
中国地质调查局发展研究中心

地质信息技术实验室

项目编号：中地调研【2011】第 219 号

三维地质网格地球关键技术研究 研究成果报告

项目编号：中地调研【2011】第 219 号

项目名称：三维地质网格地球关键技术研究

项目负责人：罗显刚

项目联系人：罗显刚

承担单位：中国地质大学（武汉）

起止时间：二〇一一年十二月~二〇一二年三月

目录

第一章	课题基本情况	1
1.1	课题背景.....	1
1.2	目标与任务.....	1
1.3	引用标准及术语.....	2
1.3.1	引用标准	2
1.3.2	术语	3
1.4	主要工作量及成果.....	4
1.4.1	主要工作量	4
1.4.2	主要成果	5
第二章	项目实施及数据源	7
2.1	项目组织与实施.....	7
2.1.1	项目组织	7
2.1.2	项目实施	7
2.2	数据来源.....	8
第三章	开发环境	9
3.1	MapGIS IGServer 平台介绍.....	9
3.2	MapGIS VirtualEarth 介绍.....	12
3.3	MapGIS VirtualEarth 安装配置流程.....	15
3.3.1	系统环境说明	15
3.3.2	安装配置流程	15
第四章	工作内容与实现	18
4.1	多源数据的三维可视化集成.....	20
4.1.1	多源数据集成方式	20
4.1.2	多源数据集成模型	21
4.1.3	多源数据的集成验证	25
4.2	全国地名数据与三维地球的可视化与查询集成.....	26
4.2.1	POI 数据的网络下载与可视化.....	26
4.2.2	POI 数据的在线查询.....	29
4.3	全国道路分析的三维地球空间分析集成.....	32
4.3.1	请求网络上道路分析服务	33
4.3.2	道路分析结果的可视化	36
4.4	提高三维可视化显示效率方法研究.....	37
4.4.1	矢量图的裁剪	38
4.4.2	矢量图的合并	38
4.4.3	客户端的地图结构	38

4.4.4	客户端和服务端端的交互	39
4.5	地上、地表、地下三维一体化的三维发布系统.....	40
4.5.1	三维建模方式	40
4.5.2	地上、地表、地下三维数据发布	46
4.6	三维地质网格地球的分布式部署.....	48
4.6.1	分布式系统	48
4.6.2	分布式部署方案	49
第五章	质量监控体系与质量保证	51
5.1	组织保证.....	51
5.2	质量评述.....	53
第六章	取得成果	54
6.1	多源数据的三维可视化.....	54
6.2	地名查询.....	57
6.3	道路分析.....	57
6.4	提高三维可视化显示效率.....	60
6.5	地上、地表、地下三维一体化的发布系统.....	61
6.5.1	地上数据发布后可视化	61
6.5.2	地表数据发布过程与可视化	62
6.5.3	地下数据发布过程与可视化	63
主要参考文献与资料		65
附件		

错误！未定义书签。

第一章 课题基本情况

1.1 课题背景

本课题旨在中国地质调查信息网格和北斗野外应急保障系统三维可视化成果基础上,研究并提高三维可视化显示效率的技术方法,开发出集地上、地表、地下三维一体化的中国地质调查信息网格三维发布原型系统,研究并提出三维地球的快速分布式部署方案。将该研究项目(项目编号:中地调研【2011】第219号)委托给中国地质大学(武汉),开展一系列的三维可视化研究工作,实现三维发布原型系统,并提供快速分布式部署方案。

1.2 目标与任务

本课题所提到的三维地球,是在大型国产 WebGIS 软件——MapGIS IGServer 平台基础上,开发出来的网络三维数字地球平台软件(MapGIS VirtualEarth),它提供基于数字地球的三维空间信息服务构建与集成。而三维地调网格地球正是在此三维地球的基础上,融合中国地质调查信息网格研究的成果,将三维地球与国家地质调查信息网格的有效结合,研究其实现的关键技术。

中国地质调查信息网格的构建,采用了先进的云计算技术和网格技术,实现了分布在不同地区的地质调查数据的集成整合、互联互通和“一站式”共享服务。它为中国地质调查信息服务提供了新模式,初步解决了网络环境下数据资源使用权与所有权分离、资源共享、网络环境下矿产资源、地下水资源区域评价的关键技术,对改变传统数据资源共享和地质工作模式,提升地质工作信息化应用水平具有重要的推动作用。

然而,随着科学技术的飞速发展和人类需求的日益增长,国家地质调查信息网格发展到今天,仍然存在一些问题,需要利用新技术来不断地改进与更新。

(1)地质工作的研究对象是地球,地质工作者可通过各种勘查技术获取空中、地面、地下的地质调查数据,而地质调查活动具有三维空间特征和动态特征,特别是复杂的地质调查现象,需要真三维才能对其做出定量化的表达和逼真的描绘,才能更好地理解地质信息的空间信息及其与地表地形之间的空间位置关系,从而提高

空间分析能力。

(2) 随着“数字地球”、“智慧地球”等概念的提出,以及空间信息和遥感数据处理技术的迅速发展,使传统的二维 GIS 技术已经无法满足三维空间信息分析与可视化的需求。其相关技术的飞速发展,大大促进了地球的三维显示技术的更新,为三维地球应用与实时显示的实现提供了条件,也对其提出了更高的要求。

(3) 随着科学技术的快速发展,三维应用技术在地质领域已得到广泛应用。这些应用(特别是其中的体绘制)通常需要处理的数据量巨大,并且要求达到实时的用户交互效果,因此面对日益膨胀的数据和高度复杂的计算模型,传统的利用本地资源或单个资源进行可视化的模式已经不能满足实际的需求。

(4) 三维空间信息在互联网上发布时,海量数据的传输又成为另一个大的难题。互联网的带宽有限,要想发布大数据量的三维空间信息,现有三维模型无法保证快速高效的传输。如何建立高效的三维数据模型,以便客户端快速访问和下载,也是一个技术难题。

(5) 目前,国内外各大厂商开发的三维地球软件众多,但仍然无法达到各行业空间信息共享的需求,同时“智慧地球”、“智慧城市”等概念的提出,急需整合各类空间信息资源,完善三维空间信息服务。

“三维地调网格地球关键技术研究”可以为解决这些问题提供较好的研究思路。三维地球的诞生弥补了国家地质调查信息网格无法处理三维空间信息数据的不足,而国家地质调查信息网格的出现为三维地球的应用带来了新的契机。三维地球与国家地质调查信息网格的有效结合,能够对多源、多尺度、多模式的三维地质调查信息进行表达,从而实现多源地质数据的三维实体的实时显示、虚拟地上、地面和地下生产活动实景的功能,能够对地质调查工作进行更加科学地指导。

1.3 引用标准及术语

1.3.1 引用标准

- (1) 《基础地理信息要素分类与代码》, GB/T13923-2006
- (2) 国家基本比例尺地形图分幅和编号》, GB/T13989-92
- (3) 《软件工程术语》, GB/T 11457-1989
- (4) 《计算机软件开发规范》, GB/T 8566-1988

-
- (5) 《计算机软件产品开发文件编制指南》，GB/T 8567-1988
 - (6) 《计算机软件需求说明编制指南》，GB/T9385-1988
 - (7) 《信息技术软件生存期过程》，GB/T 8566-1995
 - (8) 《软件文档管理指南》，GB/T 16680-1996
 - (9) 《计算机质量保证计划规范》，GB/T 12504-1990
 - (10) 《计算机配置管理计划规范》，GB/T 12505-1990
 - (11) 以太网 10Base-T 标准 IEEE802.3
 - (12) 以太网 100Base-T 标准 IEEE802.3u
 - (13) 千兆以太网标准 IEEE802.3Z
 - (14) 国际标准 ISO/IEC11801
 - (15) IEEE802.3、IEEE802.3u、IEEE802.3z 、IEEE802.3x、IEEE802.3ad、IEEE802.3ae、IEEE802.3an、IEEE802.1p、IEEE802.1x、IEEE802.3ab、IEEE802.1Q
 - (16) 生成树协议 802.1d、802.1w、802.1s
 - (17) 管理协议 SNMPv12、Web(JAVA)、CLI(Telnet /Console)、RMON(1, 2, 3, 9)
 - (18) 《信息安全技术信息系统安全管理要求》 GB/T 20269-2006
 - (19) 《信息安全技术网络基础安全技术要求》 GB/T 20270-2006

1.3.2 术语

(1) 空间信息网格：简称 SIG,是一种汇集和共享地理上分布的海量空间信息资源，对其进行一体化组织与处理，从而具有按需服务的，强大的空间数据管理能力和信息处理能力的空间信息基础设施。

(2) 数据集成：数据集成是把不同来源、格式、特点性质的数据在逻辑上或物理上有机地集中，从而为企业提供全面的数据共享。

(3) POI 数据：POI 是“Point of Interest”的缩写，可以翻译成“兴趣点”，每个 POI 包含四方面信息，名称、类别、经度、纬度。

(4) 分布式：所谓分布式就是指数据和程序可以不位于一个服务器上，而是分散到多个服务器，以网络上分散分布的地理信息数据及受其影响的数据库操作为研究对象的一种理论计算模型。

1.4 主要工作量及成果

1.4.1 主要工作量

本课题的主要研究内容包括:研究多源数据的三维可视化集成(包括 Google Earth、Virtual Earth、天地图等网络信息数据源);研究并实现全国地名数据与三维地球的可视化与查询集成;全国道路分析结果的三维地球集成;在北斗野外应急保障系统三维可视化的基础上,使用矢量数据的动态裁减瓦片技术,改进并提升其显示效率。最终实现地上、地表、地下三维一体化的中国地质调查信息网格三维发布系统,并提交三维地球的快速分布式部署方案。

1、多源数据的三维可视化

三维地调网格地球的可视化集成所依赖的是网络多源数据,既有图形形式,又有文字形式,且图形文字的类型也有多种,因此如何从各种形式的数据中提取所需的规范、统一的数据格式是一个值得研究的问题,同时在三维地调网格地球之上集成现有的多种多源网络数据,以便补充现有网格地球的可视化表达的基础数据的缺失。第四章节将介绍多源数据的集成方式以及实现的效果。

2、全国地名数据的三维可视化集成

在三维地球可视化基础上,可根据业务需要,集成全国地名数据,包括各种相关的经济、社会、人文等不同类别的 POI 数据,实现 POI 数据的整合与分析,并以三维全景可视化的方式进行展现,并且实现全球地名数据的查询与分析,能够为以后的地质调查规划工作提供科学论证、科学决策等服务提供数据支撑。第四章节将从地名数据的存储方式、可视化以及查询的实现这三个方面进行介绍。

3、全国道路分析的三维空间分析集成

在地质调查进行工作中,往往需要用到应急资源的调度、选择地基的路径等服务,此时可在三维地球上运用网络上的全国道路分析结果的集成,对空间对象进行三维空间分析和操作,用户则可在客户端实现其管理复杂空间分析能力,并在三维地球上实现实时漫游,实现实时道路信息与资源定位信息建立协同调度与信息共享一体化成为可能,同时对提高应急响应能力、提升应急救援决策支持系统的科学性具有重要的意义。

4、提高三维可视化显示效率方法研究

在地质调查信息网格三维地球可视化的基础上,如在北斗野外应急保障系统中,对于如何提高三维可视化显示效率是一个需要研究的方面。为适应这种情况,本课题运用一种矢量数据的动态裁减瓦片技术,基于瓦片分割的矢量数据的合并机制,采用分片传输矢量数据的方法,加之 Ajax 异步请求的功能,能够实现矢量地图的高效传输和动态生成。它使客户端只异步请求可视范围内的地图数据,实时显示时,只需拣取可视范围内的场景图像,从而减少数据的传输来实现更快的客户响应,减少客户端用户的等待时间,提高用户的体验。

5、地上、地表、地下三维一体化的三维发布系统

目前的三维 GIS 平台大多数只停留在可视化方面,能够提供三维空间分析的商业化三维 GIS 平台屈指可数,而且这些平台一般只针对地上环境、地表景观、地下地质环境中某些单方面的管理或模拟,尚未出现真正将地上、地表及地下整个真三维空间相结合,进行综合管理、可视化及分析的三维 GIS 平台。我们在第四章从三维建模方式、地上景观建模及可视化、地形建模及可视化、地下管线等建模及可视化这 4 个方面分别介绍。

6、三维地质网格地球的分布式部署

为了提高整个系统的安全性、稳定性和可维护性,降低信息化管控的难度;提高有效信息的共享程度,信息传递的真实性、可靠性和及时性得到保障,减少信息遗漏和信息阻塞现象,及时互通,避免信息孤岛。研究如何利用三维地球,调用分布在全国的地质调查网格信息,在全国各地设置部署,充分利用信息资源,提出一种快速分布式部署方案。在第四章中提出的分布式部署方案实现了广域网上的数据分布式部署,局域网上的 GIS 服务器和 web 服务器的分布式部署。

1.4.2 主要成果

本课题的研究成果如下:

- 1、提供一套完整的研究模型,包括多源数据的三维可视化集成、全国地名数据的三维集成、路径分析、瓦片动态裁图等关键技术研究模型,提交形式为文档与代码;
- 2、提供一套地上、地表、地下三维一体化的网络地球系统原型,提交形式为文档与代码;

-
- 3、提供三维地球的快速分布式部署方案；
 - 4、将最终所有资料刻盘包括：文档、代码、最终研究报告等，并提交最终研究报告的打印版本。

第二章 项目实施及数据源

2.1 项目组织与实施

2.1.1 项目组织

研究计划：

- 1、2011 年 10 月 1 日以前：研究中国地质调查网格的业务流程；
- 2、2011 年 10 月 1 日-11 月 30 日：结合三维地球模型，研究多源异构数据的三维可视化集成，并实现全国地名数据与三维地球的可视化与查询集成，以及全国道路分析结果的三维地球集成；
- 3、2012 年 12 月 1 日-1 月 31 日：在北斗野外应急保障系统三维可视化的基础上，改进并提升其显示效率，实现矢量数据的动态裁减瓦片，最终实现地上、地表、地下三维一体化的中国地质调查信息网格三维发布系统；
- 4、2012 年 3 月 31 日前，提交三维地球的快速分布式部署方案，并提交课题最终研究报告。

2.1.2 项目实施

项目实施过程分为以下几个阶段进行：

第一阶段：

2011 年 11 月 1 日以前收集课题研究内容的有关资料。包括多源数据集成研究，研究得出多源数据的集成方式及其实现方法；利用网络地图服务，实现地名数据查询与道路分析功能，并集成到三维地球上。

第二阶段：

（从第一阶段末）到 2012 年 2 月 1 日，完成在已有的三维地球的基础上，研究矢量数据的动态裁剪瓦片相关技术，改进并提升三维地球的可视化效率，最终实现数据的动态裁剪，提高数据的显示效率。

第三阶段：

（从第二阶段末）到本年 3 月 10 日，完成三维地球上全球地表、地上建筑设施、

地下地质层状模型数据的发布与可视化，实现三维一体化的三维发布系统。

第四阶段：

（从第三阶段末）到本年 3 月 20 日，完成三维地质网格地球的快速分布式部署方案；

第五阶段：

（从第四阶段末）到本年 4 月 1 日，完成测试并提交测试用例、测试报告和最终研究成果报告。

2.2 数据来源

本课题的研究对象是在中国地质调查信息网格平台的基础上，实现数据的三维可视化和空间分析集成。系统最终处理的数据是在中国地质调查信息网格平台上处理的地质调查成果，但在课题的研究过程中，使用的数据主要是从网络上下载的不同来源的数据，如天地图数据、google 地图数据，以及使用 MapGIS 平台生成的瓦片数据和地图文档数据。

第三章 开发环境

三维地质网格地球关键技术研究项目基于 Java 体系进行开发，具体开发语言与开发环境参见表 1 所示。

表 1 系统开发环境

类别	基于 Java 体系开发环境
操作系统	Windows Server 2003 及以上系列
Java 虚拟机	J2SDK6.0
Web 服务器	Tomcat6.0 及以上版本
开发工具	MyEclipse8.6
数据库	MicroSoft SQL Server 2005
GIS 平台	MapGIS IGServer MapGIS VirtualEarth
浏览器	IE6.0 及以上版本（Adobe Flash Player Active 10）

3.1 MapGIS IGServer 平台介绍

MapGIS IGServer 是新一代互联网 GIS 平台，是 MapGIS IMS 的升级版。MapGIS IGServer 平台底层依托超大型的地理信息系统平台——MapGIS K9，秉承其优势，在此基础上融入互联网技术，提供一个功能全面、开发简便、特点突出、界面美观的互联网地理信息系统开发平台。较 MapGIS IGServer 以前各版本，MapGIS IGServer 平台在各方面均有较大改进，并在性能、效率上有较大突破，可满足各行业各层次应用需求，给用户带来全新的体验。

MapGIS IGServer 作为互联网 GIS 服务发布单元，可支持分布式架构、跨平台调用，可部署在 Windows、Linux、Unix 和 AIX 等操作系统上，支持 .NET 和 Java 两套开发体系，支持 JavaScript、搭建式、富客户端 Flex、Silverlight 开发四种技术，支持 OGC 服务、REST 服务，提供真三维服务。

1. MapGIS IGServer 体系架构

MapGIS IGServer 的体系架构如下图所示：

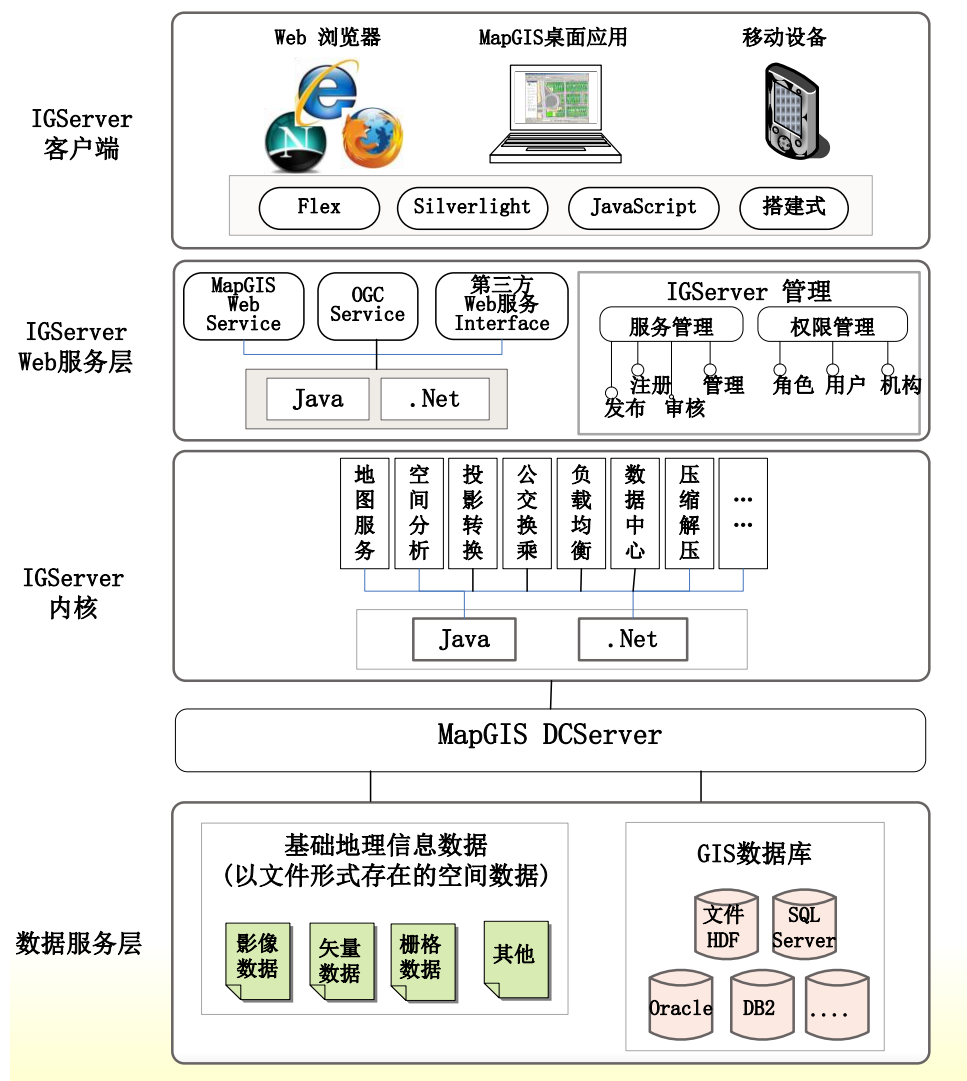


图 1 MapGIS IGServer 体系架构

➤ IGServer 客户端

支持多种 Web 浏览器（如 IE、Firefox 等），支持各种 Web 应用程序的访问或嵌入到已有 Web 应用程序中，同时支持 MapGIS 桌面应用和嵌入式移动设备开发。在客户端层面上，可支持四种开发方式，包括 Flex、Silverlight、JavaScript 和搭建式开发方式。用户通过客户端与 Web 服务层进行交互。

➤ IGServer Web 服务层

运行于 Windows/Linux/UNIX 等操作系统上，主要提供各种 Web 服务，包括 MapGIS WebService、OGC WebService 和第三方 Web 服务接口，其中 MapGIS WebService 和 OGC WebService 分别提供 .NET 和 JAVA 两个不同的版本。客户端通过浏览器或者其他的方式（桌面应用等）向 Web 服务发送请求，Web 应用服务进行响应并接收请求，返回相应的操作结果。

➤ IGServer 内核

主要负责与数据服务层的数据通信，主要提供.NET 和 JAVA 两种版本 MapGIS IGServer 内核。客户端发送数据请求，通过 IGServer 内核实现与数据服务层的通信，将数据返回到客户端缓存。基础地理信息数据和数据库中存储的数据可以通过 GIS 服务器通信处理数据请求，将处理后的结果返回给客户端。

➤ 数据服务层

包括 GIS 数据库和基础地理信息数据。基础信息地理数据包括：影像数据、矢量数据、瓦片数据等，它们都是以文件形式存放的空间数据。GIS 数据库中包含以统一的 MapGIS 数据格式——HDF 方式进行存储的 GIS 数据以及其他数据库存储的数据（例如 DB2、Oracle 等）。MapGIS IGServer 的数据调用都是基于 MapGIS K9，这充分发挥了平台管理海量数据能力和并发访问数据能力。

2. MapGIS IGServer 平台特性

➤ 分布式架构

系统是一个分布式架构，用户和服务器可以分布在不同地点和不同的计算机平台上，并可实现不同 GIS 平台的空间数据互操作。

➤ 跨平台调用

系统支持 Windows、Linux、Unix 和 AIX 等操作系统，提供 Windows+Unix 的 GIS 解决方案。

➤ 两种开发体系

系统提供.NET 和 JAVA 两种开发模式，并支持多种二次开发方式，用户可根据项目选择相应的开发模式与二次开发方式。

➤ 四种二次开发技术

MapGIS IGServer 平台以“简便、易用、高效”原则为二次开发主导思想，提供四大二次开发方式：JavaScript、Flex、Silverlight、搭建式，可覆盖初、中、高级开发者以及小、中、大型项目的开发需求。

(1) 以 J2EE 和.NET 两大开发体系为依托，客户端则采用 JavaScript（Ajax）方式进行编码式开发，集成性强、灵活性大、扩展性高、兼容性好。

(2) 全面支持 Flex 的二次开发方式，即在客户端采用功能强大、优势突出的 Flex 实现。采用 Flex 作为客户端，以用户为中心，为企业级 WebGIS 应用提供强大可靠的支持，为用户提供个性化 WebGIS 服务和丰富绚丽的视觉体验。

(3) 采用主流的富客户端模式，基于开发简便的 Silverlight，全面整合开发接口，功能全面，开发简便，界面美观。这与 Flex 的二次开发方式基本相似，只是在客户端采用 Silverlight 进行开发，这一点与 Flex 不同。同样可以为用户提供个性化 WebGIS 服务和丰富绚丽的视觉体验。

(4) 基于搭建的二次开发方式，运用 Silverlight、Flex 等新技术，结合功能仓库和数据中心的设想，在实现海量数据管理和空间信息共享的基础上，真正做到快速搭建和零编程。

➤ **支持 OGC、REST、SOAP 等多种服务类型**

MapGIS IGServer 采用面向服务的体系架构（Service-Oriented Architecture, SOA），SOA 强调使用服务封装不同的功能单元，服务所暴露的接口通过契约规定其功能性和非功能性的作用和特征，从而实现在广域网络（如 Internet）环境下的业务集成和互操作，而不受平台环境的限制并易于重用。

MapGIS IGServer 可提供的服务类型如下：

(1) REST 服务。基于 REST 的架构以资源形式提供 GIS 功能接口。包含目录服务、制图服务、数据服务、 workflows 服务等。

(2) SOAP 服务。基于 SOAP 服务直接在浏览器客户端上构建应用。如目录服务、制图服务、数据服务、 workflows 服务等。

(3) OWS 服务（OGC Web Service 服务）。OGC 标准服务，如 WMS、WFS、WCS 等。

3.2 MapGIS VirtualEarth 介绍

MapGIS VirtualEarth（简称 MVE）是一套显示 GIS 三维数字虚拟地球的软件，目前已集成到 MapGIS IGServer 中。MapGIS VirtualEarth 支持在浏览器上展示，支持海量矢量、栅格、三维数据一体化发布，支持 google、天地图、Bing、雅虎等数据，并具备地形分析和场景模拟功能。

1. 地球基本显示与浏览

以地球为主体，显示整个地球的三维模型，级数放大，则分级显示地球表面三维影像图片，用户可以任意转动地球，显示任意区域的三维影像图片。可结合提供的放大、缩小、移动、旋转、倾斜角控制、复位等功能进行三维地球的漫游。

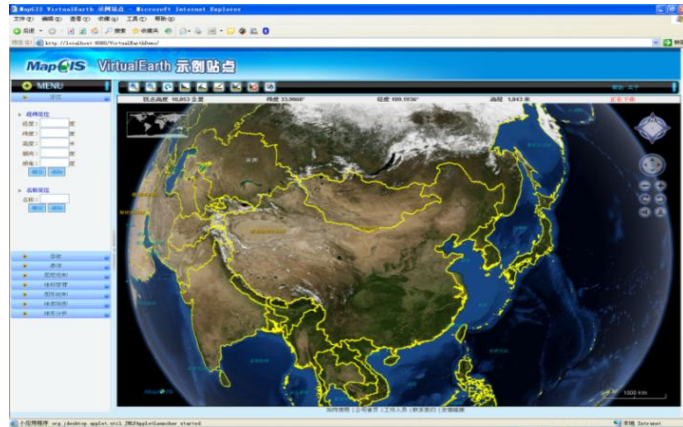


图 2 地球显示

2. 支持海量矢量、栅格、三维数据的一体化发布

MapGIS VirtualEarth 提供全面的二三维一体化网络服务，支持海量矢量、栅格、三维数据的一体化发布、存储、可视化、查询功能，可实现海量、分布式二三维空间信息的管理与发布。

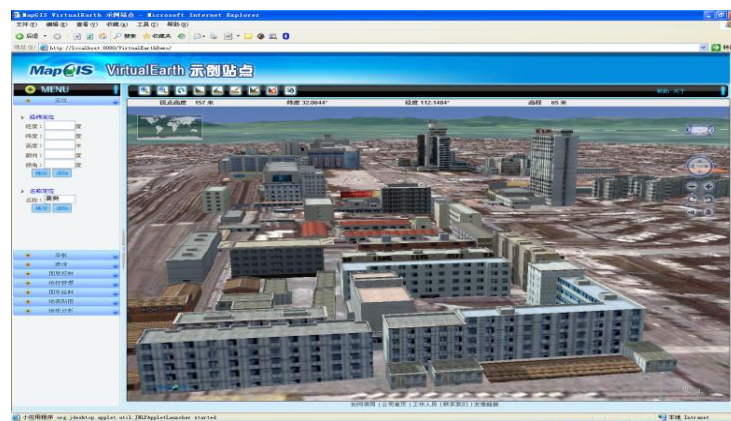


图 3 精模加载与显示

3. 支持 google、天地图、Bing、雅虎等数据

MapGIS VirtualEarth 可实现不同 GIS 厂商、多源异构数据集成，支持 OGC 服务，支持 google、天地图、Bing、雅虎等 API 与数字地球的无缝集成。



Google



天地图



Bing



雅虎

图 4 支持 google、天地图、Bing、雅虎等数据示意图

3. 提供真三维服务

提供三维地球显示、自定义海量数据发布、导航漫游、真三维模型、三维景观等。



三维地球显示



自定义海量数据发布



真三维模型

图 5 支持真三维服务示意图

4. 分析与模拟功能

包括地形分析和场景模拟等，该部分功能可根据需求进行自定义扩展。目前实现的具体功能包括地形剖面图绘制、坡度图绘制、通视情况分析、水势蔓延分析、雷达扫描模拟、气象云图模拟等。

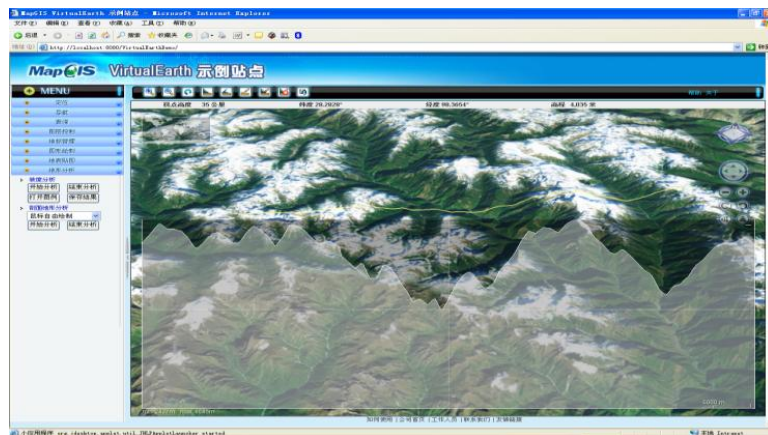


图 6 地形剖面图绘制

3.3 MapGIS VirtualEarth 安装配置流程

3.3.1 系统环境说明

1、支持操作系统

Windows 2003 及以上系列：包括 Windows Server 2003（2008）、Windows XP、Vista、Win 7 等系列版本。

2、磁盘空间需求

软件需要约 2G 的磁盘空间，推荐使用除系统盘外的磁盘，如果要存放地球源数据，则需要另外准备 40G 的磁盘空间。

3、内存需求

最小使用 512M 内存，推荐使用 1G 以上的内存。

4、显卡要求

客户端对显示加速卡的要求较高，显卡配置不好会影响地球的效果。推荐使用 512M 以上显存的独立显卡机器，不支持虚拟机。

5、JAVA 系统环境

MapGIS VirtualEarth 数据服务器必须运行在安装了 Tomcat 6.0、JDK1.6 及以上的服务器。

3.3.2 安装配置流程

1、产品安装包说明

MapGIS VirtualEarth 产品包中包括安装盘，硬件加密狗、产品使用协议，使用手册等。开始使用前请确保配件的完整，并熟悉许可协议条款。

MapGIS VirtualEarth 以基础平台+增量包的方式发布，产品安装包中包括 MapGIS K9 证书安装包、MapGIS K9 基础平台（MapGIS K9 SP2 企业版）安装包、MapGIS IGServer 增量包、MapGIS VirtualEarth 增量包等，安装文件均为 exe 文件（setup.exe）。MapGIS IGServer 增量包中提供地图数据、空间分析数据库、MapGIS IGServer 服务等。MapGIS VirtualEarth 增量包中提供全球地图数据、二次开发库、配置工具、示例站点等。

使用硬件加密狗（或网络狗）、证书服务安装文件（使用网络狗时不需安装）、

MapGIS K9 基础平台和 MapGIS VirtualEarth 增量包安装文件即可进行安装。根据版本环境的不同，主要有几种安装方式：

a) 学习版：学习版证书服务+MapGIS K9 SP2 企业版+MapGIS IGServr 增量包+MapGIS VirtualEarth 增量包；

b) 正式版（本地狗）：正式版证书服务（硬件加密狗）+MapGIS K9 SP2 企业版+MapGIS IGServr 增量包+MapGIS VirtualEarth 增量包；

c) 正式版（网络狗）：MapGIS K9 SP2 企业版+MapGIS IGServr 增量包+MapGIS VirtualEarth 增量包。

2、安装过程说明

（1）安装证书服务程序

a) 关于 MapGIS 许可证服务

必须在 MapGIS 许可证服务器的支持下，才能正常安装、使用 MapGIS IGServr 产品。MapGIS 许可证服务器可以安装配置在您的网络中任何一台装有 MapGIS K9 USB 加密狗的机器上。每一个加密狗都对应一组唯一的安装密码，在安装许可证服务器时必须输入正确的安装密码。

许可证服务器可以单独安装在服务器上，也可以和 MapGIS K9 应用程序一起安装在服务器上。

- 正式版：安装使用 MapGIS IGServr 平台的正式版，必须有一个支持 MapGIS IGServr 平台功能的 USB 硬件加密狗，然后用该加密狗安装正式版的 MapGIS K9 许可证服务器。或者局域网中已有多节点的 MapGIS K9 许可证服务器。

- 学习版：安装使用 MapGIS IGServr 平台的学习版，不需该加密狗，直接安装学习版的 MapGIS K9 许可证服务即可。

b) 正式版证书服务程序安装

正式版证书服务程序需要与中地公司发行的硬件狗配套使用，与硬件狗一起发行的还有硬件狗的安装密码，安装前请确认是否被包含到安装产品包，确认无误后进行安装。

c) 学习版证书服务程序安装

找到安装光盘中的学习版许可证服务器安装程序后，双击“setup.exe”文件进行安装。

（2）安装 MapGIS K9 基础平台

按上面所介绍的步骤安装好证书服务程序后,就可以开始安装 MapGIS K9 基础平台,即 MapGIS K9 SP2 企业版。

单机安装:安装前要确保 MapGIS K9 平台的 USB 加密狗已插在电脑上,且许可证服务处于启动状态,然后放入安装光盘到计算机光驱中(或者拷贝安装包到电脑硬盘),运行安装光盘中 MapGIS K9 SP2 企业版安装包的 setup.exe,启动安装向导。

用网络狗安装:安装前要确保网络连通,且网络狗服务器上的证书服务处于启动状态,然后把安装光盘放入计算机光驱中(或者拷贝安装包到电脑硬盘),运行 MapGIS K9 SP2 企业版安装包的 setup.exe 即可。

(3) 安装 MapGIS IGServer 增量包

安装好 MapGIS K9 SP2 企业版后,就可以开始安装 MapGIS IGServer 增量包了。MapGIS IGServer 增量包有 32 位和 64 位系统的安装文件,要根据系统环境选择相应的安装文件进行安装。MapGIS IGServer 增量包将自动安装到 MapGIS K9 SP2 企业版的目录下。

安装好 MapGIS K9 SP2 企业版,紧接着安装 MapGIS IGServer 增量包。打开安装光盘(或电脑硬盘)的安装包,运行 MapGIS IGServer 增量包中的 setup.exe 即可。

(4) 安装 MapGIS VirtualEarth

现在就可以开始安装 MapGIS VirtualEarth 了。运行安装光盘中 MapGIS VirtualEarth.exe 即可。



图 7 安装文件

第四章 工作内容与实现

本课题的系统设计与实现采用经典的多层软件体系构建，不但在逻辑上划分了各个模块的功能和相互之间的关系，而且在物理上实现了各模块的真正独立：它有四层体系构架，从下到上依次分为客户端、Web 服务层、GIS 服务层、数据层，每个模块都可以单独维护和升级更新。

三维地球的 GIS 服务层既可以部署在一个物理服务器 PC 机上，也可以部署在多个服务器 PC 机群上。通过 Web 服务层的扩展，GIS 服务层提供并行的 GIS 服务请求，也可以执行不同的任务或服务请求，如图给出了 MVE 的体系架构，水平方向表示同一个层次的逻辑单元，垂直方向表示各模块之间的交互关系。

(1) 客户端：可以支持多种 Web 浏览器，如 Microsoft IE、Mozillo FireFox 等；支持各种 Web 应用程序的访问或嵌入到已有 Web 应用程序中；支持桌面应用程序；支持移动应用程序等。Web 用户通过浏览器向 MVE Applet 服务进行交互，MVE Applet 提供与 MapGIS VirtualEarth Web 应用服务的交互，响应用户的 GIS 请求。在客户端一般要求安装 J2SDK1.6。

(2) Web 服务层：运行于 Windows 或 Linux 的操作系统上。它包括 MapGIS VirtualEarth Web 应用服务、数据服务器以及 MVE 配置工具。MVE 应用服务接收客户端发送的 GIS 请求，并返回相应的操作结果，并与数据服务器进行连接，把数据缓存到客户端，实现在浏览器中对三维地球数据的渲染表现。数据服务器接收 MVE Web 应用服务的请求，并向 GIS 服务层发送数据请求，最后将返回的数据通过 MVE 应用服务返回给客户端。MVE 配置工具：可以对 Applet 客户端站点、MVE Web 应用服务、数据服务器三者的配置文件进行修改或配置。

(3) GIS 服务层：主要负责 Web 服务层与数据层的数据通信，实现多个 GIS 服务器的负载均衡功能（即群集功能）。它主要包括 GIS 地图引擎服务和负载管理服务，即 MapGIS IGServer 应用服务。用户可以在单机上使用 GIS 地图引擎服务，也可以通过负载管理服务同时在多台 PC 机上进行数据通信，这样的体系结构充分利用设备，提高设备的利用率，同时也提高用户的并发访问量。

(4) 数据层：包括 GIS 数据库和基础地理信息数据。基础信息地理数据包括：地表数据、国界、地名、三维建模数据等基础地理数据，它们都是以文件形式存放

的空间数据。MapGIS VirtualEarth 是依托于 MapGIS 基础平台，GIS 数据库中包含的栅格/影像数据、高程数据、矢量数据等都是统一的 MapGIS 数据格式 HDF 方式进行存储，基于 MapGIS，可以充分发挥平台管理海量数据能力和并发访问数据能力的优势，构建大型的网络 GIS 系统。

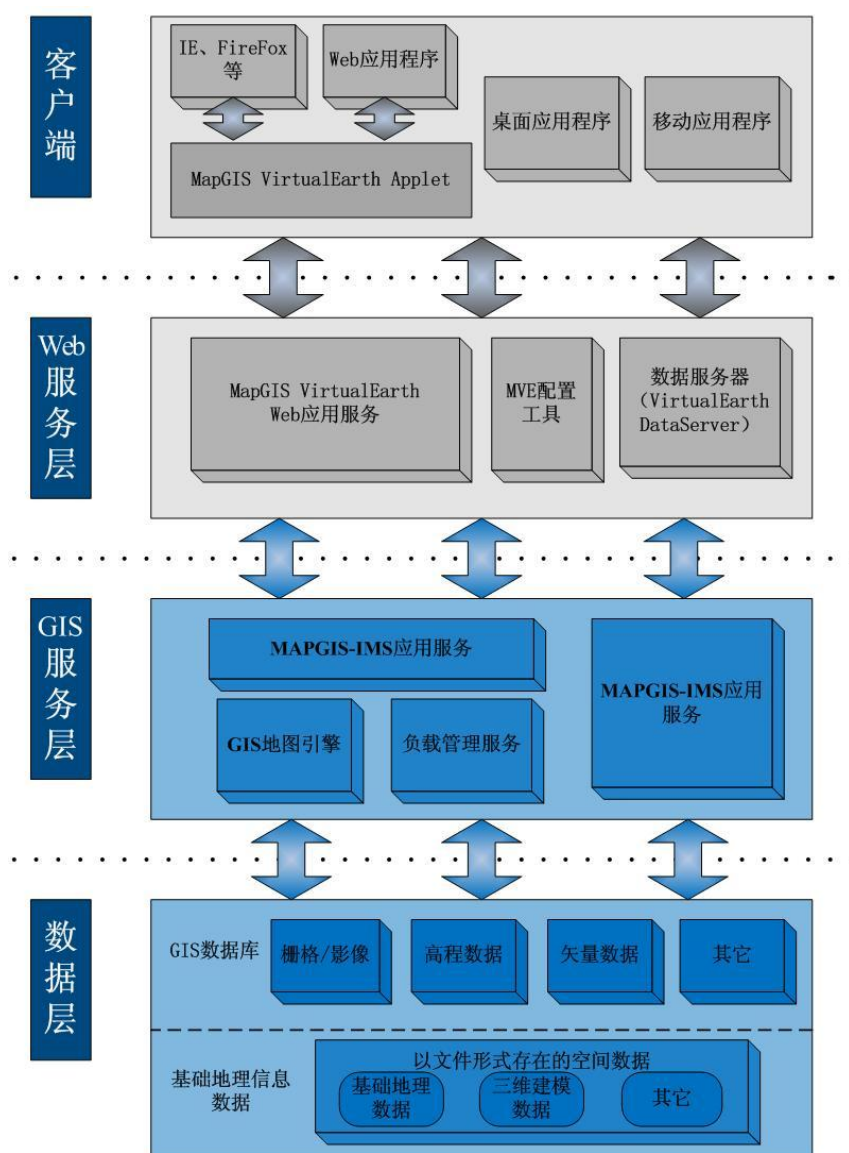


图 8 MVE 体系架构图

在以上四层结构的体系基础上，本课题将从多源数据的可视化集成，全国地名数据与三维地球的可视化与查询集成，全国道路分析的三维地球空间分析集成，提高三维可视化显示效率方法研究，地上、地表、地下三维一体化的数据发布、三维地质网格地球的分布式部署方案六个部分分别展开研究，并给出相应的实现。

4.1 多源数据的三维可视化集成

目前在线网络地图得到了广泛应用。不同的在线地图服务，在地图投影方式、地图层级范围、地理编码方式等方面存在差异，从而导致在线地图服务间彼此独立，难以集成与共享。

传统的 GIS 数据集成模式无法真正地满足在线异构地图服务对数据集成的无缝性、低冗余性及高效性的需求。我们在分析异构在线地图服务数据组织方式的基础上，提出基于全局虚拟金字塔结构模型集成在线异构地图服务数据的方法。

4.1.1 多源数据集成方式

异构地图数据的集成模式主要有：

1、通过数据转换的方式

数据转换模式是传统 GIS 数据集成方法，其他数据格式经过专门的数据转换进行格式转换后，复制到当前系统中的数据库或文件中。文件存储的数据需要按照一定的标准格式文件进行存储管理，将图形描述数据和属性数据分别存放。数据库包括关系数据库、GIS 数据库。关系数据库主要存储空间对象的属性数据和图形数据，实现了空间数据和属性数据一体化存储和管理，解决了文件系统图属分离的问题。而这些数据都可分别存放在不同类型的数据库如 SQL Server、Oracle、DB2 中，提高了数据的共享程度。这种方式是基于数据文件的转换实现的，在文件转换的过程中，由于不同数据格式在描述空间对象时采取的数据模型不同，几何要素的定义不同，对拓扑的支持程度不同，所以很容易产生数据精度的丢失。

2、通过直接访问的方式

这种方式是基于二次开发语言实现的，依赖于对异构空间数据组织结构的认识，但其重复性的数据接口的创建增加了开发的难度，降低了工作的效率。

3、通过数据组件中间件的方式

从理论上讲，这种方式由地图服务商提供屏蔽空间数据异构性的数据驱动，但目前地图服务商一般以提供自己的数据接口为主，对其他数据格式的支持不够；组件技术虽然能实现组件间的集成，但组件技术不支持跨平台的异构环境，这些使集成效果和应用范围受到限制。

4、数据互操作的方式

数据互操作模式是指在异构数据库和分布式计算的情况下，GIS 用户在相互理解的基础上，能够透明地获取所需的信息。在 web 服务中提供了解决数据互操作的 OGC 服务。GIS 互操作模式是由 OGC 指定的统一规范，使得系统同时支持不同的空间数据格式。根据 OGC 规范，把提供数据源的软件成为数据服务器，使用数据的软件成为数据客户，数据客户使用某种数据的过程就发出数据请求，有数据服务器提供服务的过程，使数据客户能够读取任意数据服务器提供的空间数据，从而实现不同来源的数据的集成。

4.1.2 多源数据集成模型

1、异构地图服务的集成

为了明确地划分异构地图服务的数据集成环节，在数据集成模型框架设计方面，我们采取了 3 层结构的设计模式，即自底向上，建立从局部概念到全局概念金字塔结构模型间的映射，使异构在线地图服务最终能通过统一的数据结构模型进行表达。模型框架如图所示。

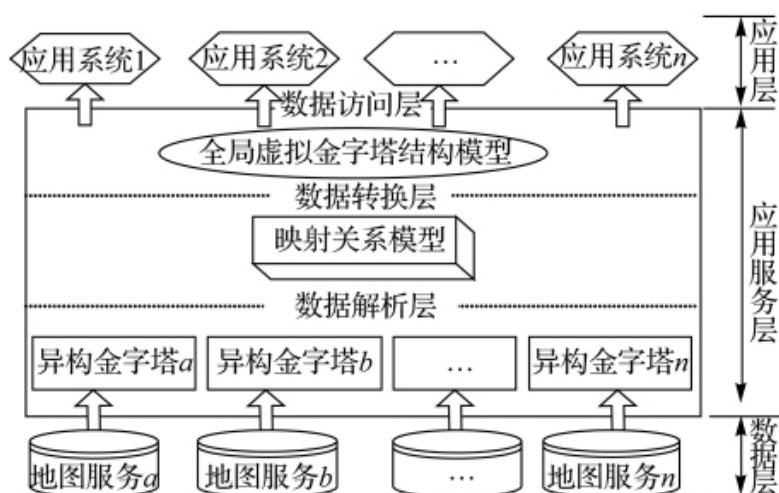


图9 模型框架图

(1) 数据层位于集成系统的底层，是各地图厂商组织和维护的各种分布式存储的异构在线地图数据，它们是需要进行数据集成的对象。

(2) 应用服务层是屏蔽在线地图服务数据组织异构性的核心环节。它包括数据访问层、数据转换层、数据解析层。数据解析层是在分析在线异构地图服务数据组织的基础上，提取异构性的数据组织，构建异构金字塔结构模型；数据访问层是在分析这些异构金字塔结构模型的基础上，构建全局虚拟金字塔结构模型，它负责提供统一数据结构的表达，方便异构地图数据的统一管理和应用；数据转换层是遵循一

定的映射规则，建立起局部异构金字塔模型和全局虚拟金字塔模型间的映射关系，最终，异构地图服务的地图数据能够通过全局虚拟金字塔模型统一访问，从而实现异构地图数据的集成。因为不同映射规则会产生不同的集成结果，所以映射规则对集成的效率和效果起决定性的作用。

(3) 应用层是 Web 的应用客户端，通过应用服务层进行在线异构地图服务的集成，为用户提供在线异构地图数据集成效果的界面。



图 10 网络在线天地图影像

2、异构在线地图数据集成模型

不同的在线地图服务，都有自己的数据组织方式，它们的地图投影方式、地图层级范围、地理编码方式等的不同，造成了在线异构地图服务间彼此独立，难以集成与共享。如谷歌地图采用墨卡托投影，地图层级范围为 0~22 级；天地图采用通用横轴墨卡托投影，地图层级范围为 2~18 级。

为了屏蔽在线异构地图服务在数据组织方式上的不一致性，实现异构地图数据的无缝集成，首先构建一个全局虚拟的金字塔结构模型，它负责统一数据结构的表达；其次建立异构金字塔结构模型与全局虚拟金字塔结构模型间的关系映射，它能准确地将异构金字塔结构模型一一对应到全局虚拟金字塔结构模型；最后通过全局虚拟金字塔结构模型实时地访问异构在线地图服务的数据，实现异构在线地图数据的集成。

I. 全局虚拟金字塔结构模型

在线异构地图服务，一般采取多分辨率金字塔结构的瓦片组织方式，即从金字塔底层到顶层，将多个采样点合并为一个，而所有层表示的范围不变，其中，每个

采样点对应一个瓦片，所以，瓦片组织方式的差异是阻碍在线地图数据集成的直接原因。为了使用统一的数据组织方式集成异构地图服务数据，数据模型的标准化采取了面向对象的方法，把模型规范成统一的结构，封装了在线地图服务的异构性。

全局虚拟金字塔结构模型来对异构金字塔结构进行统一描述，全局虚拟金字塔结构模型主要从地图投影类型、地图层级、地理编码等方面进行构建。它的瓦片组织结构如图所示，其中，I、J为地图瓦片的行列号编码。

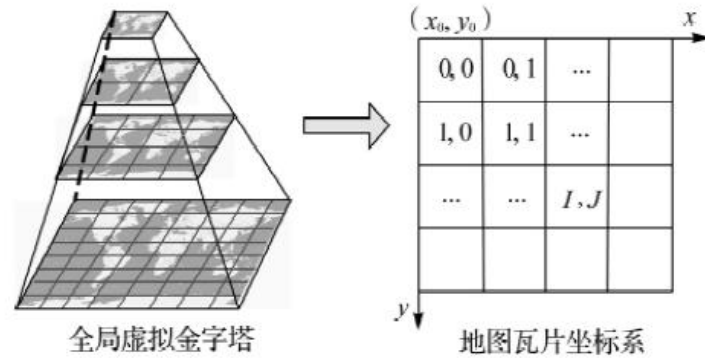


图 11 瓦片组织结构图

全局虚拟金字塔结构模型的地图投影，采取了墨卡托地图投影，它符合大多数网络地图采取的地图投影类型，在数据集成时，能方便进行地图投影转换。

全局虚拟金字塔结构模型的地图层级，采取了四叉树分层的金字塔结构。最初，把地图投影在一张固定大小为 256×256 的瓦片上，则地图最大分辨率 r 为 $360/256$ ，地图每放大 1 倍，每张瓦片分裂成 4 张相同大小的瓦片，地图分辨率缩小为原来的 $1/2$ ，则第 n 层的地图分辨率为 $res=r/2^n$ ，于是，建立起地图分辨率与地图层级的对应关系。在指定地图分辨率的情况下，可以建立起异构金字塔间的层级关系。但由于大多数网络地图都采取了这种分层组织的金字塔结构，因此在数据集成时，可以减少不必要的换算，方便了异构地图数据的集成。

全局虚拟金字塔结构模型的瓦片地理编码，采取瓦片行列号是唯一标识每块瓦片在指定图幅上相对地理位置的方式。空间参考统一选取当前图层的左上角坐标为图幅的坐标原点，瓦片左上角坐标为瓦片的参考坐标。然后，根据地图层级与地图分辨率对应关系，确定请求的地图层级 Z 与地图分辨率 res ，则单位瓦片的地理跨度为 $span=256 \cdot res$ 。参考坐标为 (x_0, y_0) ，已知瓦片左上角坐标为 (x, y) ，则瓦片行列号 (I, J) 的计算见式(1)。

$$I=(x - x_0)/span; J=(y_0 - y)/span(1)$$

虽然，异构地图数据在瓦片的地理编码上各有不同，但它们都是通过瓦片的行

列号来描述瓦片在地图图层上的相对地理位置,因此使用瓦片行列号进行地理编码,可以简化地理编码方式,提高异构地图数据的集成效率。

II. 异构金字塔结构

虚拟金字塔结构模型的映射从异构金字塔结构模型到全局虚拟金字塔结构模型的映射,是从数据结构的层面对异构空间数据进行的转换。它包括地图投影转换、地图层级的匹配、地理编码的统一等方面。

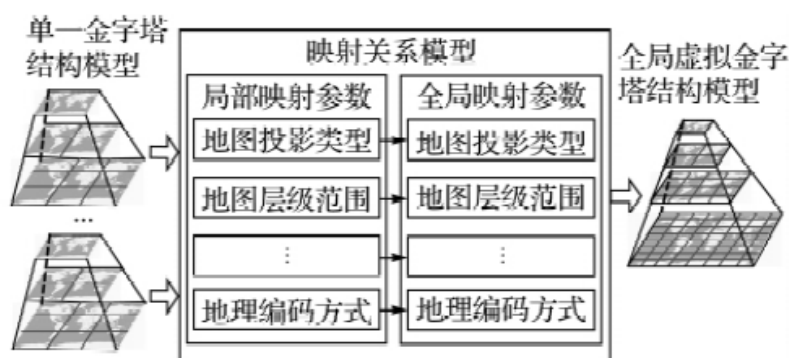


图 12 异构金字塔结构图

1)在地图投影转换方面,由于地图投影方式的不同,造成了异构空间数据集集成时地理坐标的偏移和地图影像的变形,对此可以采取墨卡托投影转换公式(2)建立实际地理坐标到地图经纬度坐标的投影转换。

$$x=R \cdot \lambda; y=R \ln \tan(\pi/4+\varphi/2) \quad (2)$$

式(2)中, R 为地球半径, (x, y) 为实际地理坐标, λ 和 φ 分别为地图上的经度和纬度。地图影像的变形处理,需要依据地图请求范围与地图比例尺的要求,根据选取的参考点坐标,不断调整纵向、横向上的变形参数,使其最终能通过统一的空间参考进行表达。

2)在地图层级匹配方面,地图层级范围的不同,造成了异构地图数据集集成时,地图分辨率无法匹配。可以运用地图层级组织的四叉树结构性质,即地图层级每增加一级,地图分辨率缩小为原来的 $1/2$,以建立地图分辨率与地图层级间的对应关系;

然后,通过当前地图分辨率指定地图层级的方式,建立不同地图服务在地图层级上的对应关系。

3)在地理编码统一方面,瓦片的地理编码描述了瓦片在服务器端的存储方式,不同的地理编码方式,造成了异构地图数据集集成时,同一图层的瓦片在空间组织上无法匹配。可以根据统一的地理编码方式,建立局部地理编码到统一地理编码的映

射关系：首先，建立局部参考坐标到统一参考坐标的映射，通常采取统一参考坐标为基准，对局部参考坐标加上一个横向或纵向的偏移量;其次，建立局部地图分辨率到统一地图分辨率的映射，即根据请求的地图分辨率，将地图分辨率对应到统一的经纬度坐标分辨率，并确定地图的层级；最后，计算瓦片行列号，通过请求的图幅范围，获得请求地图瓦片行列号范围。

系统测试在线的天地图，其结构分为三个部分，数据层主要由在线地图服务的异构地图数据组成；中间层是基于本文方法开发的异构地图数据集成模型和遵循本文提出的映射规则构建的映射关系中间件，应用层是基于不同业务需求的应用客户端。体系结构采用 Web 的架构，提供在线异构地图的 Web 集成服务。

4.1.3 多源数据的集成验证

三维地质网格地球除了支持本地数据的可视化之外，利用 SIG 技术实时地调用网络上的图像数据，合理地加载对象数据，将 Google Earth、Virtual Earth、天地图、多比例尺地质图、地下三维地质模型数据等网络信息数据源在三维地球上进行可视化集成，实现三维自适应可视化、真三维立体显示、模拟仿真和多分辨率表达。

本系统中不仅实现了本地数据，如卫星影像数据、矢量数据、瓦片数据在三维地球上的可视化，并且利用 SIG 技术实时的调用网络上的图像数据，如下面展示的天地图在三维地球上的可视化集成效果，实现了多源数据的可视化集成。下面以天地图影响的实现为例。



图 13 来自网络的天地图影像

其详细配置步骤为：在示例站点的安装路径如 D:\Program Files\MAPGIS

VirtualEarth\Web_Sample\VirtualEarthDemo\WebRoot\applet\config 中的 EarthConfig.xml 文件中，于用户自定义标签即<CustomLayers>中添加天地图的配置内容：

```
<LayerInfo>
  <DataType>10</DataType>
  <AliasName>天地图影像</AliasName>
  <LayerName>tdtyingxiang</LayerName>
  <Version>1</Version>
  <VisibleStartLevel>0</VisibleStartLevel>
  <VisibleEndLevel>16</VisibleEndLevel>
  <Visible>false</Visible>
  <DrawType>png</DrawType>
  <Opacity>1</Opacity>
  <Sector>-180,-90,180,90</Sector>
  <RequestPage>GetTianditu.jsp</RequestPage>
  <TileSize>256</TileSize>
  <Delta>45</Delta>
</LayerInfo>
```

4.2 全国地名数据与三维地球的可视化与查询集成

数字地球是以计算机技术、多媒体技术和大规模存储技术为基础，以宽带网络为纽带运用海量地球信息对地球进行多分辨率、多尺度、多时空和多种类的三维描述，并利用它作为工具来支持和改善人类活动和生活质量。可见建立数字地球的目 的是为人类生活服务，与我们日常生活相关的感兴趣的很多位置信息都可以在数字地球上展示出来，这就是地理信息系统中的术语 POI 所指的内容。

本系统集成网络上的地名数据，实时的获取各种相关的旅游、娱乐、服务、教育、政府机构、餐饮等不同类别的数据，实现 POI 数据的可视化，将数据以三维全景的方式展示在地球上，并且实现了全球地名数据的分析功能，为以后的地质调查规划工作提供科学、准确的依据。

4.2.1 POI 数据的网络下载与可视化

1、POI 数据在网络上的传输

网络带宽与空间数据量大之间的矛盾是 Internet 地图服务普及的主要障碍之一，提高空间数据的传输效率是所有 Internet 地图服务要解决的主要问题，网络环境下常见的数据压缩方法有下面四种。

第一种是用 short 型代替 float 或 double 型数据，在不减少数据精度的情况下，给坐标数据扩大一定的倍数，使其可以用 short 型数据表达。第二是用映射屏幕范围方法，将空间数据转换到屏幕空间坐标系中第三种采用虚拟屏幕进行“滤点压缩”，将数据首先映射到虚拟屏幕中的一个空间点中，然后将空间目标点相差特别近的点进行合并，实现数据的压缩。第四是用长边加点解决偏移量压，基于一条弧段上所有相邻坐标点变化不大的情况，可以采用偏移量压缩，但是如果某些结点之间坐标相差太大，可能会超过给定的字节长，造成数据错误，不过可以采用增加结点的方法来弥补这个缺陷。

这些算法都是基于素具本省精度的降低或空间坐标信息的重新组织，具有一定的实用价值。这次基础上，对形成的文件可以采用二进制，将其压缩为 zip 格式以减少网络传递信息的总字节数。

在选择地图数据的传输方式时，可以采用 socket 或是 servlet，但是 servlet 方式需要先转化为 http 协议，然后再传输、解析，效率比较低。因此我们采用 socket，基于 socket 的数据表现形式为字节流信息，在数据的传送和接受双方按照约定的方式进行数据的格式化和解释。虽然这种方式的编程方式比较麻烦，但扩展性更强。其基本的工作流程如图，服务器首先创建服务器端 Socket 并启动，监听客户端 Socket 的连接请求，一旦请求成功，随即按照双方约定的数据格式，在虚拟通道上传输数据，待数据传输结束，客户端断开连接。

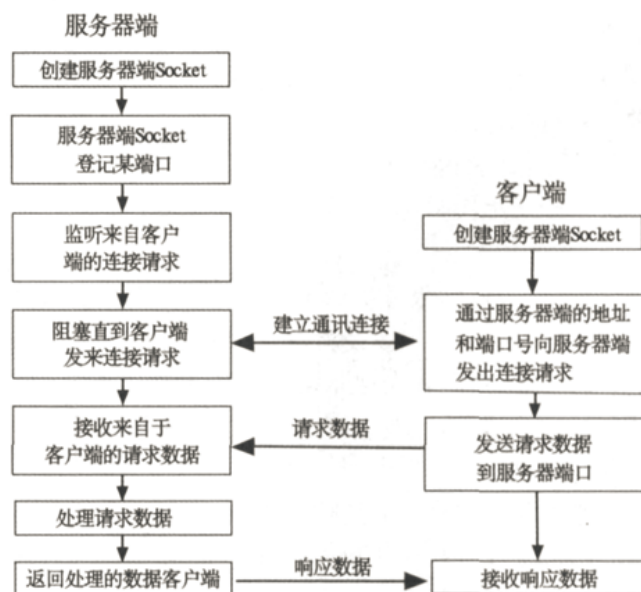


图 4 Socket 工作流程

Fig.4 Socket work flow

图 14 Socket 工作流程图

2、POI 数据的三维可视化

我们使用 google 提供的地图服务，实时的下载网络上的地名数据，将数据下载到本地并将对象数据在三维地球上进行可视化。从网络上下载地名数据，将数据文件存放到指定的数据目录下，利用 IGServr 提供的地图服务，将数据文件中地名数据的描述信息发送到 IGServr 中 GetLableXml.jsp，将数据解析为系统能够识别的地名对象，并将数据信息展示在三维地球上。

地名数据的可视化是以图层的形式展示，在图层控制中确定地名数据的可视化与否。在系统的图层控制树中有一个地名数据根结点，在此结点下可以合理的添加和删除不同地名数据图层。在站点安装目录下的路径 MAPGIS VirtualEarth\Web_Sample\VirtualEarthDemo\WebRoot\applet\config 中，修改 EarthConfig.xml，在<LableLayers>标签中添加子结点<LayerInfo>信息，如：

```
<LayerInfo>
  <DataType>6</DataType>
  <LayerName>province_36</LayerName>
  <AliasName>省级行政区</AliasName>
  <MinVisibleDistance>900000</MinVisibleDistance>
  <MaxVisibleDistance>20000000</MaxVisibleDistance>
  <Visible>>false</Visible>
  <Grid>36</Grid>
  <Sector>-180,-90,180,90</Sector>
  <RequestPage>GetLableXml.jsp</RequestPage>
</LayerInfo>
```

我们需要根据地名数据所在的文件夹名称设定 LayerName 的值，AliasName 字段是设置地名数据在图层展示中的图层的名称。在控制地名数据的显示范围时，可以设定其最大和最小的可视高度，例如 900KM，在 Visible 中设定图层初始化时是否可见。

例如添加省级行政区结点后，在地名数据结点下将增加省级行政区子结点，点击结点的单选框设置地名数据是否展示在三维地球上。



图 15 图层控制

根据地名数据的文件存储的具体信息，从中提取经度、纬度字段从而进行相应地名在地球上的定位，再根据可视化设置的显示内容、字体、大小、颜色，实现地名数据在三维地球的展示。除了以文件存储的地名数据能够实现三维展示之外，为了能够满足将来地质调查工作的高效执行，我们还提供了 P O I 数据的查询功能，并实现相应地名数据的三维展示。

4.2.2 POI 数据的在线查询

提供输入搜索关键字及搜索条件，获得住宿、餐饮、娱乐等 POI 信息的一种服务。POI 的在线查询不仅能够提供全球 POI 的查询服务，并且其查询结果随自主运营更新及地图厂商数据版本升级不断更新，实时的获取最新的 POI 信息。

在三维可视化效率逐渐提高的基础上，充分利用网络资源，实现全国乃至全球的地名查询，有效的降低了本地系统的分析代价，提高系统的整体性能。

我们利用 Google 地图 api 提供的客户端地址解析器，用于动态地对用户输入的地址进行解析，而地址解析就是将地址转换为地理坐标的过程。本系统根据输入的地名关键字，选择查询目标的分类，包括餐饮、娱乐、教育机构、政府机构、购物、住宿，只需要访问 Google 地图 api 的地址解析服务（我们组织的地址解析服务的地址为

http://ditu.google.cn/maps/geo?output=json&oe=utf-8&q=wuhan&key=ABQIAAAAzr2EBOXUKnm_jVnk0OJI7xSosDVG8KKPE1-m51RBrvYughuyMxQ-i1QfUnH94QxWla6N4U6MouMmBA&mapclient=jsapi&hl=zh-CN&callback=_xdc_.1g4gm5mh3，其中 q 的值是地名经过转换编码后的地址名称）。通过此方

式请求地址信息，返回一个 JSON 对象：

```
_xdc_.1g4gm5mh3 && _xdc_.1g4gm5mh3( {
  "name": "武汉大学",
  "Status": {
    "code": 200,
    "request": "geocode"
  },
  "Placemark": [ {
    "id": "p1",
    "address": "中国湖北省武汉市武昌区校园南路武汉大学 邮政编码：
430072",
    "AddressDetails": {
      "Accuracy" : 9,
      "Country" : {
        "AdministrativeArea" : {
          "AdministrativeAreaName" : "湖北省",
          "Locality" : {
            "DependentLocality" : {
              "AddressLine" : [ "武汉大学" ],
              "DependentLocalityName" : "武昌区",
              "Thoroughfare" : {
                "ThoroughfareName" : "校园南路"
              }
            },
            "LocalityName" : "武汉市"
          }
        },
        "CountryName" : "中国",
        "CountryNameCode" : "CN"
      }
    },
    "ExtendedData": {
```

```
"LatLonBox": {
  "north": 30.5500676,
  "south": 30.5308475,
  "east": 114.3778284,
  "west": 114.3458136
},
"Point": {
  "coordinates": [ 114.3618210, 30.5404580, 0 ]
},
}
```

Json 对象包含的信息主要有以下几个部分:

(1) Status:

request - 请求类型。在此情况下, 它始终为 **geocode**。

code - 响应代码(与 **HTTP** 状态代码类似), 用于表明地址解析请求是否成功。

(2) Placemark - 如果地址解析器查找到多个匹配项, 则会返回多个地标, 它们以 **id** 进行唯一标识。

address - 格式恰当且大小写正确的地址。

AddressDetails -- 使用 **xAL** 格式的地址, 或称为可扩展地址语言 (**eXtensible Address Language**) (一种设置地址格式的国际标准)。

Accuracy - 表示指定地址的地址解析所能达到的精确度的属性。

Point - 3D 空间中的一个点。

coordinates - 该地址的经度、纬度和海拔。在此情况下, 海拔始终设为 0。

在成功获取网络上的地址信息资源后, 需要将其解析为三维地球上的一个或多个地球上的点对象。根据解析得到的点对象, 利用三维地球的图形操作接口, 将这些得到的信息点定位到三维地球上进行展示。为了提高查询结果的可观性, 我们需要将信息点的内容以直观、全面的方式显示在三维地球上的对应位置。

为了提高系统的性能, 我们构建客户端缓存, 用来存储地址解析器的响应, 当对某个地址重复进行地址解析时, 地址解析器能够更快的做出响应。

为了满足系统在设有代理服务器的用户也能够在线使用, 提供一个系统的配置文档, 使用代理上网的用户首先需要在 src 下的 webconfig.properties 中设置 IsProxy, true 表示使用代理, ProxyIP、ProxyPort 分别表示代理服务器的 IP 与端口号, IsProxy 为 false 时表示直接连接外网, 其他的设置无效。

我们将 google 地图提供的地名查询服务有效的集成到三维地球中, 实现全国地名数据的查询和分析, 其实现效果如下所示。例如, 搜索关键字武汉大学, 得到两个有关武汉大学的地名信息点, 于是分别将这两个查询结果展示在地球上, 点击各地名上的标注, 将显示详细的地名信息, 包括经纬度坐标、地名全称、邮政编码等。



图16 POI数据的在线查询

4.3 全国道路分析的三维地球空间分析集成

在三维地球上集成网络上的全国道路分析功能, 对空间对象进行三维空间分析和操作, 用户则可在客户端实现其管理复杂空间分析能力, 并在三维地球上进行实时漫游, 实现实时道路信息与资源定位信息建立协同调度与信息共享一体化成为可能, 同时对提高应急响应能力、提升应急救援决策支持系统的科学性具有重要的意义。

Google 地图中经常用到道路指向功能很好的阐述了道路分析的过程, 它是使用 web 地图, 终端用户输入目的地与出发地, 然后提交到网络地图服务器 (WMS 服务器), 服务器通过 geocoding (地址分析) 过程将目的地与出发地的经纬度位置计算出来, 再通过查询路径或者计算最优路径来形成最终的查询结果。

本系统充分利用网络上的信息资源, 将道路分析的出发点与目标点的信息发送

到网络资源的地址，不需要本地的任何分析过程，直接获得到网络上分析得到的结果，将特定格式的地理信息数据进行适当的处理即可满足系统的需要。

4.3.1 请求网络上道路分析服务

Google Directions API 是一种是用 HTTP 请求计算多个位置间路线的服务。路线可以采用文本字符串或经纬度坐标的形式指定起点、目的地和路标。Google Directions API 可以用一系列路标传回多段路线，计算路线是一种耗时、耗资源的任务，因此我们使用了此路线服务提前计算已知路线，并将结果存储在临时缓存中。

(1) 路线请求

Directions API 的请求采用以下形式：

<http://maps.google.com/maps/api/directions/output?parameters>

其中，output 可能是以下任何一个值：

json，表示以 javascript 对象表示的形式输出

xml，表示以 XML 的形式输出

(2) 请求参数

有些参数是必填参数，有些是可选参数。根据网址的标准，所有参数均使用字符&分割。

参数名	解释	必选/可选
Origin	计算路线的起始地址名称或经纬度值	必选
Destination	计算路线的目标点的地址名称或经纬度值	必选
Mode	默认为 driving ，指定计算路线的出行方式	可选
Waypoints	指定一组路标，通过路标可改变路线走向	可选
Alternatives	设置为 true 时，可以指定路线服务在响应中提供多条备选路线	可选
Avoid	只是计算路线应该避开指定的地图项，参数值可为 tolls (指示路线避开收费路段)，或 highwas (指示计算路线避开高速公路)	可选
Units	指定显示结果时使用的单位制	可选
Region	区域代码，指定为 ctld 双字符值	可选
Language	传回结果时使用的语言	可选
Sensor	指示路线请求是否来自装有位置传感器的设备，值为 false 或 true	必选

注意，可以使用多个参数传递地址或经纬度坐标。如果将地址作为字符串传递，路线服务将会对该字符串进行地址解析，然后将其转换为经纬度坐标，计算路线。若传递的是坐标值，则保证经纬度之间无空格。

(3) 返回路线

我们使用请求输出格式为 XML，请求 url 示例如

<http://maps.google.com/maps/api/directions/xml?origin=30.4739174,114.4066726&destination=30.554201,114.3876237&alternatives=true&mode=driving&sensor=false&language=zh-CN>，此请求传回的 XML 如下所示：

```
<?xml version="1.0" encoding="UTF-8" ?>
- <DirectionsResponse>
  <status>OK</status>
- <route>
  <summary>珞喻路</summary>
- <leg>
- <step>
  <travel_mode>DRIVING</travel_mode>
- <start_location>
  <lat>30.4738200</lat>
  <lng>114.4066800</lng>
</start_location>
- <end_location>
  <lat>30.4737900</lat>
  <lng>114.4058600</lng>
</end_location>
- <polyline>
  <points>k|~xDw`hyTDbD</points>
</polyline>
- <duration>
  <value>9</value>
  <text>1 分钟</text>
</duration>
  <html_instructions>从<b>软件园路</b>向<b>西</b>方向，前往<b>软件园东
路</b></html_instructions>
- <distance>
  <value>79</value>
  <text>79 米</text>
```

```

    </distance>
    </step>
+ <step>
+ <step>
+ <step>
+ <step>
+ <step>
+ <step>
+ <step>
+ <duration>
+ <distance>
+ <start_location>
+ <end_location>
    <start_address>中国湖北省武汉市洪山区软件园路 4 号</start_address>
    <end_address>中国湖北省武汉市武昌区黄鹄路 158 号 邮政编码:
430077</end_address>
    </leg>
    <copyrights>地图数据 ©2012 GS(2011)6020 Google, Mapabc</copyrights>
+ <overview_polyline>
+ <bounds>
- <southwest>
    </route>
+ <route>
- <end_location>
+ <route>
</DirectionsResponse>

```

该 XML 响应包含一个<DirectionsResponse>和两个顶级元素<status>（包含请求中的元数据）、<route>（零个或多个<route>元素，每个元素均包含起点和终点之间的一组路线信息）。

当 Google Directions API 传回结果时，它会将这些结果放在 routes 数组中。即使该服务不传回任何结果（例如，如果起点和/或目的地不存在），它仍会传回一个空的 routes 数组。

routes 数组的每个元素都包含一个来自指定起点和目的地的结果。此路线可能

包含一个或多个 **legs**，具体取决于是否指定了任何路标。此外，除路线信息外，该路线还包含必须向用户显示的版权和警告信息。

routes 字段中的每个路线都可能包含以下字段：

summary 包含路线的简短文本描述，适于命名路线以及消除路线与备选路线之间的歧义。

legs[] 包含一个数组，该数组包含关于路线的一段路程（介于指定路线中两个位置之间）的信息。系统将会为指定的每个路标或目标显示一个单独的路程。（一个没有路标的路线在 **legs** 数组中仅包含一段路程。）每个路程均由一系列 **steps** 组成。

waypoint_order 包含一个指示计算路线中所有路标顺序的数组。如果通过某请求的 **waypoints** 参数向该请求传递了 **optimize:true**，则这些路标可能会重新排序。

overview_polyline 包含一个对象，该对象包含一组表示生成路线的近似（平滑）路径的已编码 **points** 和 **levels**。

bounds 包含该路线的视口边框。

copyrights 包含将要为此路线显示的版权文本。

warnings[] 包含将在显示这些路线时显示的一系列警告。

4.3.2 道路分析结果的可视化

在得到路径分析结果后，为了更好的将查询结果展示在用户界面上，我们将得到的 **xml** 信息序列化为 **Route** 对象，得到路径分析结果的各个路段信息，将这些信息组织为制定的格式，以字符串的形式返回到客户端。然后在客户端对路径信息进行分割组合，得到能够在网页中正确显示的内容，将其以直观、简洁的形式在分析结果列表中详细列出。最后，利用系统的处理图形的接口，用

removeAllSurfaceShapeOf 接口，在绘制当前查询结果之前，清除其他操作在地球上绘制的所有图形。根据路径上各个关键点的坐标值，依次定位关键点并连接各路段，将查询得到的路径绘制到地球上，进一步提高了查询结果的可视化效果。

本功能的具体实现举例如下所示。

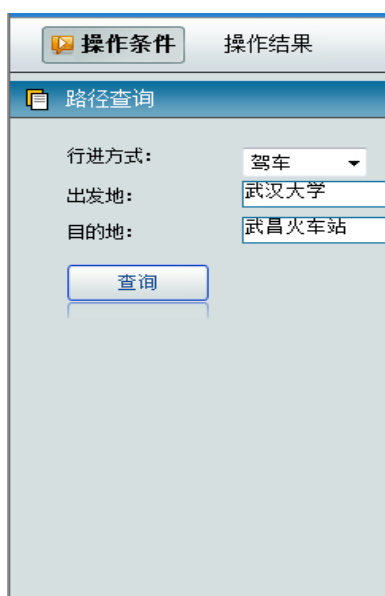


图17 路径查询

本系统集成网络上的全国道路分析功能，根据起始点、终点的地名信息，并提供多种出行方式的分析功能，包括驾车、步行、公交或是不考虑出行方式。获取两点之间所有可能的路径，将道路信息详细的显示在列表中，包括各条路径的名称、全程总长、耗时以及每个道路拐点的信息，指明出发地与目标地在地球上的位置，最终将所有路径显示在三维地球上。



图18 路径查询操作结果

4.4提高三维可视化显示效率方法研究

三维地球的可视化应用通常需要处理的数据量巨大，所以这些数据通常是分布式存放，在使用时需要在各个网格节点间进行传输，因而一个好的数据管理方案对提高基于网格的三维可视化处理速度非常重要。

为了提高矢量地图的传输和显示效率，我们借鉴 google 的地图瓦片的思想，利用 GML 文件表示地理数据，矢量数据的切割和合并的方法，以此快速给用户需要的数据，同时提高用户的客户端体验，在客户端用到基于瓦片的地图结构，起到瓦片缓存处理的功能。

4.4.1 矢量图的裁剪

矢量图形是通过矢量数据来表达的，因此矢量图形的裁剪本质是对矢量数据的分割。描述一个地理实体的矢量数据不仅包括它的坐标信息，还带有实体的属性信息，因此在裁剪过程中要考虑属性信息的继承关系。矢量图形的信息是按照分图层、分类型（点、线、面）的形式存储的，一个图层上只存在一种类型的数据，对于地理实体的裁剪就具体到单个图层上对点、线、面坐标信息的分割处理。

基于瓦片的图幅裁剪，可以定义裁剪窗口为正方形或者是长方形，首先从矢量文件中读出这个图层的外接矩形坐标，根据外接矩形坐标和瓦片的规定大小，求出应当分割成的瓦片数和每个瓦片的边界坐标，并对每一个瓦片赋予唯一的 ID，为每个瓦片建立一个矢量文件，写入对应的 ID 和边界坐标。矢量图形裁剪为指定的瓦片数后生成多个矢量图形文件分别存储分割后的矢量数据，为实现数据请求时的分片传输矢量数据提供数据基础。

4.4.2 矢量图的合并

当三维地球上的用户操作请求到达服务器时，地图服务器将地图数据切割成不同的瓦片，然后将请求范围内及其附近的瓦片传回到客户端。在客户端，地图瓦片被合并呈现给用户可视的三维显示。为了降低数据的传输量和客户的等待时间，采用瓦片机制及与之对应的瓦片合并和缓存机制，进一步提高数据的三维显示效率。

4.4.3 客户端的地图结构

瓦片在服务器端裁剪，而在客户端进行合并，这就需要一个中介完成两者的联系。我们使用双层结构的地图文档，完成地图在客户端的合并。第一层为准备层，例如 5x5 的瓦片网格，它负责从客户端的浏览器或服务器抽取瓦片，提供给显示层使用，这一层对用户来说不可视，起到缓冲的作用。第二层是显示层，如 3x3

瓦片合并而成。它从准备层抽取在显示范围内的瓦片，在这一层进行合并，形成可视的地图在地球上展示。当用户执行漫游操作时，准备层中拖动后将在可视范围内的瓦片提取到显示层，在显示层进行合并后，显示给用户；同时，准备层会为用户的下次操作而向浏览器缓存或服务器发出请求，预取可能用到的瓦片存入准备层中，多余的瓦片被丢掉。

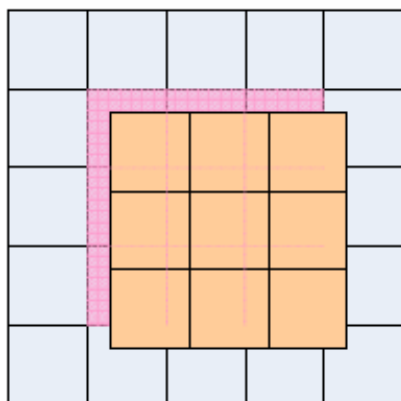


图 19 地图文档结构

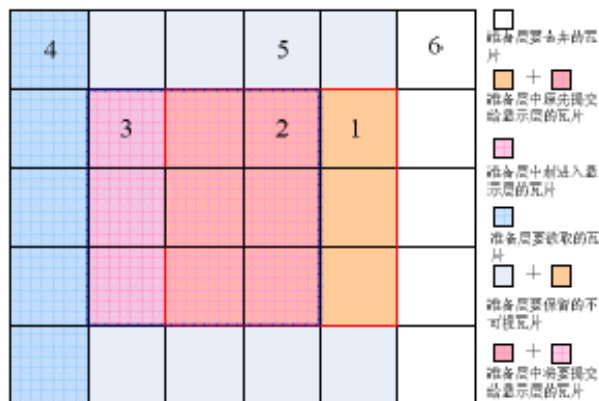


图 20 漫游中地图文档的变化

4.4.4 客户端和服务端端的交互

Ajax 是多种技术的综合，它使用 XHTML 和 CSS 的标准化呈现，使用 XML 和 XSTL 进行数据交换和处理，使用 javascript 绑定和处理数据，发送 XMLHttpRequest 请求进行数据的异步读取、使用 DOM 技术来实现页面的动态显示和交互，使得客户端只异步请求可视范围内的地图数据。由于用户可以在页面内和服务器进行交互，避免了页面的整体刷新，从而减少数据量的传输，极大的改善用户的交互体验，使得用户有操作桌面应用的感觉。

矢量数据在地球上的显示如下。



图 21 矢量数据的显示

矢量数据的动态瓦片裁剪效果如下图。



图 22 矢量数据的动态瓦片裁剪效果图

矢量数据以瓦片地图的展示方式实现了动态刷新浏览页面：



图 23 动态刷新浏览页面

4.5 地上、地表、地下三维一体化的三维发布系统

4.5.1 三维建模方式

三维一体化的建模与可视化主要研究包括：三维地形建模及可视化，三维建筑物建模及可视化，三维地下管网、地质构造的建模与可视化。这里可以使用两种不同的建模方法，根据实际应用中的需要，选择最为合适的建模方法。

1、 三维地表粗精度模型

三维地表粗精度模型的建立是基于面结构的建模方式。基于面结构建模的数据结构有：规则格网(Grid)，不规则三角网(TIN)，边界表示法(Boundary

Representation, B-Rep)和参数函数法。我们用目前应用较普遍的边界表示法(B-Rep)

的思想实现建筑物的粗精度模型的建立。边界表示法是以物体的边界面为基础定义和描述三维物体的方法,强调物体外表的细节,详细记录构成物体的所有几何元素的几何信息及其相互的连接关系及拓扑信息。

在二维可视化系统中,建筑物一般只用其投影到地面的轮廓线来表达,并且将该轮廓线作为地图的一个面对象存储在地图数据中。三维地表粗精度模型的建模原理是:在二维图形的基础上,利用现有的地图数据,从中提取这些轮廓线,用它们作为三维建筑物的底面,然后结合建筑物的高度数据构造一个三维几何实体。实际构建时,一般通过对底面多边形先用三角剖分方法进行剖分。然后将其拉伸到一定的高度,就形成三维实体,实体的数据按照“点-线-面-体”的逻辑结构组织。这样建立的三维实体模型,结构比较简单,通常一个几何实体仅用该建筑物的平均高度或者中心点的高度来表示。

(1) 粗模信息文件

构建三维地表粗精度模型所需要的信息包括建筑物的轮廓信息、高度、中心点的位置等几何信息和空间信息以及建筑物的名称、类型等属性信息,我们通过 xml 文件来记录这些信息,实现结构化的组织,这个文件我们称其为粗模信息文件。在数字地球漫游过程中,我们通过读取该文件,并解析其结构,从中提取建筑物的相关数据,完成模型的构建。在 xml 粗模信息文件中我们需要对底面多边形构造出三角面数组,每个三角面包含三个顶点的索引,通过索引可以检索到其对应的顶点。

三角面数组的存储结构是我们设计粗模信息文件时主要考虑的问题。我们知道,对于每一个三角形,都需要用 3 个顶点来表示。例如在一个立方体模型中,一共有 $6 \times 2 \times 3 = 36$ 个顶点。而在这 36 个顶点中,有相当数量的顶点是重合的。如果把这些重合的顶点都一一表示出来,就太浪费存储空间了。因此我们采用顶点索引的想法解决空间占用问题。顶点索引的思想是:建立两个数组,一个数组用于存储模型中所有的顶点坐标值,另一个数组则存储每一个表面所对应的三个顶点在第一个数组中的索引。

粗模信息文件中一个几何实体信息的结构如下图所示,由 r 节点包含的内容表示一个建筑物几何实体所具有的几何信息、空间信息和属性信息。其中 h 表示建筑物的高度, name 表示建筑物的名称, type 表示建筑物的类型, center 表示该建筑物的中心位置坐标;对底面多边形的表示,我们用 c 存储构成建筑物的轮廓的顶点坐标封闭串的值,用 t 存储建筑物底面的三角剖分的结果(即对底面多边形构造的三

角面数组)。

```
<r>
    <h>8</h>
    <name></name>
    <type></type>
    <center>114. 220662, 30. 538521</center>
    <c>114. 220662, 30. 538521#114. 220168, 30. 538260#114. 220071, 30. 538398#114. 220565, 30. 538659#114. 220662, 30. 538521</c>
    <t>2, 1, 0#2, 0, 3</t>
</r>
```

粗模信息文件的结构片段

(2) 粗模信息文件的管理

建筑物模型的构建及渲染是实时进行的，为了实现可视化的快速性与连续性，不可能把全球所有的模型数据一次性绘制出来，只需对视区内的建筑物进行建模及可视化。如何组织与管理粗模信息文件，使得数字地球客户端应用程序或浏览器能够快速获得当前视区范围内的粗模信息，通过对其进行解析，快速建立起建筑物的模型？按照三维地形可视化中对地形数据、遥感影像数据的组织与管理策略，我们对建筑物模型也采取类似的组织方式，将地表按照一定的经纬度跨度（综合考虑模型的渲染速度与文件下载速度后，选择 0.015° ）进行划分，得到一个瓦片矩阵。对每一个瓦片建立一个粗模信息文件，这样在漫游时就可以通过实时计算当前视点范围对应的瓦片，然后通过瓦片的行号和列号查找相应的粗模信息文件。

在数据服务器端，我们建立一个根文件夹。在根文件夹中为每一行瓦片建立一个文件夹，第 0 行建立文件夹 0，第 1 行建立文件夹 1，……，第 n 行建立文件夹 n。然后在每一行的文件夹中，对应于每一个瓦片建立一个粗模信息文件，每一个粗模信息文件记录的是相应的瓦片上的建筑物的信息。当一个建筑物跨越多个瓦片时，通过建筑物的底面中心点的位置确定每一个建筑物属于哪个瓦片。粗模信息文件的命名采用“行号_列号.xml”的格式，这样就可以通过计算瓦片的行号和列号，然后索引到相应位置的建筑物模型信息文件。

(3) 三维地表粗精度模型可视化

粗模信息文件存储在数字地球的服务器端，当我们需要构建模型时，首先计算当前视点范围内的瓦片的行号和列号，然后向 Web 服务器层发出请求下载相应位置的粗模信息文件。Web 服务器端接到请求后，通过数据缓存机制、数据的高效传输与分析机制等获取到请求的 xml 粗模信息文件，将其下载到本地内存中，同时将其写入本地数据缓冲池。

在构建模型时，通过解析粗模信息文件，同时使用 OpenGL 的接口函数对地表建筑物进行模型构建与渲染。根据三维地表粗精度模型构建的原理，读取粗模信息

文件中每一个建筑物的信息，首先按照 **c** 中的顶点坐标串将三维实体在地面上的投影平面的轮廓构造好，作为三维实体的底面，然后按照 **center** 中记录的该实体的中心位置将其嵌到数字地球的表面上，并对这个底面进行三角剖分，再将剖分过的底面拉伸到 **h** 中所表示的高度，这个实体的轮廓就完成了。然后给这个实体添加 **name** 属性和 **type** 属性，当然还可以添加其他属性。

2、 三维地表高精度模型

对于空间几何形状比较复杂的建筑物，或者要求模型比较精致美观的情况下，就需要应用专业的建模工具软件定制该地物模型，常见的建模工具有 3DMax, MAYA, AUTOCAD 等。模型建好后，将其保存为系统可解释的数据格式（如 3DS, OBJ），最后根据坐标位置、方向、尺寸等约束条件将建好的模型嵌入到三维场景中。这就是我们所研究的三维地表高精度模型的建模及可视化原理。我们的研究重点是如何将在建模工具中建好的 3D 模型在三维场景中渲染出来。重点研究了 3ds 文件和 OBJ 文件格式的几何实体模型在数字地球上的可视化。

（1） 模型文件结构分析

1) 3DS 文件

3DS 文件是 3DMax 的一种常用模型文件格式。3DS 文件是用“块(chunk)”存储的，**chunk** 是 3DS 文件的基本构成单位。每一个 **chunk** 包括一个头和一个主体。**chunk** 是相互嵌套的。**chunk** 的头又由两部分组成：ID 长一个字，**chunk** 的长度（以字节为单位，包括头）长一个双字。ID 表示 **chunk** 的含义。一个 3DS 文件事实上有上千个 **chunk**，它们相互嵌套构成了一个复杂但灵活的文件系统。块结构给我们带来了很大的方便，我们在读入时可以根据自己的需要选择性地读入自己需要的块，而不需要知道所有块的结构。通过分析 3DS 文件的结构，我们了解到文件通过相互嵌套的定义给出了场景物体的层次表达，它把面的数据和顶点的数据分开存储，几何信息与材质的信息分开存储，并建立索引使文件所需保存的信息量最小。同时将动画、关键帧的信息也写入文件中，它还把配置，各种视口，背景，颜色，相机信息等场景信息也包含在内。

2) OBJ 文件

OBJ 文件是 Alias|Wavefront 公司为它的一套基于工作站的 3D 建模和动画软件“Advanced Visualizer”开发的一种标准 3D 模型文件格式，很适合用于 3D 软件模型之间的互导，目前几乎所有知名的 3D 软件都支持 OBJ 文件的读写。OBJ 文件以

纯文本的形式存储了模型的顶点、法线和纹理坐标和材质使用信息，整个文件由一行行文本组成。OBJ 文件中每一行都很相似，其格式为“前缀 参数 1 参数 2 参数 3”。其中，前缀标识了这一行所存储的信息类型，参数则是具体的数据。

在一个 OBJ 文件中，首先是以符号“#”为开头注释行，然后由一些以 v、vt 或 vn 前缀开头的行指定了所有的顶点、纹理坐标、法线的坐标。然后再由一些以 f 开头的行指定每一个三角形所对应的顶点、纹理坐标和法线的索引。在顶点、纹理坐标和法线的索引之间，使用符号“/”隔开的。空格和空行可以随意加到文件中以增加文件的可读性。一个 f 行可以以下面几种格式出现：

① f 1 2 3

表示以第 1、2、3 号顶点组成一个三角形。

② f 1/3 2/5 3/4

表示以第 1、2、3 号顶点组成一个三角形，其中第一个顶点的纹理坐标的索引值为 3，第二个顶点的纹理坐标的索引值为 5，第三个顶点的纹理坐标的索引值为 4。

③ f 1/3/4 2/5/6 3/4/2

表示以第 1、2、3 号顶点组成一个三角形，其中第一个顶点的纹理坐标的索引值为 3，其法线的索引值是 4；第二个顶点的纹理坐标的索引值为 5，其法线的索引值是 6；第三个顶点的纹理坐标的索引值为 6，其法线的索引值是 2。

④ f 1//4 2//6 3//2

表示以第 1、2、3 号顶点组成一个三角形，且忽略纹理坐标。其中第一个顶点的法线的索引值是 4；第二个顶点的法线的索引值是 6；第三个顶点的法线的索引值是 2。

(2) 三维几何实体模型的管理

三维几何实体模型的组织与管理方式与粗模信息文件的组织原理相同，也是将地表按照一定的经纬度跨度（综合考虑模型的渲染速度与文件下载速度后，选择 0.015° ）进行划分，得到一个瓦片矩阵。在数据服务器端建立一个根文件夹，在根文件夹中对应于每一行建立一个文件夹，第 0 行建立文件夹 0，第 1 行建立文件夹 1，.....，第 n 行建立文件夹 n。然后在每一行的文件夹中存储这一行中每一个瓦片对应的所有三维几何实体模型文件（.3ds 文件或者.obj 文件）以及对这些模型文件的一个配置文件（.xml 文件）。当一个 3D 模型跨越多个瓦片时，通过模型的中心点位置确定该模型属于哪个瓦片。

模型配置文件的命名采用“行号_列号.xml”的格式，这样就可以通过计算瓦片的行号和列号，然后索引到相应位置的模型文件，进而通过解析模型配置文件中对三维实体模型的配置信息来查找并下载相应的模型文件。模型配置文件中主要记录的是模型的名称、模型的文件类型（3DS 或者 OBJ）、模型的中心位置信息（中心纬度、中心经度）。这些信息都是在查找模型文件以及进行模型渲染时需要的索引条件。

在数据服务器上，我们对模型文件以及配置文件进行了压缩存储，从而减少了存储模型文件的磁盘空间，并且可以极大地提高数据的下载速度，在客户端解析配置文件或者渲染模型时再将文件做解压处理。

（3） 三维高精度模型可视化

三维几何实体模型文件和模型配置文件均存储在数字地球的服务器端，当我们需要渲染模型时，首先计算当前视点范围内的地表瓦片的行号和列号。然后根据行号、列号信息在本地数据缓冲池中进行查找。若发现相应的模型配置文件，则解析文件；否则，向 Web 服务器层发出请求下载相应位置的模型配置文件。Web 服务器端接到请求后，由瓦片的行号、列号对模型文件夹进行索引，得到请求的模型配置文件，然后通过数据缓存机制、数据的高效传输与分析机制等将其下载到本地内存中，同时将其写入本地数据缓冲池。

配置文件下载完成后，客户端解析模型文件，得到该瓦片上的所有模型的信息，包括模型的名称、模型的文件类型（3DS 或者 OBJ）、模型的中心位置信息（中心纬度、中心经度）等。然后根据瓦片的行号、列号信息以及模型的名称、类型信息在本地数据缓冲池中查找相应的几何实体模型文件。若发现该模型文件存在，则进行模型渲染；否则，再次向 Web 服务器端发出请求，下载模型配置文件中记录的所有模型的模型文件。Web 服务器端接到请求后，由瓦片的行号、列号结合模型的名称、类型信息对模型文件夹进行索引，得到请求的模型文件，然后通过数据缓存机制、数据的高效传输与分析机制等将其下载到本地内存中，同时将其写入本地数据缓冲池。在就建筑物模型渲染时，通过解析 3DS 或者 OBJ 格式的三维几何实体模型文件，获取到模型的轮廓信息、纹理坐标、材质信息等，然后使用 OpenGL 的接口函数。

4.5.2 地上、地表、地下三维数据发布

1、三维地上景观、建筑物建模及可视化

三维建筑物的构建需要三维空间数据和纹理数据。三维空间数据包括平面数据和高度数据。建筑物平面数据,主要指的是建筑物在三维俯视图中投影到地表平面的轮廓线。三维建筑物的建模主要有基于面结构的建模和基于体结构的建模。前者是通过空间面来表示 3D 空间目标,后者则是侧重三维空间体的表示。

三维地表模型的可视化需要借助三维图形可视化渲染工具来实现,这里可以选用 OpenGL 作为底层可视化渲染工具。OpenGL(Open Graphics Library)是 SGI 公司开发的一个底层三维图形库,几乎在各种操作系统上都有相应的版本。其最显著的特点是平台无关性,可以方便地将应用程序移植到另一个操作系统中。

我们在数字地球上漫游时,到达什么高度就需要渲染建筑物模型。依据 LOD 的基本思想,当视点距离地球表面小于一定距离时才能看得到地球表面的建筑物,而在大于此距离的高度时是无法清楚地看到建筑物的轮廓的。因此,我们根据实际情况设定视点距离地球表面的高度小于 2000m 时才对视区内的建筑物构建模型并进行可视化,在 2000m 之外的高空建筑物的轮廓是无法清楚地看到的,也就不需要绘制。



图 24 三维地上景观、建筑物建模

2、三维地形建模及可视化

数字地形通常有三种不同的表示方式等高线、格网、不规则三角形。在三维地形的可视化中,为了渲染地形地貌的高低起伏,普遍采用数字高程模型 (Digital Elevation Model, 简称 DEM) 方法。数字高程模型常用的数据结构是规则网格 (Grid) 和不规则三角网 (TIN)。其中,由于规则网格的数据结构其顶点呈规则分布,因此

只需要记录数据的基本信息和每个格网点的高程值，就可以很容易的绘制三维地形地貌特征，其数据结构简单、应用操作方便、便于简化数据模型，因此非常适合于大规模三维地形数据的组织和管理。

三维地形的建模及可视化主要包括地形数据的简化、构建地形数据的层次模型、地形数据的渲染三个过程。在三维地形可视化中，通过 LOD 技术调用不同层次的地形瓦片数据实现不同视点下地形的渲染。要表达地形的真实感，可以通过地形数据与影像数据的叠加（即纹理映射）来实现。遥感影像作为地形表面的真实纹理，其内容丰富、现实性强、具有高度真实性，是地表纹理的最重要来源。



图 25 三维地形建模

3、三维地下管网、地质构成建模及可视化

构造地下管网等虚拟场景一般采用三种建模方法：基于模型的绘制方法、基于图像的绘制方法、基于图形和图像混合建模方法。对于埋藏在地下的管网、地质结构等难以进行图像采集，基于图像的绘制方法和图形和图像混合建模的建模方法并不适用，这里最适合采用基于模型的绘制方法。

基于模型的绘制的建模法首先对真实场景进行抽象，用多边形构造三维几何模型，并建立虚拟环境中的光照和材质模型，然后进行纹理映射及控制参数设定，完成模型的构建。构建地下三维模型主要有手工建模、基于标准件的半手工建模和自动建模三种实现方式。手工建模速度慢，使用于单体建模；半手工建模受模型标准件限制，使用于特定的环境；自动建模基于数据自动生成模型，具有生成速度快、精度高的特点，能够有效运用于工程应用中大场景建模。

三维地下管网、地质构造场景庞大，空间分布复杂，变化大、增长速度快，因此使用自动建模的方式更加合适。

三维模型的地质层的剖面，使用多种颜色来区分不同地层结构。

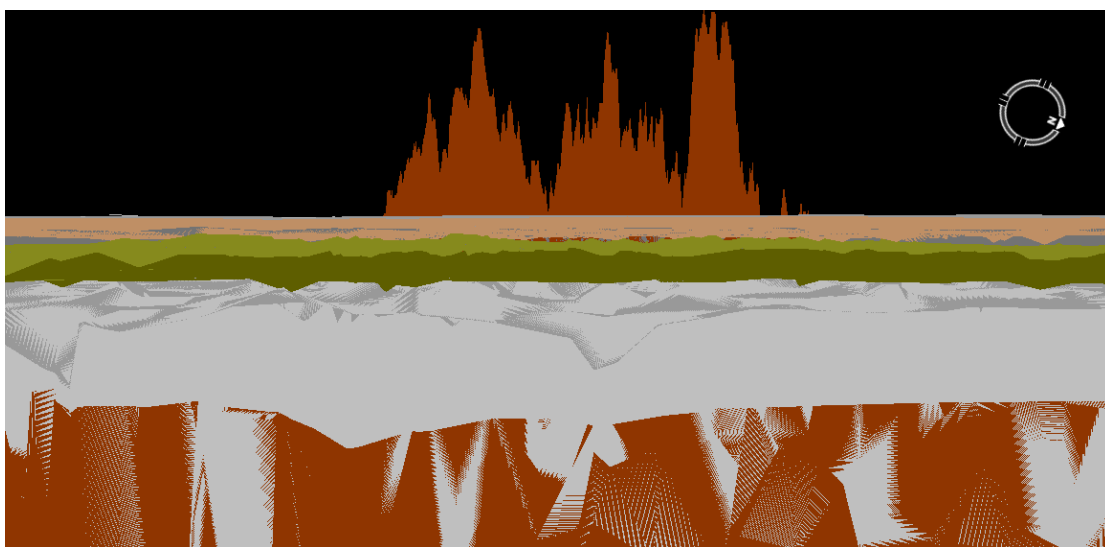


图 26 三维地下地质建模

4.6 三维地质网格地球的分布式部署

4.6.1 分布式系统

分布式系统（distributed system）是建立在网络之上的软件系统。正是因为软件的特性，所以分布式系统具有高度的内聚性和透明性。因此，网络和分布式系统之间的区别更多的在于高层软件（特别是操作系统），而不是硬件。内聚性是指每一个数据库分布节点高度自治，有本地的数据库管理系统。透明性是指每一个数据库分布节点对用户的应用来说都是透明的，看不出是本地还是远程。在分布式数据库系统中，用户感觉不到数据是分布的，即用户不须知道关系是否分割、有无复本、数据存于哪个站点以及事务在哪个站点上执行等。

分布式应用是为适应分布式数据存储的需要、减少集中服务器的负荷而提出的结构。采用面向 Internet 的分布式计算技术，可实现数据服务、GIS 服务、Web 应用服务的分布式部署，灵活应用，能满足各种环境的配置需求。分布式部署方式主要有两种，一种是针对局域网的分布式部署，另一种则是局域网和 Internet 结合的分布式部署。对于局域网的分布式部署方式相对比较简单，数据与数据服务、GIS 服务、Web 应用服务可部署在一台机器上，也可以部署在多台计算机上。

4.6.2 分布式部署方案

为了有效利用分布在全国各地的地质调查网格信息，提高数据的共享程度，及时互通，避免信息孤岛，我们提出了在全国各地分布式部署信息网格的方式，其结构图如下所示：

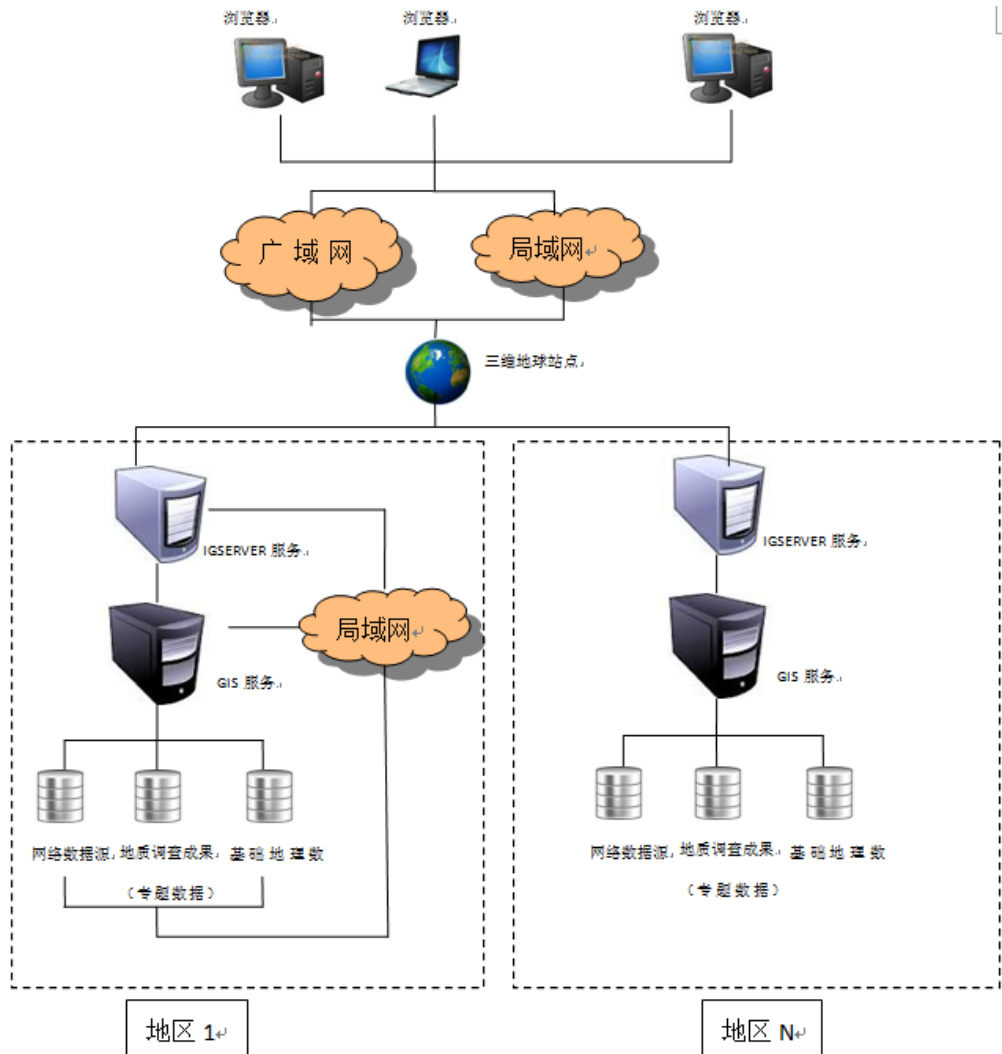


图 27 分布式部署结构图

在该分布式部署方式中，实现了地质调查成果的分布式部署和管理。地调成果可以由关系数据库、空间数据库或文件系统进行组织与管理，分别部署在各个地区的数据服务器中，每个部署数据的地区都需要有提供相应的数据服务和空间分析的 GIS 服务器，再通过同一个地方安装 web 服务器 IGServer，将可以共享的地质调查数据发布到三维地球，提供数据共享以及数据分析和管理服务。互联网中具有权限的用户可以通过浏览器访问发布在网络中的三维地球站点，根据用户提交的对三维地球操作请求，向分布在全国各地的地调局提取相应的数据。

分布式部署的实现首先需要在网络中发布一个站点，使得具有权限的用户能够访问三维地球并请求相应的数据，而访问请求到达指定的部署地区时，通过部署 IGServer 服务器，当客户请求到达 GIS 服务器，经过分析处理提取相应的数据返回给客户端。而提供数据的服务端则需要安装 IGServer 提供 web 服务，安装 MapGIS K9 提供 GIS 服务，以及 SQL Server 或 Oracle、DB2 提供部分属性数据的存储。每一个部署数据的地区都必须有自己的 web 服务器和 GIS 服务器，不支持 web 服务和 GIS 服务在广域网上的分布式，但可以将这些服务部署在局域网上的不同机器上，实现网内的分布式部署。

第五章 质量监控体系与质量保证

5.1 组织保证

课题负责人代表课题承担单位与参加单位对项目负责，并领导课题的实施；本课题设立一个项目管理组，负责整个课题的软件开发规划、技术研讨、技术咨询、技术监测，把握技术方向，解决技术关键，保障决策的科学化。项目组下设研究小组，每个小组负责一个方面的具体研究工作，每组设立组长一人，组长作为项目管理组成员，参与技术研讨、方案制定等工作；组长负责本小组的研究开发按照计划实施，并负责贯彻和检查有关软件开发规范；研究小组组长向课题负责人负责。

系统实施过程中将成立项目实施组织，包括项目过程管理组织、项目执行过程组织和项目支撑组织等。

（1）项目过程管理组织

项目过程管理组织将负责项目的立项、项目的规划、项目的监控、风险管理、需求管理和结项管理等项目过程的管理。

（2）项目执行过程组织

项目执行过程组织将负责本课题的多源数据的集成研究、提高三维可视化显示效率的研究、系统设计与开发、系统测试、性能改进、安装部署和集成验收等系统内容的具体实施。

① 三维可视化相关研究

成立专门的可视化技术研究小组，对网络数据源的三维可视化技术、异构空间数据的集成方法以及异构数据集成模型进行研究和实现研究。研究如何提高海量的地图数据的显示效率，包括适量地图的裁剪、适量地图的合并、以及数据的异步请求方式等。

② 系统设计与开发

在可视化研究成果的基础上，进行系统的设计与开发工作。首先搭建出一个原型系统，实现主要的功能点，然后基于原型系统，设计并开发具体的突发地质灾害应急预案可视化系统，包括数据库设计、输入输出设计、处理过程设计、用户界面设计、代码设计等。

系统实施阶段将继承设计阶段的工作成果，将技术设计转化为物理实现。系统实施包括系统编码、系统测试和系统安装等主要工作。

③ 系统测试

系统测试是质量可靠性的关键，运用一定的测试技术和方法，根据所制定的测试计划，通过单元测试、集成测试、接口测试和系统测试等步骤，查找发现可能存在的问题。

④ 性能改进

根据系统测试的结果，对系统的可视化效率性能进行优化，对界面的美观性方面进行改进。

⑤ 安装部署

在确认承建方提供的产品无误的情况下，承建方将派遣有经验的技术人员组成工作小组到现场实施，负责安装、部署、配置和调试，主要目标是使产品能够在用户指定的环境中正常运行，并能够完全支持突发地质灾害应急响应支撑项目及本项目的各项需求。

承建方在安装产品之前，将先制定详细和可行的安装调试计划，并对业主方相关技术人员进行现场培训，然后在业主方相关技术人员监督下进行软件的安装、调试。不得在现场安装未经批准的其他产品。

⑥ 集成验收

在符合要求的产品安装、部署完成，与设备集成后，承建方将接受业主方的系统集成验收。

在系统集成开发阶段，承建方将派驻 2 名以上具有 3 年以上所投产品定制开发经验的主要技术人员免费驻现场服务 3 个月，提供技术支持、参与应用系统中与地理信息系统相关内容的设计和开发。此阶段将完成系统的编码与单元测试，并在总集成方的统一部署和安排下，完成产品在指定环境上的配置和系统集成，保证系统软件满足平台的总体要求。

(3) 项目支撑组织

项目支撑组织将负责系统的配置管理、系统执行的质量保证、系统执行过程中的培训管理和系统集成后的技术支持和服务维护。从多个方面保障软件开发过程的动态化、可视化管理以确保项目按计划保质保量在合同规定的期限内完成，并保障软件的稳定性、实用性、扩展性。

5.2 质量评述

以国家系统工程施工质量验收统一标准及验收规范等为依据，全面实现施工合同约定的质量目标；对工程项目施工全过程实施质量控制，以质量预控为重点；对工程项目的人、机、料、法、环等因素进行全面的质量控制，将地质大学的质量管理体系、技术管理体系和质量保证体系落实到位；严格执行有关材料、测试制度和设备检验制度。系统测试对研究成果进行了功能的完善性、软件可靠性、应变能力、安全性等多个方面的测试，且得出了与预期相近的结果，有效的认证了系统的质量。

第六章 取得成果

本课题研究成果的主要界面如下所示：

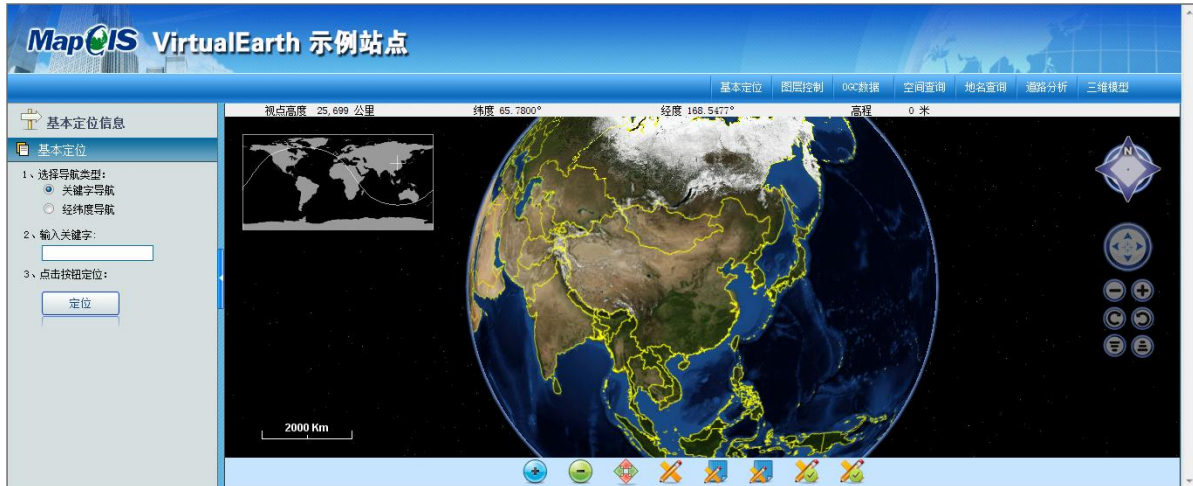


图 28 系统主界面

6.1 多源数据的三维可视化

本系统中不仅实现了本地数据，如卫星影像数据、矢量数据、瓦片数据在三维地球上的可视化，并且利用 SIG 技术实时的调用网络上的图像数据，如下面展示的天地图在三维地球上的可视化集成效果，实现了多源数据的可视化集成。

(1) 在三维地球上添加全球影像数据的显示效果：



图 29 全球影像数据显示

高清卫星影像的显示：



图 30 高清卫星影像显示

(2) 在线地图的三维可视化集成



图 31 来自网络的天地图注记



图 32 来自网络的天地图影像

其详细配置步骤为：在示例站点的安装路径如 D:\Program Files\MAPGIS VirtualEarth\Web_Sample\VirtualEarthDemo\WebRoot\applet\config 中的 EarthConfig.xml 文件中，于用户自定义标签即<CustomLayers>中添加天地图的配置

内容:

```
<LayerInfo>
  <DataType>10</DataType>
  <AliasName>天地图影像</AliasName>
  <LayerName>tdtyingxiang</LayerName>
  <Version>1</Version>
  <VisibleStartLevel>0</VisibleStartLevel>
  <VisibleEndLevel>16</VisibleEndLevel>
  <Visible>false</Visible>
  <DrawType>png</DrawType>
  <Opacity>1</Opacity>
  <Sector>-180,-90,180,90</Sector>
  <RequestPage>GetTianditu.jsp</RequestPage>
  <TileSize>256</TileSize>
  <Delta>45</Delta>
</LayerInfo>
```

(3) 矢量数据的显示效果:



图 33 矢量数据显示

(4) 瓦片数据的显示效果:



图 34 瓦片数据显示

我们主要研究了在 Web 架构中通过构建全局虚拟金字塔结构模型进行异构在线地图数据的集成方法。Web 的架构模式提高了异构在线地图数据集成的效率，全局虚拟金字塔结构模型的构建，实现了异构在线地图数据的无缝集成。

6.2 地名查询

我们将 google 地图提供的地名查询服务有效的集成到三维地球中，实现全国地名数据的查询和分析，其实现效果如下所示。例如，搜索关键字武汉大学，得到两个有关武汉大学的地名信息点，于是分别将这两个查询结果展示在地球上，点击各地名上的标注，将显示详细的地名信息，包括经纬度坐标、地名全称、邮政编码等。



图 35 地名查询

6.3 道路分析

在路径分析页面输入查询条件，如下：

-
- 7、向右转，进入**文安街** 0.8 公里
 - 8、稍向右转，继续沿**文安街**前行 0.1 公里
 - 9、向左转，进入**静安路** 0.3 公里

到达终点

民主路 全程 7.0 公里,14 分钟

从起点

- 1、从**校园南路**向西方向，前往**迎宾路** 0.4 公里
- 2、在第一路口向左转，朝**校园西路**行进 0.2 公里
- 3、在第一路口向右转，朝**茶港路**行进 0.3 公里
- 4、稍微向右转，进入**洪山侧路** 1.1 公里
- 5、向左转，进入**东一路** 0.3 公里
- 6、在第一路口向右转，朝**八一路**行进 0.5 公里
- 7、继续前行，上**广场环路** 0.4 公里
- 8、向左转，前往**民主路** 0.2 公里
- 9、继续直行进入**民主路** 1.6 公里
- 10、向左转，进入**中山路** 0.9 公里
- 11、稍向右转，继续沿**中山路**前行 0.3 公里
- 12、在第三路口向左转，朝**紫阳东路**行进 0.2 公里

到达终点

武珞路 全程 7.5 公里,16 分钟

从起点

- 1、从**校园南路**向西方向，前往**迎宾路** 0.4 公里
- 2、在第一路口向左转，朝**校园西路**行进 0.7 公里
- 3、向右转进入**珞珈山路** 0.5 公里
- 4、稍微向左转，进入**珞珈山路** 1.0 公里
- 5、向右转，进入**武珞路** 3.0 公里
- 6、在**交岔路口**继续向右 0.3 公里
- 7、从环岛的 3 出口上**中山路** 0.6 公里
- 8、稍向右转，继续沿**中山路**前行 0.3 公里
- 9、在第三路口向左转，朝**紫阳东路**行进 0.2 公里

到达终点

这里列出了多种行进方式，分别用标黑的粗体字表示各条路径的名称。

6.4 提高三维可视化显示效率

矢量数据在地球上的显示如下。



图 38 矢量数据显示

矢量数据的动态瓦片裁剪的显示效果如下图。



图 39 矢量数据的动态瓦片裁剪显示效果图

矢量数据以瓦片地图的展示方式实现了动态刷新浏览页面：



图 40 动态刷新浏览

6.5 地上、地表、地下三维一体化的发布系统

6.5.1 地上数据发布后可视化

下面的两幅图分别是同一建筑物的不同视角的可视化效果示例：



图 41 三维建筑物可视化效果图



图 42 三维建筑物可视化效果图

注：当景观数据范围较小时，不易查看三维景观所在的位置，则可通过三维模型数据文件中景观的中心经纬度来定位目标点，放大此目标点到一定程度则可清晰查看三维模型的三维效果。

6.5.2 地表数据发布过程与可视化

下图是地表起伏的显示效果：



图 43 地表起伏显示效果图



图 44 地表起伏显示效果图

6.5.3 地下数据发布过程与可视化

下面是三维层状地质体模型实现的三维效果如下所示：

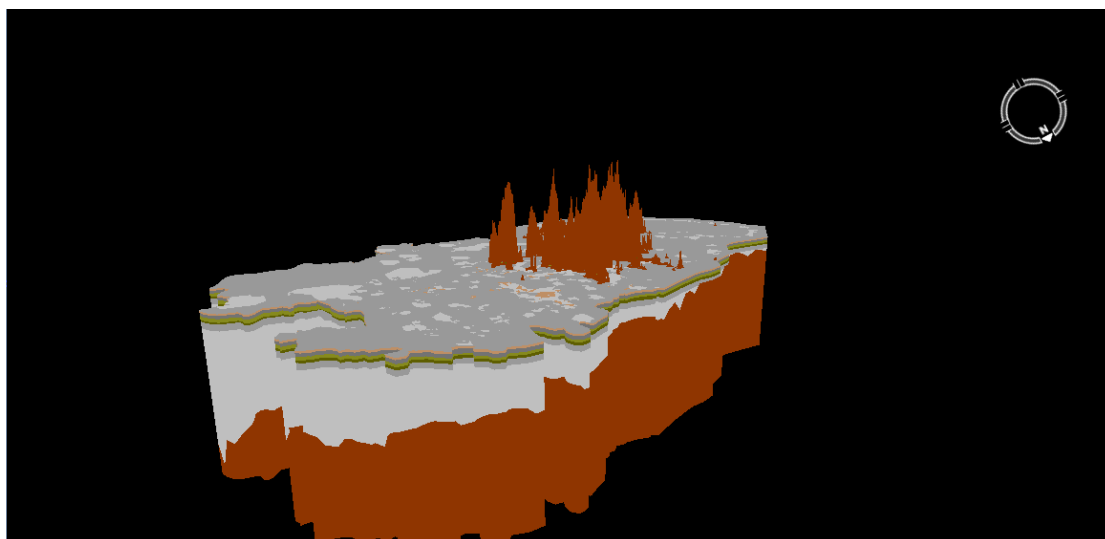


图 45 三维层状地质体模型三维效果图

三维模型的地质层的剖面，使用多种颜色来区分不同地层结构。

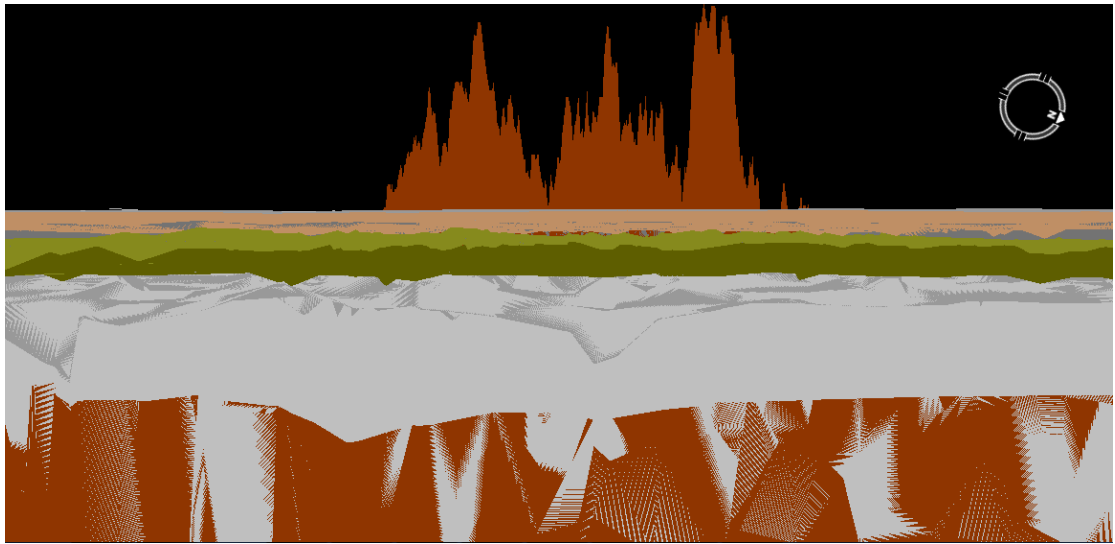


图 46 三维模型的地质层的剖面图

下面是地下地层结构的仰视图，能够清晰的看出层状地质的起伏形态。

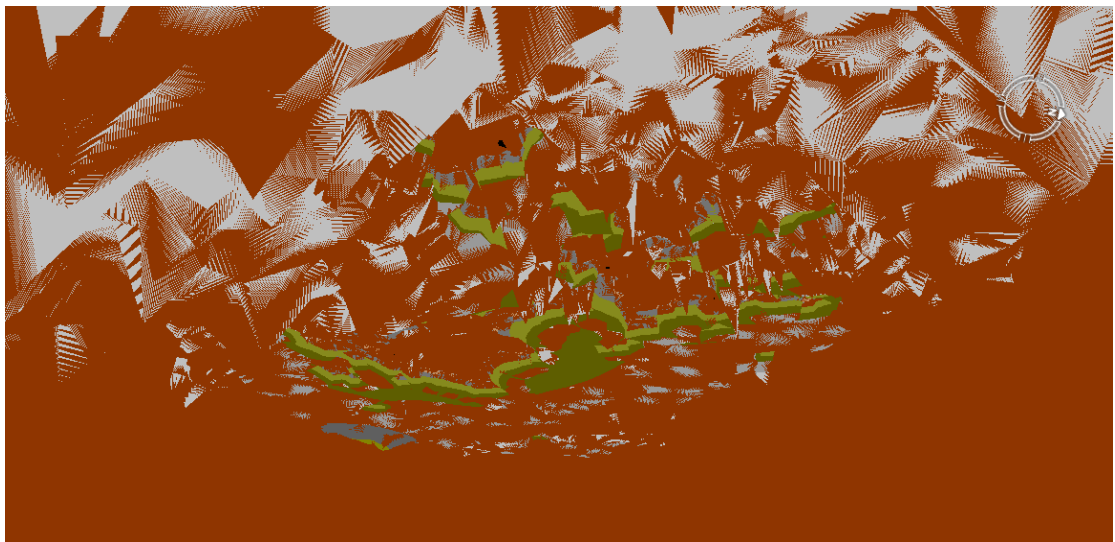


图 47 三维地下地层结构的仰视图

主要参考文献与资料

- [1] 李南樱.基于网格的数字地球研究[J].电脑与信息技术, 2006, 14(5):1-2
- [2] 数字地球. [EB/OL].[2010-01].http://baike.baidu.com/view/8443.htm?fr=ala0_1.
- [3] 郑文峰.面向服务的空间信息共享[D].成都理工大学, 2008:8-13.
- [4] 沈荣骏.数字地球在应用中求发展[J].装备指挥技术学院学报.2009,20(1):1-5
- [5] 唐桂文.基于数字地球平台的地理信息服务[D].首都师范大学, 2008: 15-20.
- [6] 施加松.网络三维GIS若干技术问题研究[D].解放军信息工程大学, 2006: 7-11.
- [7] 周倜.数字地球及国家空间数据基础设施标准化建设[J].航天标准化, 2009(1).
- [8] 郭华东.数字地球: 10年发展与前瞻[J].地球科学发展, 2009,24(9).
- [9] 马照亭, 潘懋等. 三维数字地形图上漫游路径的选取与显示[J].计算机应用研究,2004(4).
- [10] 陈健, 高井祥.三维景观网络发布的研究与实现[J].测绘工程,2006,15(3).
- [11] 杜剑侠, 李凤霞, 战守义. 面向数字地球的海量地形技术研究[J].中北大学学报,2007,28(2).
- [12] 吴信才等. 地理信息系统原理与方法(第二版) [M]. 北京: 电子工业出版社, 2009.
- [13] 尚武.网络地理信息系统(WebGIS)的现状 & 前景[J].地质通报, 2006, 25(4):533—537.
- [14] 卢亚辉, 杨崇俊.基于Web Service的WebGIS系统的研究[J].计算机工程与应用, 2003, 39(25):153—159.
- [15] 蒋杰, 吴玲达, 徐江斌. 基于数字地球的洪灾演进模型表现[J].计算机工程与应用.2009,45(36).
- [16] 苗放, 叶成名, 刘瑞, 孔祥生, 李康荣, 徐松浦, 张远红. 新一代数字地球平台与“数字中国”技术体系架构探讨[J].测绘科学.2007,32(6).
- [17] 吴慧欣.三维GIS空间数据模型及可视化技术研究[D].西北工业大学, 2007.
- [18] 李军.三维GIS空间数据模型及可视化技术研究[D].国防科学技术大学, 2000.
- [19] 李德仁.数字地球与“三S”技术 [OL].<http://www.gischina.com/maindoc/simchin/gisforum/thesis.2001>.
- [20] 翟巍.三维GIS中大规模场景数据获取、组织及调度方法的研究与实现[D].大连: 大连理工大学, 2003.
- [21] 于文洋.面向数字地球的三维景观构造关键技术研究[D].北京: 中国科学院遥感应用研究所, 2006.
- [22] 张立强.构架三维数字地球关键技术研究[D].北京: 中国科学院研究生院.2004.
- [23] 郑琪. 基于Web的地形三维可视化技术研究 & 实现[D], 成都: 电子科技大学, 2005.
- [24] 罗显刚, 谢忠, 徐欣.基于GML的WMS研究与实现[J].计算机应用, 2004, 24(12):132-135.
- [25] 崔铁军, 郭黎, 白振会, 张斌. 基于网格的地理空间信息服务关键技术研究[J].测绘科学技术学报, 2007, 24(5).
- [26] 钱贞国.面向互操作的分布式网格地理信息系统研究[D].北京: 中科院遥感应用研究所, 2004.
- [27] 崔铁军, 郭黎, 白振会, 张斌. 基于网格的地理空间信息服务关键技术研究[J].测绘科学技术学报, 2007, 24(5).
- [28] 张雷.对OpenGIS标准的分析和空间数据引擎的实现[D].杭州: 浙江大学, 2002: 14.
- [29] 王宇翔.分布式网络地理信息系统研究[D], 北京: 中国科学院地图学与地理信息系统, 2002.
- [30] 颜辉武, 吴涛, 王方雄.网络地理信息系统[M].北京:测绘出版社, 2007.
- [31] 何建邦, 阎国年, 吴平生.地理信息共享的原理与方法[M].北京:科学出版社, 2003.
- [32] 钟珞, 潘媛媛, 徐勇, 曹皓.分布式异构空间数据共享研究[J].计算机应用与软件, 2005, 22(10):52-54
- [33] 郑南, 郑扣根.支持分布式异构地理数据共享 & 集成的GIS设计与实现[J].计算机应用研究, 2004, (8):200-205
- [34] 郑文峰.面向服务的空间信息共享[D].成都理工大学, 2008:8-13.
- [35] 李琦, 曾澜, 苗前军, 史文勇.空间信息基础设施 & 互操作[M].北京:科学出版社, 2003.
- [36] 张建兵.基于网格的空间信息服务关键技术研究[D].中国科学院研究生院, 2006: 13-58.
- [37] 马照亭, 潘懋等. 三维数字地形图上漫游路径的选取与显示[J].计算机应用研究,2004(4).

-
- [38]高小伟, 王萍, 王春泉.网络三维GIS的设计与实现[J].GIS技术.2006,3.
- [39]夏启兵, 李胜全, 张明亮. 基于数字地球的航标信息系统技术研究[J].海洋测绘.2009,29(1).
- [40]陈鹏, 孟令奎, 宋杨. 复杂三维空间对象的模型可视化研究[J].GIS技术.2007,4.
- [41]张立强. 构建二维数字地球的关键技术研究[D]. 中国科学院遥感应用研究所, 2004,6.
- [42]尚武.网络地理信息系统(WebGIS)的现状 & 前景[J].地质通报, 2006, 25(4):533—537.
- [43]杨阳.Google Earth 和 World Wind 平台的对比以及进行 GIS 开发的可行性、各自的优缺点[J].科技信息.
- [44]World Wind 技术手册.<http://www.docin.com/p-1622413.html>.
- [45]Virtual Earth. [EB/OL].[2009-08].
<http://hi.baidu.com/mstcinxdu/blog/item/df60ae1658adfc4321a4e951.html>.
- [46]Google Earth 和 MS Virtual Earth: 虚拟地球简介. [EB/OL].[2006-03].
<http://www.cnblogs.com/StinJia/archive/2006/03/02/341601.html>.
- [47]Cignoni P, Ganovelli F, Gobbetti E, et al. Planet-sized batched dynamic adaptive meshes (prbdam). In: Proceedings IEEE Visualization 2003. Seattle, WA., USA.: IEEE Computer Society Press, 2003. 147-154.
- [48]Cignoni P, Ganovelli F, Gobbetti E, et al. Interactive Out-of-Core Visualisation of Very Large Landscapes on Commodity Graphics Platform. In: Balet, et al., eds. ICVS 2003. Berlin Heidelberg: Springer Verlag Press, 2003. 21-29.
- [49]Platings M, Day A M. Compression of Large-Scale Terrain Data For Real-Time Visualization Using a Tiled Quad Tree. Computer Graphics, 2004, 23(4): 741-759.
- [50]Clark James. Hierarchical Geometric Models for Visible Surface Algorithms [J]. Communications of the ACM (S0001-0782). 1976, 19(10): 547-554.
- [51]Geography Markup Language (GML) V3.0 Implementation Specification, OpenGIS Consortium inc.
- [52]Han, K. Development of an Interoperable Geographic Information System Platform for Transportation Application [D]. Canada: Department of Civil and Geological Engineering University of Manitoba, 2001: 1-148.
- [53]Pundt, H & Y. Bishr. Domain ontologies for data sharing an example from environmental monitoring using field GIS [J]. Computer & Geosciences, 2002, (28): 95-102.
- [54]Kim, T. J. Metadata for geo-spatial data sharing: A comparative analysis [J]. The Annals of Regional Science, 1999, (33): 171-181.
- [55]Nogueras-Iso, J. F. J. Zarazaga-Soria, R. Bejar, P. J. Alvarez & P. R. Muro-Medrano. OGC Catalog Services: a key element for the development of Spatial Data Infrastructures [J]. Computers & Geosciences, 2005, (31): 199-209.
- [56]Frehner, M & M. Braundli. Virtual database: Spatial analysis in a Web-based data management system for distributed ecological data [J]. Environmental Modelling & Software. 2006 (21): 1544-1554.
- [57]Jia, W. J. Y. M. Chen & J. Y. Gong. Implementation of OGC Web Map Service Based On Web Service [J]. Geospatial Information Science, 2004, 7(2): 148-152.