

分布式空间数据管理技术研究

乔彦友 赵健

(中国科学院遥感应用研究所, 北京 100101)

摘要 对分布式空间数据库管理的有关技术问题进行了探讨;对商业软件 Oracle 及 ESRI 系列的分布式空间数据管理功能进行了分析;最后提出了建立分布式空间数据应用系统的一种实用策略。

关键词 分布式空间数据库 空间元数据 互联网

中图法分类号: TP311.133.1 TP316.4 **文献标识码:** A **文章编号:** 1006-8961(2001)09-0873-06

Techniques in Distributed Spatial Database Management

QIAO Yan-you, ZHAO Jian

Institute of Remote Sensing Applications, CAS, Beijing 100101

Abstract Techniques needed in the management of distributed spatial databases are discussed, abilities of distributed management in the commercial software Oracle and ESRI products are described, and a practical methodology to build distributed spatial applications is suggested. In a distributed spatial database, there is usually a main server and many data servers, and all of them are linked by networks such as internet. In order to guarantee the normal data flow and analysis among clients, main server and data servers, issues of net communication, metadata management, data consistency, data serving, spatial data search engine, seamless data integration, quick updating, data transfer, data queuing and security protection should be solved. Oracle and ESRI series products can act as frames in distributed spatial database management applications, and the performance will be much better if some key components can be developed from bottom.

Keywords Distributed spatial database, Spatial metadata, Internet

0 引言

随着遥感(RS)、地理信息系统(GIS)及全球定位系统(GPS)3S技术的飞速发展,空间数据获取的数量正在以前所未有的速度增加,且数据生产及应用的行业分工也越来越细,这就需要在不同数据生产者 and 使用者之间建立合理的共享机制,由此就提出了分布式空间数据存储和管理的思想^[1,2]。网络技术为数据分布存储提供了基础,它可以将异地配置的若干空间数据存储站点连接起来,实现不同站点之间空间数据库的透明连接^[3,4]。为了实现空间数据由集中式存储管理向分布式存储管理的变化,就要解决一系列的技术问题,如网络技术、多源数据综合

管理技术、数据传输技术、空间数据搜索引擎技术、系统安全技术等。

1 分布式空间数据库

1.1 定义

分布式空间数据库系统由若干个站点集合而成,这些站点又称为节点,它们通过网络连接在一起,每个节点都是一个独立的空间数据库系统,它们都拥有各自的数据库和相应的管理系统及分析工具。整个数据库在物理上存储于不同的设备上,而在逻辑上则是一个统一的数据库。在应用时,用户可以不考虑数据存储的具体物理位置,就象对集中式数据库一样来访问分布式数据库。

1.2 特点

分布式空间数据库有如下特点:

(1) 在分布式数据库系统里不强调集中控制概念,它具有一个以全局数据库管理员为基础的分层控制结构,但是每个局部数据库管理员都具有高度的自主权。

(2) 数据独立性 在集中式数据库系统中,数据独立性包括两个方面:数据的逻辑独立性与数据的物理独立性,其含义是用户应用程序与数据的全局逻辑结构、数据的存储结构无关。在分布式数据库系统中,数据独立性除具有逻辑独立性和物理独立性之外,还有数据分布独立性,即分布透明性(Distribution Transparency)。尽管数据库可能位于不同的物理节点,但用户看到的是一个完整的统一的数据库,即逻辑数据库,可以很方便地访问逻辑数据库中的任何数据,而不需关心他所需要的数据是存储在哪一个站点上。

(3) 适当的数据冗余 与集中式数据库系统不同,数据冗余在分布式系统中被看作是所需要的特性。首先,如果在需要的节点复制数据,则可以提高局部的应用性;其次,当某节点发生故障时,可以操作其他节点上的复制数据,这可以增加系统的有效性。

2 分布式空间数据库的结构

分布式空间数据库的结构有多种,其中比较常用的一种是基于空间元数据的结构。所谓空间元数据(Metadata)就是关于空间数据的数据,它是对空间数据库有关情况的描述。一般说来,元数据库是以关系型数据为基础建立的。在空间元数据库中,每条记录都代表一个空间数据库,它包括对该空间数据库基本情况的描述,例如所覆盖的地理范围、数据精度、投影方式、数据用途、数据生产时间、数据生产方式等一系列标准因素。在分布式存储框架下,元数据的一个重要字段就是服务器地址,即某个空间数据库所处的服务器的网络地址或序号。在基于空间元数据的分布式空间数据库中,除了一系列用于提供空间数据服务的服务器站点以外,还应当有一个中心服务器,用于对上述所有服务器进行总控管理。其中最主要的就是对所有空间数据库的元数据进行管理,即存储了一个统一的空间元数据库,它描述了上述所有服务器站点上所有空间数据库的情况,并可以通过每条元数据访问到它所对应的空间数据库。在每个空间数据库服务器上,也有一个元数据库,它

描述了该服务器上所有空间数据库的情况。当该元数据库发生更新时,服务器会通过消息将更新情况发到中心服务器,由中心服务器上的元数据管理系统自动更新总的空间元数据库。客户对分布式数据库的访问一般是通过中心服务器进行的,当然也可以直接访问各个分站点。这种分布式空间数据库的框架结构可由图1所示。

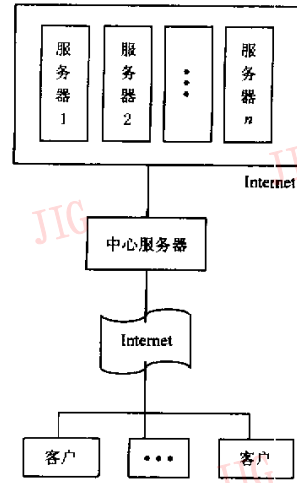


图1 基于元数据的分布式空间数据库框架

3 关键技术

3.1 网络通信技术

良好的通信机制是实现分布式数据库系统的基础。目前的一些开发工具提供了直接访问远程数据库的能力,但大部分都没有提供直接的消息传递机制。而作为分布式系统,直接的消息传递机制是系统通信的重要组成部分。

分布式数据库管理系统中的消息传递机制是通过使用 Windows socket 消息驱动程序实现的。一般使用“可定制用户对象”在客户机和服务器之间通过函数传递信息。首先定义一个可定制用户对象,并定义一系列用户对象函数。之后,在服务器端为该用户对象定义一个代理(proxy),然后把所定义的代理对象转移到客户机上去。这样在客户机端,用户可以使用在服务器上定义的函数。当系统发现所用到的函数是在代理对象中定义的时,系统将不在本地而是到服务器端执行函数。通过这种方式,客户机和服务器之间实现了消息的传递。

除客户机和服务器之间进行消息传递外,服务

器和服务器之间也进行消息传递,因为系统中存在着多个服务器,任何一个进入系统的服务器都要与中心服务器进行连接,以便确立它是分布式数据库的一个数据站点。

3.2 空间元数据库及其管理

空间元数据库中存储了系统中全部空间数据库的某些标准属性,以便进行总控管理。在中心服务器上存储着一个总的空间元数据库,而在各个空间数据服务器上又各自有一个较小的空间元数据库,它描述了本站点上所有空间数据库的情况,所有分站点上空间元数据的总和应当等于中心服务器上总的空间元数据库。空间元数据库的建立一般采用关系型数据库来实现。当一个站点上的空间数据库情况发生变化时,它应当能够自动修改本站点上空间元数据库的内容,并向中心服务器发送一条消息,以便及时更新总的空间元数据库。必须解决总空间元数据库与各个站点元数据库在管理时的协调问题,才能使运行正常,这主要是通过代理程序来实现,即在所有服务器上实时运行着代理程序,当空间元数据库的内容发生变化时,通过网络向中心服务器及本身发送消息,服务器上的代理程序在接受消息以后,就会执行有关空间元数据管理方面的操作。

此外,空间元数据库的管理还包括空间元数据库的建立、空间元数据内容的输入,空间元数据的查询与发布等基本功能。为了能够在中心服务器及各个站点服务器实现对空间元数据库的管理,必须开发专用的软件。

3.3 数据冗余与一致性

为保证系统中数据的安全,在系统发生局部故障时仍能保持对发生故障部分的数据的正常访问,系统在其他节点保留该部分数据的副本,这也就带来了数据的冗余及一致性问题。用户对若干数据副本中的任一个的修改必须能同时反映到所有副本中去,否则系统中的数据不一致将会带来系统的混乱。

对于系统中存在的同一个库的多个副本,系统元数据库中记录了所有副本的位置,状态(是否正在被访问),以及是否与最近的副本一致等,并考虑允许某些副本的延迟更新。首先,如果系统中的另外一副本处于加锁状态(即该副本正在被访问),那么所做的修改不能被提交,因为同时被访问的两个副本所做的修改有可能发生冲突。其次,如果系统发现根本无法连接到某一副本(可能该副本所处的节点未开机或出现故障),那么系统将暂时忽略该副本,并

提交所做的修改。但这并不意味着抛弃该副本,而是允许该副本延迟更新。系统把所做的修改做成一个事务文件保存起来,同时设置空间元数据库中对应该副本的与最近副本非一致标志,表明该副本有未提交的修改,并把在该副本处未提交的事务文件名称记录到元数据库中去。在该副本所在的节点启动或重新正常加入系统中时,它将首先查询元数据库中最近副本非一致的标志。如果该标志被设置,当前节点先运行相应的事务文件。若某一个事务文件无法提交,表明该副本已经与其他副本产生了不可弥补的差别,该节点就通知系统抛弃该副本,并建议系统重新制作该节点处的数据副本。如果所有事务文件全部提交成功,该副本将与其他副本重新保持一致,并继续保留在系统中。这种方法维护了系统的数据一致性,同时使系统拥有了更大的灵活性。

3.4 数据服务技术

在分布式空间数据库中,空间数据分布于各个数据服务器站点上。在应用这些数据时,一般由客户端向中心服务器发出数据请求,中心服务器在接受请求后,调用总的空间元数据库,确定存储所需数据的服务器,然后向具体的数据服务器发出数据请求,由相应数据服务器上的服务程序从有关空间数据库中生成所需要的数据,并将结果传回中心服务器,进而传回客户端。中心服务器既是整个分布式空间数据库的核心,又是为客户进行数据服务的代理。空间数据服务采用存根/架构(stub/skeleton)机制,即处于客户端的应用程序通过一个存根程序来代理数据请求,它生成有关数据请求命令,向中心服务器发出数据服务请求,并最后负责接收服务器端传回的结果数据;在中心服务器上有一个处于激活状态的架构程序,它负责监测接收来自客户端的数据请求,对收到的命令请求进行分析,得出数据所在的站点地址,将数据请求命令发向相应的数据站点,并负责接收从数据服务器返回的结果数据,然后再将这些数据传到发出初始命令的客户端;在每个数据服务器上,也有一个处于激活状态的架构程序,它接收由中心服务器发来的数据请求,对数据请求的内容进行分析,自动调用数据查询和分析工具,对本站点上的空间数据库进行操作,得到所需要的结果数据,最后再将这些数据传回中心服务器。总的结构如图2所示。

stub/skeleton 是应用程序跟空间数据库之间的接口,它实际上是扮演着代理的角色,stub 和 skeleton 分别是客户程序和服务程序的代理人。在

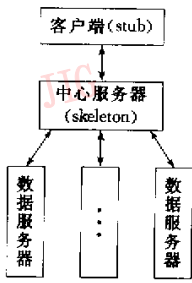


图2 分布式空间数据服务

分布式空间数据库中采用数据服务策略有其显著优点,它使得对空间数据库的调用完全限制在所感兴趣的数据集上,即只有需要的那部分数据才通过网络传输,从而避免了海量地理空间数据的传输,不致引起网络堵塞。

3.5 空间数据搜索引擎

空间数据搜索实际上是一类特殊的空间数据服务,用于从海量空间数据库中找出感兴趣的的信息,例如从地理信息数据库中找出满足某种属性限制条件的所有地理特征,从遥感影像数据库中找出具有某种资源分布的地区等。空间数据搜索本身就是一个非常复杂的问题,由于空间特征的语义表达缺乏统一的标准,而地理特征之间的关系运算量也非常大,因而搜索的效率也就比较低。在分布式存储的情况下,数据的总量急剧上升,且在搜索时要大量涉及客户端与服务器端、服务器相互之间的协调问题,因而问题更为突出,必须根据分布式数据库的框架对空间搜索算法进行优化,以满足空间数据搜索的需要。

空间数据服务和空间数据搜索的不同之处在于调用数据服务器的数量。一般说来,每次数据服务请求的命令都是发到一台数据服务器,由该服务器上的服务程序将所需要的数据生成并传回,虽然不同的命令所发往的数据服务器可能不同。对于空间数据搜索来说,每次搜索都要在所有数据服务器站点范围内进行,即命令要发到所有的数据服务器上,并由各自服务器上的服务程序完成对站点上数据库的搜索,所有这些搜索结果再传回中心服务器并进行综合,最后将综合后的结果传回客户端。

3.6 数据无缝连接技术

空间数据的格式有多种多样,如不同的传感器所获取的遥感图象格式各不相同,不同地理信息数据库建立者所生产的数据各有特点,而不同站点上的数据用途及格式也不同,所有这些数据都应成为

整个分布式空间数据库的组成部分,这就要解决它们之间的无缝连接问题。

3.7 数据快速更新技术

分布式空间数据库的更新体现在多个方面,必须分别制定合理的更新方法。例如,可以有新的空间数据站点加入进来,从而扩大了数据库的范围,这时要将新站点注册到中心服务器,并用其空间元数据库的内容来补充中心服务器中的元数据库;有的数据站点根据形势的变化而必须退出分布式空间数据库,这时要在中心服务器上注销相应的数据服务器,并在总的空间元数据库中删除有关元数据;在某个数据站点内部,随着时间的变化,可能生成了新的空间数据库,这时要将新数据库的空间元数据发送到中心服务器,以便及时更新总的空间元数据库;在很多情况下,数据站点上的数据库个数及名称都没有发生变化,而某些数据库的内容却发生了变化,这时只要像集中式数据库的更新那样更新相应的数据库即可。

由于各个站点上空间数据库的更新是各自独立完成的,而很多情况下它们要引致对中心服务器上元数据库的更新,这就存在并发控制问题,可以通过在架构程序的内部设置命令来更新队列,从而使得总元数据库的更新依命令接收时间次序进行。

3.8 数据中转技术

在某些特定条件下,有的数据服务器可能会与中心服务器联系中断,但却与另一台数据服务器联系畅通,即可以通过该服务器中转,因此必须存在数据库服务器之间的转发机制,这可以通过数据库服务器上的架构程序来完成。源服务器上的架构程序负责发送数据,而目标服务器上的架构程序则负责接收数据,并将它发往中心服务器。数据中转的最初命令仍是由中心服务器发出,中心服务器在需要某个数据服务器上的数据而又不能与其连通时,就会将命令发向另一台数据服务器,由该服务器向要连通的数据服务器转发数据请求命令,如果仍不通,中心服务器就会重新选择一台数据服务器,直到能够中转为止。数据服务器在收到中转过来的命令以后,就会根据要求将数据发出。具体的流程如图3所示。

3.9 数据传输队列技术

在分布式空间数据库的应用流程中,经常要将数据同时请求发到各个数据服务器,有些还是通过中转而发,这样就会同时有大量数据往中心服务器传输,因此在中心服务器必须有一个接收分时控制

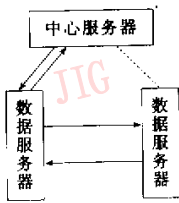


图3 数据中转流程图

程序,以保证数据接收的正常运行.从每台数据服务器来看,它可能接收了多条数据请求,要与中心服务器及其他数据服务器进行数据传输,因此必须有一个队列管理程序来控制所有数据传输任务的执行.

3.10 系统安全性

系统的安全性,即保证系统在局部故障条件下能持续运行并且系统中的数据无丢失现象,它是设计分布式数据库管理系统所必须考虑的问题之一^[5].在系统运行的任一时刻,都由中心服务器对整个系统进行维护,包括系统空间元数据的维护、安全权限管理等.同时,在中心服务器出现故障时,系统中的任何一台数据服务器都有可能被提升为中心服务器,这就需要将所有空间数据服务器上的元数据库传输到新的中心服务器,并由该服务器对它进行管理和维护.

为了保证分布式空间数据库系统的安全性,可以采取系统登录检测、系统操作权限的控制、存取控制、使用日志、权限分级等技术.

4 现有系统的分布式空间数据管理功能

4.1 Oracle 数据库

Oracle 是以高级结构化查询语言(SQL)为基础的大型关系数据库,它用方便逻辑管理的语言操纵大量有规律数据的集合,是目前最流行的客户/服务器(CLIENT/SERVER)体系结构的数据库之一^[6].其中,Oracle8I 是一个面向 Internet 环境的数据库,它为用户提供了—个开放的、高性能的 Web 服务器系统.

Oracle 系统有许多优良特性,特别是它提供了分布式数据库能力和空间数据管理功能,可通过网络较方便地读写远端数据库里的数据.Oracle Spatial Data Option 对数据存储进行了扩充,是目前唯一能支持空间数据操作的关系数据库,它提供了对大量空间数据的存储、管理和查询检索功能.

SQL * NET 是 Oracle 分布式处理的基础.通过 SQL * NET,一个应用程序可以并行存取本地或

远程的多个空间数据库.当存取远程数据库时,在客户端,SQL * NET 将来自用户程序接口层的子程序调用组装成信息报文,经过异种机环境所需要的数据转换后,通过网络将报文发送给远程计算机;在服务器端,SQL * NET 接受报文,做必要的数据库转换,并将子程序调用参数传送到适当的 Oracle 核心入口点,在相反方向上服务器端驱动 SQL * NET 传送的是数据和返回码信息.

Oracle WebDB 是一个用于建立、部署和灵活监视 Web 数据库应用和内容驱动的 Web 站点的解决方案.通过组合直观的 HTML 界面和基于浏览器方式的完整 HTML 工具集,它允许用户轻松、快捷地开发 Web 数据库应用.

Oracle Data Server 中的 Java VM 允许开发人员在数据库中编写、存放和执行 Java 代码,允许用 Java 来创建存储过程、函数和触发器,这大大降低了对数据库编程技术的要求.客户或服务器 Java 代码可以轻易地嵌入 SQL,实现与关系数据库的通信.客户方 SQL 语句中的 Java 代码通过一个 Java 预编译,可以生成基于 JDBC 的 Java 可执行程序.服务器方 Java 代码则可通过自身接口直接访问数据库.

4.2 ESRI 系列产品

ESRI 为各种平台提供了全系列的 GIS 解决方案,以对那些用于标准 SQL 数据库软件的分布式空间数据进行管理.对于分布式环境来说,在 Internet/Intranet 上发布地图数据的方法有两种.一是基于 MapObjects 的 IMS,二是基于 ArcView 的 IMS.基于 MapObjects 的 IMS 可将用户用 MapObjects 开发的应用及系统数据在 Internet(或 Intranet)上发布出来.MapObjects 为用户提供了一种基于 OLE 的开发环境,包括 ActiveX 控件 Map 和 WebLink 以及 35 个以上的可编程组件;MapObjects 可以广泛地使用在各种开发平台上,如 VisualBasic、Delphi、VisualC++、Power Builder 等,使用户可以根据自己的实际情况,开发出符合自己要求的 Internet 应用.

基于 ArcView 的 IMS 提供了一种极为简便快捷的方法帮助用户将自己指定的应用(及其数据)自动地发布于 Internet(Intranet)上,用户在这整个的过程中都不需编程.

ESRI 提供空间数据服务的软件为 SDE,而客户端软件则有 ArcView,MapObjects,ARC/INFO,

Intranet/Internet Map Server, MapObjects IMS, ArcView IMS 等.

5 建立分布式空间数据库应用系统的策略

要建立一个基于分布式空间数据库的应用系统,主要有两种策略.一是完全从底层开始进行开发,二是在现有商业产品的基础上进行二次开发.

如果选择从底层开始开发,就要应用网络编程技术,采用 VC++, VB, JAVA 等开发工具,搭建分布式空间数据库管理的各项功能,包括底层数据结构的设计,网络通信软件的编写,数据传输流程的制定等一系列问题.其优点是针对性强,能够根据实际应用的开发具体的功能,而不用开发那些不需要的功能.由于所有功能都有源程序,因而开发者在对它进行完善及维护时都比较容易,可以方便地增加新的功能.其最大的缺点就是工作量大,开发周期长,往往要经过多年的测试,才能形成稳定产品,因此一般不采取这种策略.

由于 Oracle, SDE, ArcView 等商业软件的分布式空间数据库管理功能已经比较成熟,如果在它们的基础上进行集成,采用二次开发的方法进行开发,就能比较快地搭建功能比较完善的分布式空间数据库应用系统.这种方案的优点是开发周期较短,可以及时提供应用,且用于集成的各个主要模块都是成熟产品,因而在功能上可信度较高,是一种较为可行的开发策略.当然,在采用这种方法时,并不是所有的模块都有现成的组件可用,有时还要用高级编程语言开发一些关键的部件,如客户端界面、队列管理程序、空间元数据库管理程序及某些插件等.通过自行开发这些部件,将会使各种商业软件各种功能之间的联系更为紧密,协调性更好,从而更大程度地发掘它们的应用潜力.例如,可以用 Oracle 数据库作为建立分布式空间数据库的基本软件,来存储空间数据,包括矢量数据和栅格数据;用 Oracle 存储空间元数据并建立元数据管理系统;用 SDE 做为空间数据服务软件,实现空间数据服务器与中心服务器及客户端之间的数据传输与服务.这种方案的缺点是费用比较高,必须制定合理的开发方案,以便在总费用较小的情况下,保证系统开发完成.

6 结论

分布式空间数据库管理涉及中心服务器、空间数

据服务器客户端及网络通信等多个环节,需要解决一系列的技术问题,才能实现分布式空间数据库的透明性,像访问集中式数据库一样访问分布式数据库.在现有商业软件的基础上进行集成,并用高级语言开发一些关键的部件,就可以比较快地建立基于分布式空间数据库的应用系统.

参考文献

- 1 张犁,林晖,李斌. 互联网时代的地理信息系统[J]. 测绘学报, 1998, 27(1): 9~15.
- 2 袁相儒,龚健雅,陈莉丽等. 多种数据源地理信息处理的 Internet GIS 方法[EB/OL]. <http://www.cpst.net.cn/cpst/xslt/xslw/oa99lw/oa99lw34.htm>, 1999.
- 3 Kenn Gardels. A comprehensive data model for distributed, Heterogeneous Geographic Information[EB/OL], <http://www.regis.berkeley.edu/gardels/geomodel-def.html>, 1998.
- 4 Coddington P D *et al.* Interfacing to on-line geospatial imagery archives [EB/OL]. <http://www.dhpc.adelaide.edu.au/reports/071/abs-071.html>, 1999.
- 5 贾楠,吴裕树. 分布式数据库管理系统中的安全管理设计[EB/OL]. <http://www.cpst.net.cn/cpst/xslt/xslw/oa99lw/oa99lw34.htm>, 1999.
- 6 ORACLE 数据库简介[EB/OL]. <http://hopf.chem.brandeis.edu/yanglingfa/online/oracle/orademo.htm>, 1999.

乔彦友 1964 年生,博士,研究员.主要从事地理信息系统、空间决策支持系统及遥感应用等领域的工作.

赵健 1963 年生,副研究员.主要从事地理信息系统、计算机网络应用及遥感应用领域的研究工作.