

网格环境下空间数据共享与互操作技术研究

王方雄¹ 侯英姿^{1,2} 杨俊¹

(辽宁师范大学城市与环境学院 大连 116029)¹

(武汉大学测绘遥感信息工程国家重点实验室 武汉 430079)²

摘要 空间数据共享与互操作是空间信息网格的关键技术之一。分析了网格环境下中空间数据共享与互操作涉及的角色与关键环节,提出了空间数据共享与互操作框架,探讨了基于 GML/SOAP 的空间数据互操作表达技术,通过扩展 OGSA-DAI 实现了对分布、异构的空间数据的访问与集成,并开发了一个原型系统,验证了基于 GML/SOAP/扩展的 OGSA-DAI 的空间数据共享与互操作技术的可行性。

关键词 空间数据互操作, SOAP 消息, GML, OGSA-DAI

Geospatial Data Sharing and Interoperability Technologies in Grid Environment

WANG Fang-xiong¹ HOU Ying-zi^{1,2} YANG Jun¹

(School of Urban and Environmental Sciences, Liaoning Normal University, Dalian 116029, China)¹

(State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan 430079, China)²

Abstract Geospatial data sharing and interoperability are one of key technologies of Spatial Information Grid. Based on the analysis of the roles and the critical steps of the geospatial data interoperability in Grid Environment, a geospatial data sharing and interoperability framework was put forward. The representation technologies for geospatial data interoperation were presented in detail based on Geographical Markup Language (GML) and Simple Object Access Protocol (SOAP) messages. OGSA-DAI technology was extended to access and integrate distributed heterogeneous geospatial data in Grid Environment. A prototype system reveals a good feasibility of the geospatial data interoperability technologies based on GML, SOAP and extended OGSA-DAI.

Keywords Geospatial data interoperability, SOAP messages, GML, OGSA-DAI

网格环境下,空间数据具有异构性、时态性、多源性、海量性以及存储格式多样性等特点,并无所不在地分布于众多的网格结点,空间数据共享与互操作异常艰难。空间数据互操作是在分布式计算和异构空间数据库的情况下出现的,它强调空间数据集之间相互透明的访问,其目标就是简单、透明、开放、统一地交换空间数据^[1],数据共享是数据互操作的最低层次。空间数据共享与互操作是空间信息网格(Spatial Information Grid, SIG)的关键技术之一^[2,3],现有 4 种实现方法^[3]:①空间数据转换方法;②基于直接数据访问的互操作方法;③基于公共接口的互操作方法;④基于 Web 服务的互操作方法。方法①是实现空间数据共享与互操作的最低层次,没有考虑空间数据的处理,还不能达到真正的互操作。方法②建立在充分了解空间数据格式的基础之上,要求 GIS 软件实现各种格式的访问接口,并要随之不断更新,软件开发工作量非常大,难以适应分布动态的网格环境。方法③实现的基础是 OGC 制定的 COM/CORBA/SQL 简单要素实现规范,但由于接口规范还不太成熟,COM, CORBA 等标准间的不兼容以及不能穿越防火墙等缺点,不能满足网格环境下分布异构的空间数据共享与互操作要求。方法④是目前实现网格环境

下空间数据共享和互操作的有效途径。OGC 和 ISO/TC211 协作推出了一系列基于 Web 服务的空间数据服务规范^[4],主要有 Web 地图服务(WMS)、Web 要素服务(WFS)和 Web 覆盖服务(WCS)等。通过空间数据服务的标准接口(如 WMS 的 GetMap, WFS 的 GetFeature 等),以地理标记语言 GML (Geographical Markup Language)^[5]作为空间数据交换标准,就可实现空间数据的共享与互操作。空间数据服务规范正逐渐成为国际标准,提供标准的空间数据服务是一种理想的空

间数据互操作实现方法,但需要在空间数据源的前端(由用户或 GIS 软件商)实现上述各种 OGC 空间数据服务。然而,空间数据服务规范目前还很不成熟,众多的规范之中只有 WMS 成为了国际标准草案(ISO/DIS 19128),而且,作为空间数据交换标准的 GML 在影像及三维空间数据的描述方面还不是很成熟,因此基于空间数据服务规范实现完全的空间数据共享与互操作还有待时日。

本文首先提出了网格环境下空间数据共享与互操作框架,指出了空间数据共享与互操作面临两个关键问题:空间数据表达和空间数据访问与集成,进而探讨了基于 GML/SOAP 的空间数据互操作表达技术,然后通过扩展分布式数

到稿日期:2008-01-22 本文受辽宁省教育厅高校科研计划项目(2006B056),国家 863 计划项目(2006AA09169)资助。

王方雄 博士,副教授,研究方向为网络 GIS、空间信息网格, E-mail: wfxwhu@163.com; 侯英姿 博士生,讲师,研究方向为 GIS 应用; 杨俊 讲师,研究方向为 GIS 应用。

据访问和集成的通用方法——OGSA-DAI(Open Grid Services Architecture - Data Access and Integration)^[6,7]有效地实现了网格环境下的空间数据共享与互操作,并开发了一个原型系统,验证了其可行性。

1 空间数据共享与互操作框架

网格环境下空间数据共享与互操作的实现会涉及到4种角色:数据请求者、数据提供者、数据注册中心和服务提供者,如图1所示。数据提供者拥有可通过网络访问的空间数据,并定义其空间元数据,发布到空间数据注册中心。数据请求者通过空间数据注册中心查询空间元数据,发现所需的空间数据,然后向服务提供者发出请求,调用其网格数据服务,访问数据提供者的空间数据。

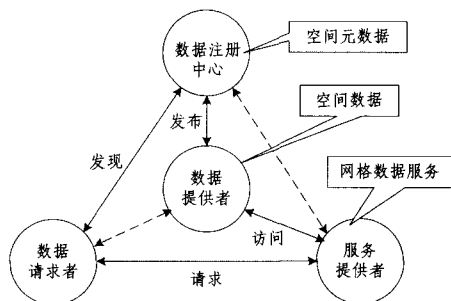


图1 网格环境下空间数据共享与互操作框架

数据提供者:一个可通过网络寻址的空间数据宿主(如空间数据库服务器),它将空间数据发布到数据注册中心(发布内容就是空间元数据),并负责空间元数据的维护与更新。数据提供者接受和执行来自(服务提供者的)网格数据服务的访问与查询等远程操作。

数据注册中心:一个空间元数据目录系统,负责空间元数据的组织、存储与管理,它支持空间数据的发现与发布,扮演着数据提供者和数据请求者之间的数据中介角色。

数据请求者:一个应用程序、组件或网格服务等,它访问数据注册中心,查询空间元数据,发现所需空间数据的数据提供者甚至服务提供者(如果空间元数据同时提供了网格数据服务的信息)。当然,数据请求者也可以从其它途径发现合适的服务提供者。然后,数据请求者向服务提供者发出数据请求。

服务提供者:一个可通过网络寻址的网格数据服务宿主(如网格服务容器),它拥有网格数据服务,接受并执行来自数据请求者的数据请求,并提供网格数据服务去访问数据提供者所提供的空间数据(即空间数据的分布式访问与集成),然后将数据访问与集成的结果返回给数据请求者。

从空间数据共享与互操作框架可以看出,数据请求者、服务提供者与数据提供者之间才是真正发生空间数据互操作的地方。因此,空间数据共享与互操作可以分解为两个关键环节:(1)数据请求者与服务提供者的空间数据表达;(2)服务提供者与数据提供者之间的空间数据访问与集成。

2 空间数据互操作表达

网格环境下空间数据互操作角色(如数据请求者与服务提供者)之间如何有效实现复杂的空间数据的交换,即空间数据的互操作表达,是空间数据共享与互操作首先面临的关键

问题。GML是OGC定义的用来建模、传输和存储地理信息的XML编码^[5],继承了XML跨平台、跨语言的数据传输能力,克服了由于数据模型差异和二进制传输造成的数据封闭性,为网格环境下空间数据共享与互操作提供了数据交换的解决方案。

简单对象访问协议(Simple Object Access Protocol, SOAP)^[8]是Web服务3大核心技术之一。SOAP为在分散、分布式环境中利用XML实现对等地交换结构化、类型化的信息提供了一个简单且轻量级的机制。SOAP是XML编码的一种字符序列消息,接受和传送数据参数也采用XML作为数据格式,是一种可扩展的、与编程语言、平台以及环境无关的网络协议,具有简单轻量 and 能够穿越防火墙两大优势,因而得到了业界的广泛支持。一个SOAP消息(Message)就是一份XML文档,它以信封(Envelope)为框架,以消息头(Header)和消息体(Body)为内容,还可以携带附件(Attachment)^[9]。一个完整的SOAP消息如图2所示。SOAP消息的Header(可选的)提供了一个灵活的框架,以满足应用程序级的附加需求,例如在此提供权限验证、事务管理等信息。SOAP消息的Body(必需的)包含着应用程序的调用信息,要传递的信息通常放在此处。在SOAP消息中,复杂的数据类型如对象、结构、图像、声音等都可以嵌入在消息中,利用XML机制来表达数据。正是考虑到SOAP的优越性,因此可以将空间数据以XML/GML格式嵌入到SOAP消息中进行空间数据交换。

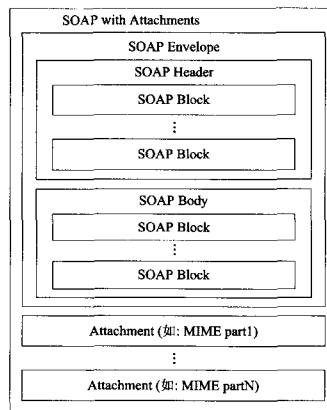


图2 带附件的SOAP消息结构

网格环境下分布异构的空间数据交换不仅包含空间数据本身,还包括其描述性信息,即空间元数据。数据请求者最先关注的就是空间元数据,如空间参考系、数据来源、图层名称、坐标范围等信息,因此可以将空间元数据采用XML格式组织在SOAP Header中。如果空间元数据的内容项比较多,可以适当分组,然后组织在SOAP Header的几个Block中。二维空间数据(如矢量数据和属性数据)用GML已能很好地表达,可以很方便地以图层为基本单元组织在SOAP Body中,一个GML图层对应一个SOAP Body的Block。但如果二维空间数据的图层及要素较多,GML文档将会较大,直接嵌入在Body中会使SOAP消息过于臃肿。而且对于复杂的三维空间数据(如DEM数据、三维模型(如3ds等格式)及纹理数据等)以及遥感影像数据,在最新规范GML3.1中,用于表达它们的Coverage规范^[5]还不成熟,目前用GML还难以很好地表达。而且这些数据的数据量一般都较大,在数据库

中通常都以 BLOB 进行存储,即便是将它们转换为 XML 格式嵌入在 Body 中,也会使 SOAP 消息过于臃肿,为网络传输带来沉重的负担,造成传输时间和解析时间过长。

SOAP 消息实际上是一段 XML 文本,可以使用多种网络协议(如 HTTP,FTP 甚至 SMTP)进行传输。要加速 SOAP 传输,就要减少传输的空间数据量,从两个方面:一是对 GML 数据和复杂空间数据(影像及三维空间数据)进行压缩处理,使 GML 文本以及复杂空间数据更紧凑,减少传输的数据量;二是对压缩后的空间数据采用合适的传输策略。

2.1 空间数据的压缩

GML 的主要目标是建立一种用于网络传输地理信息的 XML 编码语言。但 XML 采用文本格式描述,与二进制数据格式相比,冗余大,数据量大。同样的空间数据,如果采用 GML 描述,大小约为使用二进制数据格式描述的 3~5 倍。若空间数据的非空间属性项很多,还会更高。空间数据本身具有海量性特点,经过 GML 描述,数据变得更大,为网络传输带来了沉重的负担。因此,对于数据量较小的二维空间数据,以图层为单位用 GML 表达,直接嵌入在 SOAP Body 的 Block 中进行传输;而对于数据量较大的 GML 数据文档,则采用标准的 zip 或 gzip 文件方式将其压缩成二进制数据。GML 数据相对于二进制数据的压缩比大约在 10 倍左右,压缩和解压缩时间远远小于大数据量 GML 文档的传输时间^[10]。但 SOAP 消息中不能携带非 XML 格式的二进制数据,因此需要对其进行再处理。Base64^[11]是目前网络上最常见的用于传输 8Bit 字节代码的编码方式之一,它将二进制数据编码为字母和数字。因此,GML 数据压缩为二进制数据后,经 Base64 编码可以构成 XML 文档,然后嵌入在 SOAP Body 中传输。使用 Base64 编码二进制数据实现在 SOAP 中传输的基本步骤为:

(1)服务端读取数据保存在 byte[] 中,使用编码器 Base64Encoder(如 sun.misc.BASE64Encoder)把 byte[] 编码成 String;

(2)客户端发出请求时,就返回这个 String;

(3)客户端从 SOAP 消息中读取这个 String;

(4)使用解码器 BASE64Decoder 把 String 解码成 byte[];

(5)对 byte[] 进行处理,还原为原始数据。

Base64 编码是把 3 个 8Bit 字节的二进制数据转换为 4 个 6Bit 字节的字母和数字,然后把 6Bit 再添两位高位 0,组成 4 个 8Bit 的字节。也就是说,转换后的字符串理论上要比原来的长 1/3。使用 Base64 编码方法来传递 GML 数据,甚至传递影像数据及三维空间数据,主要用于数据集较小的情况。对于大数据集的传输,尤其是遥感影像与三维空间数据的传输显然不现实,这时采用 SOAP 消息包^[9]是唯一的选择。

2.2 空间数据 SOAP 消息包

一个 SOAP 消息包包含一个主体 SOAP 消息,可以携带 0 个或多个附件(见图 2)。SOAP 消息包采用 MIME Multi-part/Related 进行封装,在语义上等同于一个 SOAP 协议绑定^[9]。MIME(Multipurpose Internet Mail Extensions,多用途 Internet 邮件扩展)^[11]最初的用途是在 Internet 上为 e-mail 传递附件。MIME 是一种打包机制,可将所有附件与 SOAP 消息打包在一起,构成 SOAP 消息包进行传输。SOAP 消息

及附件的 MIME 打包程序实现方法主要有两种:JAX-RPC^[12]和 SAAJ^[13]。JAX-RPC 是一种高层次的 API,比 SAAJ 更抽象。它在 RMI 层背后隐藏了大部分面向 SOAP 协议的问题,开发人员处理的是 Java 对象,预处理程序将其转成 SOAP 节点。JAX-RPC 使用 java.awt.Image 和 javax.activation.DataHandler 类表示附件。SAAJ 更接近于协议,使用 SAAJ 创建 SOAP 消息包和 JAX-RPC 相比要做更多的工作(而且没有提供到 WSDL 的自动链接),因此多数情况可能更愿意使用 JAX-RPC。大数据集的 GML 数据、影像数据以及三维空间数据(建筑物、地形等的纹理数据、DEM 数据、3 维模型数据等)采用标准的 zip 或 gzip 文件方式以图层为单位压缩成二进制数据,然后作为 MIME 附件(一个或多个,一个附件对应一个图层)附在 SOAP 消息体后采用 JAX-RPC 或 SAAJ 方法打包在一起。空间数据 SOAP 消息包理论上对附件的数量和大小没有限制,考虑到网络环境下带宽的瓶颈,如果附件数量过多或容量过大,将会造成传输时间和解析时间过长,可以分为多个 SOAP 消息包进行传输。采用 JAX-RPC 方法的三维空间数据 SOAP 消息包(建筑物 3ds 模型及纹理数据)的结果示例,见图 3。

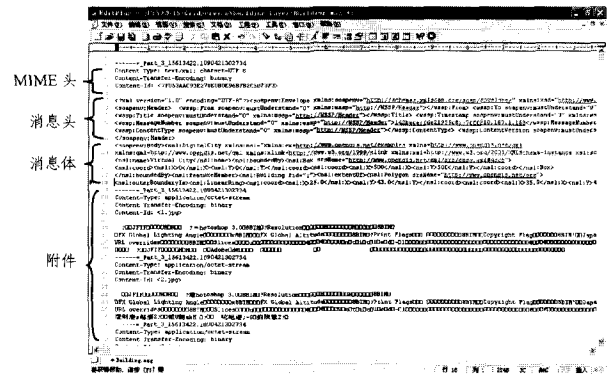


图 3 空间数据 SOAP 消息包的显示结果

3 空间数据访问与集成

空间数据的访问与集成是网络环境下空间数据共享与互操作的第二个关键环节,理想的方法是基于 OGC 空间数据服务规范来实现,但这些规范目前还很不成熟。本文基于 OGSA-DAI 提出了一种新的实现方法。OGSA-DAI 是网络环境中实现分布式数据的访问和集成的一种简便而通用的方法^[6,7],是全球网络论坛(GGF)的数据访问与集成服务(DAIS)工作组正在制定的数据访问与集成规范的一个开放源码实现。由于 OGSA-DAI 是面向通用应用的,只支持关系数据库、XML 数据库和数据文件的访问与集成,但它为特殊应用的数据访问与集成提供了良好的扩展机制。扩展点就是 OGSA-DAI 的活动(Activities),活动是 OGSA-DAI 的核心组件网格数据服务(Grid Data Service, GDS)所支持的数据源操作,包括数据查询与更新操作、数据转换操作、数据传递操作^[7]。目前 GDS 所支持的数据访问与集成的活动有:关系数据库活动、XML 数据库活动、文件活动、数据传递活动和数据转换活动^[7]。因此,在 OGSA-DAI 中增加新的活动就可以实现面向新应用的数据访问与操纵功能。

网络环境下,空间数据的可能存储方式有 3 大类:①空间数据文件,如 Shapefile, Tab, DXF, SDTS, CNSDTF, GML,

JPEG 等格式;②对象-关系数据库,空间数据库引擎(SDE)+关系数据库管理系统(RDBMS),是目前空间数据管理的主流解决方案,主流的 SDE 产品有 ArcSDE,Oracel Spatial 等;③XML 数据库,空间数据以 GML 格式存储于 XML 数据库管理系统(XDBMS),如 eXist。网格环境下空间数据将会主要以空间数据库的形式存在,即 XDBMS 和 SDE+RDBMS 的解决方案。如果空间数据以 GML 格式存储在 XML 数据库中(或以 GML 文件形式存在),则 GDS 可以通过 XML 数据库活动(或文件活动)实现直接的空间数据访问与集成。对于采用 SDE+RDBMS 和文件系统管理的空间数据,则必须对 OGSA-DAI 予以扩展,才能支持对空间数据的访问与集成。因此,在 OGSA-DAI 中增加空间数据访问活动,在活动中实现空间数据访问的接口,就可使 GDS 支持空间数据的访问与集成。主流的 SDE 产品中,ArcSDE 占有绝对的技术优势,应用也最广,下面以 ArcSDE 数据为例来阐述扩展 OGSA-DAI 的具体方法。

3.1 空间数据访问接口的设计

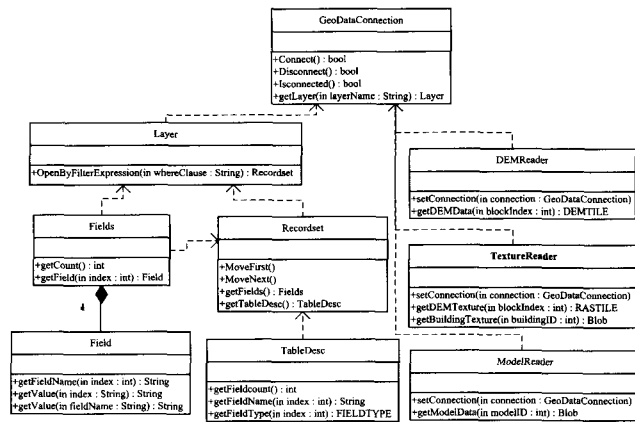


图 4 空间数据访问接口的对象关系图

ArcSDE 是 ESRI 推出的空间数据库管理软件,提供了两种 SDE 服务器:SDE for Coverages 和 SDE for DBMS。前者用于支持 ArcGIS 数据文件(如 Coverage、Shapefile 等)的访问,后者使用商用关系数据库来存储和管理所有空间数据及属性数据。ArcSDE 为客户提供了对空间数据及属性数据进行高效率操作的 3 套数据库接口:Java API,C API 以及 SQL API。活动是 OGSA-DAI 的一个扩展点,针对 ArcSDE 数据源,在 OGSA-DAI 中增加一个空间数据访问活动(GeoSQLQueryStatement),并在活动中实现一套基于 ArcSDE 的 Java API 空间数据访问的接口,就可使 GDS 支持 ArcSDE 管理的空间数据的访问与集成。图 4 是空间数据查询接口的对象关系图,空间数据访问主要通过这些接口来进行操作。数据库连接对象 GeoDataConnection 完成空间数据库的连接操作;图层操作对象 Layer 通过 SQL 语句查询满足条件的记录集;记录集对象 Recordset 完成记录相关操作;字段集对象 Fields 完成字段集的各项操作;字段对象 Field 完成单个字段的相关操作;表结构描述对象 TableDesc 获得表的结构描述。DEM 数据获取对象 DEMReader 完成 DEM 数据的获取操作,三维模型数据读取对象 ModelReader 完成三维模型数据的获取操作,纹理数据获取对象 TextureReader 完成地形纹理(遥感影像)、三维模型纹理(如建筑物外表面的图

像)等的读取操作。

3.2 空间数据访问活动类的设计

空间数据访问活动类(GeoSQLQueryStatementActivity)继承于 SQLActivity 类,见图 5,它通过调用上述各空间数据访问接口的方法来实现空间数据的访问。

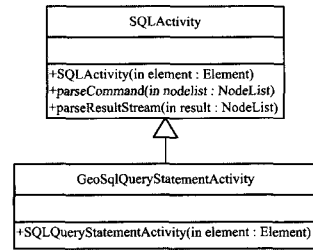


图 5 空间数据访问活动类

3.3 空间数据的集成

空间数据访问活动对空间数据的访问是以图层为单位进行的,客户要访问哪些图层,只需在执行文档中增加相应的空间数据访问活动即可。下面是一个执行文档的实例片段:

```

<gridDataServicePerform
  xmlns = "http://ogsadai.org.uk/namespaces/2003/07/gds/types"
  <documentation>
    Performs two Geospatial SELECT statement.
  </documentation>
  <GeoSQLQueryStatement name="statement">
    <expression>
      select * from District
    </expression>
    <webRowSetStream name="statementOutput"/>
  </GeoSQLQueryStatement>
  <GeoSQLQueryStatement name="statement">
    <expression>
      select * from Road
    </expression>
    <webRowSetStream name="statementOutput"/>
  </GeoSQLQueryStatement>
</gridDataServicePerform>

```

该空间数据访问的执行文档中有两个活动,分别访问同一空间数据库中的两个图层:行政边界层(District)和道路层(Road)。客户提交执行文档,通过 GDS 的 Perform 操作执行各个图层的访问,并将获取的空间数据转换为 GML 格式返回给客户,客户便可从响应文档中得到请求的两个图层的空间数据的集合(GML 格式)。因此,利用 GeoSQLQueryStatement 活动可以实现对 ArcSDE 数据库及 Shapefile, Coverage 等文件的访问。

通过对 OGSA-DAI 的扩展,一个 GDS 中利用多个空间数据访问活动实现对同一空间数据库中多个图层的空间数据的访问与集成。采用多个 GDS 就可实现对多个分布异构的空间数据源的访问与集成。对于不同类型的空间数据源(如不同的 SDE 与空间数据文件),就要对 OGSA-DAI 扩展相应的空间数据访问活动,分别采用相应的 GDS 去访问空间数据,将多个 GDS 获取的 GML 格式的空间数据合并为一个 GML 文档提供给客户。

4 原型系统

基于 GML/SOAP/扩展的 OGSA-DAI 的空间数据互操作方法本质上是基于 Web 服务的互操作方法的一种具体实现。为了验证其可行性,原型系统采用 4 层的 B/S 结构,由客户端(IE+Applet)、Web 服务器(Tomcat5)、网格应用服务器(Tomcat5)以及数据库服务器(ArcSDE+ Oracle9i/SqlServer)构成,异构的二维、三维空间数据采用分布式存储,主机 IP 为 192.168.1.119 的空间数据库服务器(ArcSDE for Oracle9i/RedHat9)中存储湖北省市县行政边界以及道路数据;主机 IP 为 192.168.1.25 的空间数据库服务器(ArcSDE for SqlServer2000/Window XP)中存储湖北省市县行政边界数据和三维空间数据(包括建筑物 3ds 模型与纹理数据、DEM 数据、影像数据)。原型系统基于 Globus Toolkit 3.2.1 和 OGSA-DAI 4.0 开发,将空间数据 SOAP 消息的打包和解包功能封装在 GDS 中,形成相对独立的操作,实现了 2 个扩展的 GDS,并部署在 Tomcat 中,见图 6。客户调用 GDS 去访问与集成后端的空间数据,并使用 GDS 的活动 zipArchive 或 gzip-Compression^[10] 将其压缩为二进制格式,然后调用 GDS 的打包操作按上述空间数据在 SOAP 消息中的组织方式完成 SOAP 消息打包,以字节流的方式返回给客户。客户获得空间数据 SOAP 消息包后,再调用 GDS 以完成 SOAP 消息包的解包工作,解包以后便可得到空间元数据、GML 数据(二维)、三维空间数据等。对于简单的 GML 空间数据就不必调用压缩活动、打包及解包操作。

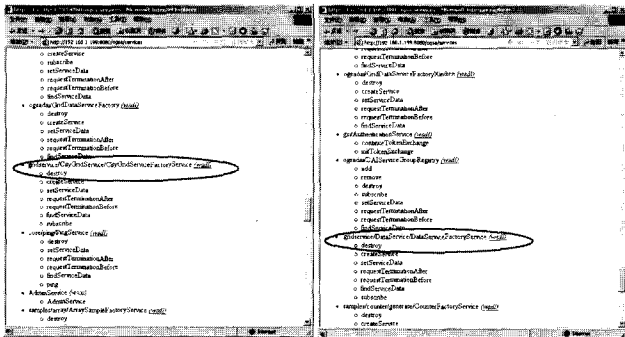


图 6 部署在 Tomcat 中的扩展的网格数据服务

原型系统的客户端采用 Applet 技术,服务器端采用 Servlet 或 JSP 作为 GDS 的客户。对于简单的二维空间数据,服务器端通过 JSP 调用扩展的 GDS 去访问和集成,以 GML 格式返回给客户端,通过网页来表现。图 7 为从两个异构空间数据库中各获取一个图层合并后的显示结果。对于三维

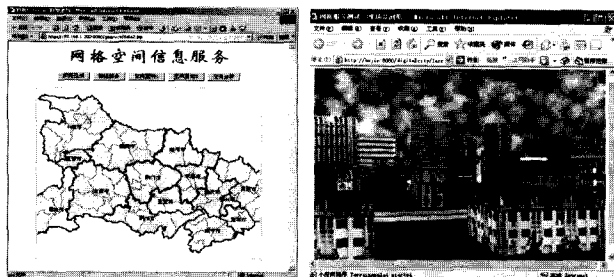


图 7 2 个矢量图层合并后的显示结果 图 8 三维空间数据的显示结果

空间数据,服务器端通过 Servlet 调用扩展的 GDS,获取三维空间数据 SOAP 消息包(包括建筑物 3ds 模型与纹理数据、DEM 数据)。解析数据包后,在 DEM 上通过插算得到建筑物和其它地物特征的高程数据,以参数形式传递给客户端的 Applet,实现 Applet 与 Servlet 的通讯。三维场景的显示采用 Java3D 技术在 IE 浏览器实现,示例结果见图 8。

结束语 本文深入分析了网格环境下空间数据共享与互操作涉及的角色与关键环节,提出了空间数据共享与互操作框架,探讨了基于 GML/SOAP 的空间数据互操作表达技术,通过对网格环境中数据访问和集成的通用方法 OGSA-DAI 的扩展,有效地实现了对分布、异构的空间数据访问与集成,并设计和开发了一个原型系统,验证了基于 GML/SOAP/扩展的 OGSA-DAI 的空间数据共享与互操作技术的可行性。

参考文献

- [1] Levinsohn A. Geospatial interoperability: The Holy Grail of GIS. GeoWorld, Octobe, 2000. <http://www.geoplace.com/gw/2000/1000/1000data.asp>.
- [2] Deren L.I. Is geo-services ready? -- On generalized and specialized spatial information grid// ISPRS Workshop on Service and Application of Spatial Data Infrastructure, XXXVI (4/W6). Hangzhou, China, Oct. 2005. http://www.commission4.isprs.org/workshop_hangzhou/papers/1-6%20Deren%20LI-A020.pdf
- [3] 龚健雅. 空间信息资源共享与互操作技术. 国土资源信息化, 2003(5)
- [4] Open GIS Consortium (OGC). OpenGIS Web Services Architecture (Version 0.3), 2003. <http://www.opengis.org/docs/03-025.pdf>
- [5] Open GIS Consortium (OGC). Geography Markup Language (GML 3.1), 2004. http://portal.opengis.org/files/?artifact_id=4700
- [6] Krause A, Malaika S, McCance S, et al. Grid Database Service Specification (Version 0.2), 2002. <http://www.cs.man.ac.uk/grid-db/papers/DAIS;StatementSpec.pdf>
- [7] OGSA-DAI Project. OGSA-DAI User Guide Index, 2004. <http://www.ogsadai.org.uk/docs/R4.0/doc/>
- [8] World Wide Web Consortium (W3C). SOAP Version 1.2 Part 1: Messaging Framework, 2003. <http://www.w3.org/TR/2003/REC-soap12-part1-20030624/>
- [9] World Wide Web Consortium (W3C). SOAP Messages with Attachments, 2000. <http://www.w3.org/TR/2000/NOTE-SOAP-attachments-20001211>
- [10] 贾文珏, 龚健雅, 李斌. Web 要素服务的优化方法. 测绘学报, 2005, 34(2)
- [11] Internet Engineering Task Force (IETF). Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies. November 1996. www.ietf.org/rfc/rfc2045.txt
- [12] Sun Developer Network (SDN). Java API for XML-Based RPC (JAX-RPC), 2005. <http://java.sun.com/webservices/jaxrpc/index.jsp>
- [13] Sun Developer Network (SDN). SOAP with Attachments API for Java (SAAJ), 2005. <http://java.sun.com/webservices/saaaj/index.jsp>