18

基金项目: 中国下一代互联网示范工程 (CNGI) 移动奥运项目资助 (CNGI-04-17-2A)

作者简介: 陈良臣 (1982-), 男, 硕士, 高级工程师, 研究方向为 Web2.0、网络计算与信息安全、地理信息系统。

储、管理、检索、分析和描述地理空间数据,适时提供各种空间和动态的地理信息,用于管理和决策过程的计算机系统。地理信息系统主要由四部分组成:计算机硬件系统,计算机软件系统,空间数据及系统的组织和人员。其核心内容是计算机硬件和软件,空间数据反映了应用地理信息系统的信息内容,用户决定了系统的工作方式^[2]。

地理信息系统按地理坐标或空间位置对空间数据进行处理和管理,它研究各种空间实体及其相互的空间关系;通过综合分析,它还可以迅速地获取满足应用需要的信息,并能以地图、图形或数据的形式表示处理的结果,地理信息系统的数据流程如图 1 所示。

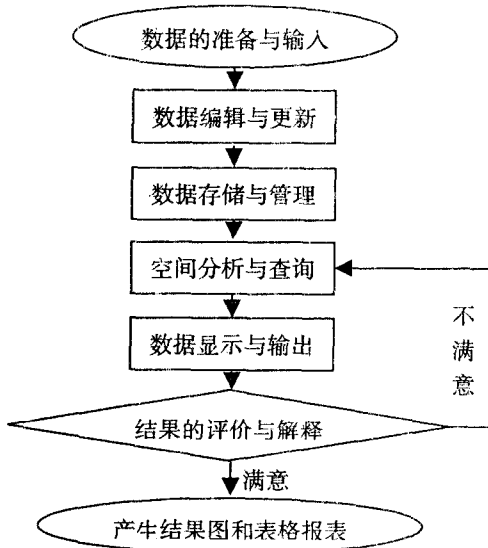


图 1 地理信息系统的数据流程

1.2 空间数据

地理信息系统中存储和处理的数据可分成两大类:第一类是反映事物地理空间位置的信息,称空间信息或空间数据;第二类是与地理位置有关的反映事物其它特征信息,称属性信息或属性数据。

空间数据是指用来表示空间实体位置、形状、大小及其分布特征诸多方面信息的数据。它具有定位、定性、时间和空间关系等特征。定位是指在已知的坐标系里空间目标都具有唯一的空间位置;定性是指有关空间目标的自然属性,它伴随着目标的地理位置;时间是指空间目标随时间的变化而变化;空间关系通常用拓扑关系表示。空间数据是一种用点、线、面以及实体等基本空间数据结构来表示人们赖以生存的自然世界的的数据^[3]。地理信息系统的核心是空间数据。除了具有一般数据的特征外,还具有很多不同,主要表现在:空间性,抽象性,多尺度与多态性,多时空性。

1.3 空间数据库

空间数据的存储和管理一直是地理信息系统研究

的重点。地理信息系统管理空间数据,最初采用文件管理方式,后来随着数据库技术的不断发展,逐步过渡到采用空间数据库管理方式。

利用空间数据库存储和管理空间数据,最先采用文件系统和关系数据库的混合管理模式;后来随着面向对象技术与数据库技术的结合,出现了面向对象空间数据模型,但目前在 GIS 领域不太通用;而基于对象关系数据库系统的空间数据模型,很好地体现了面向对象思想,并且充分利用了现有关系数据库技术,逐渐成为空间数据库建模的主流^[4]。

2 空间数据模型

数据模型是现实世界数据特征的抽象。空间数据模型考虑现实世界中的地理对象及其相互联系,为空间数据的组织和空间数据库的模式建立提供了基本方法。目前通用的主要有混合数据库模型、扩展结构模型、全关系模型、面向对象模型和对象关系模型^[5]。

2.1 混和数据库模型

在以往的地理信息系统中,空间数据和属性数据一般是分开存储的,即采用混合数据库模型。它利用文件存储空间数据及其拓扑关系,利用关系数据库存储属性数据,并通过唯一的标志符来建立它们之间的连接,如图 2 所示。

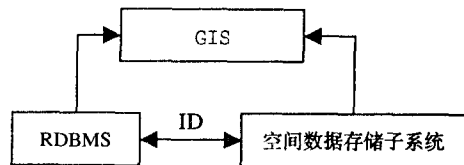


图 2 混合数据库模型

混合数据库模型一方面利用了 GIS 产品提供的空间索引、空间查询、空间分析的能力;另一方面也利用了关系数据库对属性数据的强大的查询、定位能力。但是混合数据库模型也存在以下几点明显的不足:

- 1)空间数据采用各个 GIS 厂商定义的各种格式,不利于数据共享;
- 2)空间数据以文件格式存储,安全性难以得到保证,也不利于管理;
- 3)空间数据、属性数据的一致性维护比较困难;
- 4)对 C/S、B/S 体系结构的支持能力差,不利于数据的传输^[6]。

2.2 扩展结构模型

混合数据模型的缺陷是因为两个子系统具有各自的职责,相互难以保证数据存储、操作的统一。扩展结构模型采用 DBMS 存储空间数据和属性数据,其做法是在标准的关系数据库上增加空间管理层,即利用该

层将地理结构查询语言转换为 SQL 查询,借助索引数据的辅助关系实施空间索引操作,如图 3 所示。这种模型的优点是省去了空间数据库和属性数据库之间的繁琐联结,空间数据存储速度较快。但由于是间接存储,在效率总是低于 DBMS 中的直接操作过程,并且操作过程复杂^[7]。

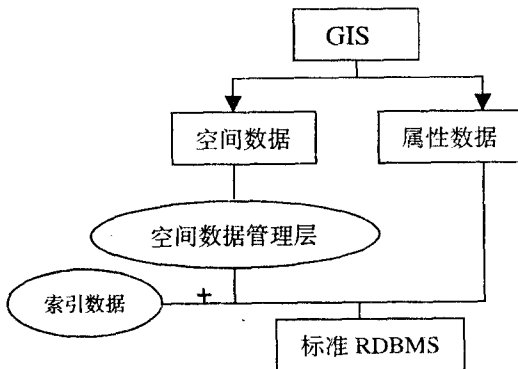


图 3 扩展结构模型

2.3 全关系模型

为了克服采用混合数据库模型的缺点,商用成熟的 GIS 软件都朝着将空间数据与属性数据存储于同一关系型数据库的方向发展。在这种全关系模型中,空间数据和属性数据都用关系数据的关系来存储,使用标准关系连接机制建立空间数据和属性数据的关联,如图 4 所示。

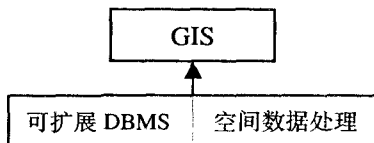


图 4 全关系模型

由于 RDBMS 具有良好的企业级数据库管理及异构数据的访问能力,以及完整性、持久性、并发控制、可恢复性、一致性和查询能力,但缺乏对海量数据、特别是空间数据类型的支持,全关系数据模型具有关系数据库的查询、检索、数据完整性和安全性等特点,但是数据类型定义方面有一定的局限性,缺乏空间 SQL 的支持^[8]。

2.4 面向对象模型

面向对象数据库是面向对象技术和数据库技术相结合的产物。它的产生和发展是为了满足数据库管理上的新要求:复杂数据类型的描述、数据之间的关联语义的管理等。而地理信息系统当中对数据的管理要求符合面向对象数据库的特点,因此,面向对象数据库是存储空间数据的较优选择^[9]。

面向对象数据模型是一种抽象模型,允许设计者在基本功能上选择最为适用的技术,具有可扩充性,它

可以模拟和操作复杂对象。面向对象数据库中任何复杂的事物都可以由封装了数据集及操作集的对象来表示,使用面向对象的模型可以很好地描述空间地物。但是面向对象数据库还没有普遍遵循的工业标准,且在安全性、完整性等许多方面还不完善,因而其使用也受到了限制^[10]。

3 对象代理空间数据模型

在对象代理模型中,用对象来描述现实世界的实体,代理对象可以继承源对象的部分或者全部的属性和方法,同时可以为不同应用需要而增加源对象中没有的属性和方法。对象代理数据库的突出优点首先表现在类的概念把一组对象所共有的特征集中起来,以说明该组对象的性质和功能^[11]。

3.1 代理机制中的类

对象代理机制针对数据库中高效管理复杂数据对象的要求,是对传统面向对象数据模型的一种扩充。客观世界的任一实体都被统一模型化为一个对象,并且每个对象都有唯一标识,称为对象标识^[12]。

(1) 源类的定义。

每一个地理对象都有一个对象标志符号、一些属性和相关的一些方法。具有相同属性和方法的地理对象被定义为一个类,它由名字、外延和类型组成。这个类主要包含一些基本的非空间属性,以及与空间属性的关联关系。一个类 C 可以表示成:

$$C = \langle \{o\}, \{T_a: a\}, \{m: \{T_p: p\}\} \rangle$$

1) 符号 o 表示一个具体的地理对象;

2) $\{T_a: a\}$ 是 C 的属性的集合,其中 a 是属性名, T_a 是属性类型,每个对象的属性值表示为 $o.a$ 。值得注意的是,在这个类里还定义了一个地理属性,其类型是 geometry,与空间数据类相连。对于每个属性 $T_a: a$,有两种基本的操作如下:

$$\text{read}(o, a) \Rightarrow \uparrow o.a$$

$$\text{write}(o, a, v) \Rightarrow o.a: v$$

这里, \Rightarrow , \uparrow 和 $=$ 分别代表了激活、结果返回和赋值操作。

3) m 是方法名, $\{T_p: p\}$ 类中对象方法的集合。对某个方法的操作可定义如下: $\text{apply}(o, m, \{p\})$ 。

(2) 空间数据类的定义。

空间数据具有单独的函数和方法,因此文中为地理对象的空间数据也定义了一个类,具体表现其地理属性:坐标值,参考坐标,维度等。定义为:

$$C^s = \langle \{o_g\}, \{T_b: b\}, \{f: \{T_p: p\}\} \rangle$$

1) $\{o_g\}$ 表示对象的集合,称为类 C_g 的扩充。

2) $\{T_b: b\}$ 包含空间数据的维度、投影坐标系统、几何数值等属性。

3) $f: \{T_p: p\}$ 定义了空间数据上的一些函数, 如表 1 所示。

表 1 空间数据的基本函数

名称	描述
NumPoints (geometry)	求几何体中第一条线点的个数
NumInterior (geometry)	求几何中首多边形内环编号
IsClosed(geometry)	判断几何起点与终点是否重合
NumGeometrie(geometry)	求几何集合体中几何体的个数
GeometryN(geometry)	求几何集合体的第 N 个几何体
SetSRID(geometry)	求几何的空间参考系统参数
Centroid (geometry)	求中心标识的函数
IsClosed(geometry)	判断几何起点与终点是否重合

(3) 代理类的定义。

代理对象是源对象的扩展或者客户化, 一个对象可以有很多的代理对象, 以此来适应于不同的应用环境和地理对象的不同形式。代理对象可以从源对象或者其他的代理对象继承相关的属性和方法, 同时, 它自身可以有额外的属性和方法。文中把这样的一类代理对象归结为一个代理类, 设定一个源类的形式为:

$$C^s = \langle \{o^s\}, \{T_a^s: a^s\}, \{m^s: \{T_p^s: p^s\}\} \rangle$$

那它的代理类被定义为:

$$C^d = \langle \{o^d \mid (o^d \rightarrow o^s) \vee (o^d \rightarrow \dots \times o^s \times \dots) \vee (o^d \rightarrow \{o^s\}), sp(o^s) \vee cp(\dots \times o^s \times \dots) \vee gp(\{o^s\}) = \text{true}\}, \{T_a^d: a^d\} \cup \{T_a^s: a^s\}, \{m^d: \{T_p^d: p^d\}\} \cup \{m^s: \{T_p^s: p^s\}\} \rangle$$

1) 其中 $\{o^d \mid (o^d \rightarrow o^s) \vee (o^d \rightarrow \dots \times o^s \times \dots) \vee (o^d \rightarrow \{o^s\})$ 表示 o^d 是 $o^s, \dots \times o^s \times \dots$ 或者 $\{o^s\}$ 的代理对象; 另外, sp, cp 和 gp 分别表示选择、联合和分组谓词。

2) $\{T_a^d: a^d\} \cup \{T_a^s: a^s\}$ 定义了代理类 C^d 的属性集, $\{T_a^s: a^s\}$ 是继承源对象的属性, $\{T_a^d: a^d\}$ 是附加的属性。

3) $\{m^d: \{T_p^d: p^d\}\} \cup \{m^s: \{T_p^s: p^s\}\}$: 代理 C^d 类的方法集, $\{m^d: \{T_p^d: p^d\}\}$ 是继承的方法, $\{m^s: \{T_p^s: p^s\}\}$ 是附加的方法。

3.2 代理代数

代理类通过创建代理对象而作为它的实例, 产生切换操作实现属性和方法的继承, 并能加入一些额外的属性和方法的定义。对象代理模型提供了对象代理代数用于派生代理类。

Select 操作用于派生一个代理类, 其实例为源类中根据选择谓词而被选择的实例的代理对象。

(定义) 如果 $C^s = \langle \{o^s\}, \{T_a^s: a^s\}, \{m^s: \{T_p^s: p^s\}\} \rangle$ 为一个源类, 通过 Select 操作派生的代理类是

$$C^d = (C^s, sp)。$$

1) C^d 的外延是 C^s 中满足选择谓词 sp 的那些实例的代理对象的集合, 被表示为 $\{o^d \mid o \rightarrow o^s, sp(o^s) = \text{true}\}$ 。

2) C^d 的 attribute 集被定义为 $\{T_a^d: a^d\}$, 它从 C^s 的属性 $\{T_a^s: a^s\}$ 继承而来。以 $\{T_a^d: a^d\}$ 的形式继承 $\{T_a^s: a^s\}$ 切换操作通过以下方式实现:

$$\text{read}(o^d, a^d) \Rightarrow \uparrow f_{T_a^s \mapsto T_a^d}(\text{read}(o^s, a^s))$$

$$\text{write}(o^d, a^d, v^d) \Rightarrow \text{write}(o^s, a^s, f_{T_a^s \mapsto T_a^d}(v^d))$$

3) C^d 的方法集被定义为 $\{m^d: \{T_p^d: p^d\}\}$, 它从 C^s 的 $\{m^s: \{T_p^s: p^s\}\}$ 继承。以 $m^d: \{T_p^d: p^d\}$ 的形式从 $m^s: \{T_p^s: p^s\}$ 继承的切换操作通过以下方式实现:

$$\text{apply}(o^d, m^d, \{p^d\}) \Rightarrow \uparrow \text{apply}(o^s, l^s, \{f_{T_p^s \mapsto T_p^d}(p^d)\})$$

Project 操作用于仅仅从源类继承部分属性和方法而派生代理类。每个操作的结果是一个能被代理操作进行操作的代理类, 这样就达到了与关系代数一样的灵活性。

3.3 SQL 语法

基本类是没有源类的类, 其所有属性都是实属性。定义基本类的命令包括: 定义类名、属性名和属性类型; 建立约束; 创建方法, 定义方法名称、参数类型、返回类型及函数体。命令形式如下:

CREATE CLASS(class_name)

{[ATTRIBUTE]}(<column> <type> <attr_constraint>),
[<class_constraint>])

[METHOD(<method_definition>)];

其中有:

<class_name>: 类的名字;

<column>: 属性名;

<type>: 属性的数据类型;

<attr_constraint>: 各种列约束语句;

<class_constraint>: 各种类(表)约束语句;

<method_definition>: 方法定义子句

一个方法定义子句中可以有多个方法定义体, 它们之间用逗号隔开。如果不需要方法, 则省略方法定义子句。其命令形式如下:

<method_definition>::=

<method_name>(<parameter_type>)

RETURNS<return_type>

AS<method_body>

LANGUAGE<language>

其中:

<method_name>: 方法名;

<parameter_type>: 函数参数的数据类型表;

<return_type>:函数返回的结果类型;

<method_body>:方法体(程序代码);

<language>:方法体所使用的程序语言

代理类的定义包括定义类名、代理类型、扩展属性和方法,以及代理规则。通过删除类的命令,用户可以将任何类从数据库中删除。

4 结束语

分析并指出了地理信息的存储和管理是当前地理信息系统发展的重点和难点。通过对地理信息系统空间数据建模的阐述,分析了混合数据库模型、扩展结构模型、全关系模型、面向对象模型和对象关系模型解决的问题和依然存在的不足,引出了使用对象代理机制建立地理信息系统模型的思想,并提出了对象代理空间数据模型。

文中的创新之处在于把对象代理机制引入到空间数据库的建模以及地理信息的组织和显示中,完成了地理信息系统中的空间数据存储和管理的新形式,更大程度地满足了用户的操作需求。虽然文中的研究取得一些成果,但很多地方仍需要进一步研究和改进。

参考文献:

- [1] 许捍卫,冯学智.空间数据存储机制研究[J].计算机应用研究,2003,20(2):39-40.

(上接第174页)

4 结束语

整个系统主要在Windows系列平台上运行,采用的开发工具为MICROSOFT VC6.0。通过网络攻防训练模拟平台,可以较好地进行攻防理论及技巧的训练,增强受训人员的训练效果。整个平台以插件式实现,功能可以根据网络攻防相关技术的发展而扩展,具有可扩展性、低成本性和易用性。同时,通过在网络训练模拟平台中加入辅助决策功能,可以增强受训人员的决策能力。

参考文献:

- [1] Department of Defense, USA. Network Centric Warfare, DoD Report to Congress[R]. Washington DC: DoD, 2001.
- [2] SAlberts D, Garstka J J, Stein F P. Network centric warfare: developing and leveraging information superiority[M]. Washington DC: DoD Command and Control Research Program, 1999.
- [3] 王慧强,赖积保,朱亮,等.网络态势感知系统研究综述[J].计算机科学,2006,33(10):5210-5215.
- [4] 景旭,唐磊,韩永国.基于信息对抗的网络集成防御系

- [2] 张成才,孙喜梅,黄慧. SDE的实体关系模型空间数据管理方式研究[J]. 计算机工程与应用, 2003, 39(2): 199-201.
- [3] 罗忠文. 应用对象关系型数据库存储GIS数据[J]. 中国地质大学学报: 地球科学版, 2002, 27(3): 267-270.
- [4] 彭智勇, 彭煜玮, 翟博. 一个基于对象代理模型的多表现地理信息系统[J]. 计算机应用, 2006, 26(9): 2016-2019.
- [5] 谢卫平. 基于对象代理模型的地理信息建模和存储管理[D]. 武汉: 武汉大学, 2006.
- [6] 周迪民, 段国云. 地理信息系统属性数据不确定性的研究[J]. 计算机技术与发展, 2009, 19(12): 174-177.
- [7] 蔡正林, 韩金华, 李梦琪. 网格GIS体系结构研究及应用[J]. 计算机技术与发展, 2006, 16(7): 221-223.
- [8] 朱光, 季晓燕, 戎兵. 地理信息系统基本原理及应用[M]. 北京: 北京测绘出版社, 1997: 253-324.
- [9] Tang A, Adams T, Usery E. A Spatial Data Model Design for Feature Based Geographical Information Systems[J]. Geographical Information Systems, 1996, 10(5): 643-659.
- [10] Borges K, Davis J, Laender A. OMT-G: An Object-Oriented Data Model for Geographic Applications[J]. GeoInformatic, 2001, 5(3): 221-260.
- [11] Kosters G, Pagel B, Six H. GIS-Application Development with GeoOOA[J]. Geographical Information Science, 1997, 11(4): 307-335.
- [12] Shekhar S, Coyle M, Goyal B, et al. Data Models in Geographic Information Systems[J]. Commun ACM, 1997, 40(4): 103-111.

统[J]. 微计算机信息, 2006, 8(3): 99-100.

- [5] 卢昱. 协同式网络对抗[M]. 北京: 国防工业出版社, 2003.
- [6] Kumar S. Classification and Detection of Computer Intrusions[D]. West Lafayette, Indiana: Department of Computer Science, Purdue University, 1995.
- [7] Aslam T, Krsul I, Spafford E. Use of a Taxonomy of Security Faults[C]//Proceedings of the 19th NIST-NCSC National Information Systems Security Conference. [s.l.]: [s.n.], 1996: 551-560.
- [8] 肖道举, 杨素娟, 周开锋, 等. 网络安全评估模型研究[J]. 华中科技大学学报: 自然科学版, 2002, 30(4): 37-39.
- [9] 冯登国, 张阳, 张玉清. 信息安全风险评估综述[J]. 通信学报, 2004, 25(7): 10-18.
- [10] 郭亚军. 综合评价理论与方法[M]. 北京: 科学出版社, 2002.
- [11] 汪渊, 蒋凡, 陈国良. 基于图论的网络安全分析方法研究与应用[J]. 小型微型计算机系统, 2003, 24(10): 1865-1869.
- [12] 汪渊, 蒋凡, 陈国良. 一种基于安全案例推理的网络安全分析方法[J]. 小型微型计算机系统, 2003, 24(12): 2082-2085.