

## 摘要

网络地理信息系统(WebGIS)作为网络技术和 GIS 技术的结合点,从其诞生以来,它的发展就与计算机技术的进展息息相关。从 90 年代开始,随着以 Internet 为代表的网络技术的迅猛进步, WebGIS 的应用得到了飞速发展,也出现了一系列的问题。本课题研究的目的是根据 WebGIS 发展的需求,研究目前的实现技术,探讨基于 .NET 技术的 WebGIS 可行性,并将其应用到具体系统开发实践中。

.NET 是用以创建下一代互联网运行平台的一系列的技术。.Net 的核心技术是 WebServices,它是一个具有革命性的,基于标准的框架结构。它可以让分布式空间处理系统使用目前广为流行的技术,例如 XML 和 HTTP 来通过 Web 进行互相通讯。它提供了与厂商无关的,可互操作的框架结构来对多源、异构的空间数据进行基于 Web 的数据发现、集成、分析、决策支持和可视化。用户可以通过在任何地方、任何时间、通过任何设备来获取数据和互相交流。

本文着重于新的网络技术及其在地理信息领域的应用。首先研究了国内外 WebGIS 的发展现状,并针对目前主流的 Web 地图发布技术的特点和实现方式进行了详细的比较。其次着重研究了 .NET 以及其核心 Web Services 技术,并学习了 OGC 制定的一系列 Web Service 相关的服务框架、协议栈和规范。

然后在对 OGC 相关标准研究的基础上,本文提出了基于 .Net 及地图平台 MapXtreme 来实现电网中配电线路和设备管理的 WebGIS 解决方案,详细阐述了关键技术的实现方法以及应用服务器和客户端的设计,并最终实现了“基于 .NET 的电网配电线路和设备 WebGIS 管理系统”。由于采用了瘦客户端的应用模式,所以客户端的实现主要是界面设计以及与应用服务器交互两部分内容。应用服务器端主要是利用 Web Service,向客户端提供各种服务功能。从而用 Web Service 的方法实现了一个三层结构的可扩展的 WebGIS 系统。

最后,本文对基于 Web Service 的 WebGIS 系统的研究和实现做了总结,并提出了在 Web GIS 的应用中需要进一步研究和解决的问题以及前景展望。

关键词: .NET 架构, 网络地理信息系统, OpenGIS, OGC, 地图服务平台

## Abstract

As a conjunction of Internet technology and GIS technology, the development of WebGIS has closely related with the development of Computer technology. From 90s, as the network technology developed quickly, which presented by Internet, WebGIS applications have developed rapidly. Meanwhile, there also appeared many questions in WebGIS systems. Some of these questions are brought by Web software, such as the cross-platform and cross-language problems. Others are brought by GIS software, such as the data interoperative. So, we must do research in both GIS and Web areas to efficiently solve these problems.

.NET is a set of software technologies for building next generation information, people, systems, and devices. Web Services are core of .Net. It allows distributed geoprocessing systems to communicate with each other across the Web using familiar technologies such as XML and HTTP. Web Services provide a vendor-neutral, interoperable framework for web-based discovery, access, integration, analysis, exploitation and visualization of multiple online geodata sources..NET expands opportunities to act on data and communicate anywhere, anytime, on any device.

This dissertation focuses on the introduction of new coming Internet technology and its application in GI domain. First, It introduces the development of WebGIS and the architecture of the popular Web Mapping technologies and compares the characteristic and limitation of them. Second, It introduces the .NET and its core-Web Service, with Service frame, protocols, criterion.

Then Based on the research about OGC standards, I present the detail design and implementation procedure of this WebGIS electric power net displaying information of electric line and equipment, which include the design and implementation of Application Server and Client. Because I use thin client application mode, so the implementation of the client just includes two aspects: the interface design and the interaction with Application Server. In the implementation of the Application Server, I mainly use the Web Service to provide the various functions to client. By these steps, I implement a 3 tier scalable WebGIS system by Web Service.

Finally, I conclude the research and implementation of WebGIS system and then present the problems that need to be solved further and the vista of Web Service.

**Key words:** .NET framework, WebGIS, GIS WebServices, OpenGIS, OGC,  
Map Service Platform

# 第1章 绪论

地理信息系统(Geographical Information System, 简称 GIS)是在计算机软硬件支持下,对地理空间数据进行采集、输入、存储、操作、分析和建模,以提供对资源、环境及各种区域性研究、规划、管理及决策所需信息的人机系统<sup>[1]</sup>。

网络地理信息系统(WebGIS)指在 Internet 或 Intranet 网络环境下应用采用了 WWW 协议的一种兼容、存储、处理、分析和显示与应用地理信息的计算机信息系统<sup>[2]</sup>。国际学术界把类似于万维网地理信息系统称之为 WebGIS (万维网地理信息系统),这主要是由于大多数的客户端应用采用了 WWW 协议。它的基本思想就是在互联网上提供地理信息,让用户通过浏览器浏览获得地理信息系统中的数据和服务。

## 1.1 选题背景

与传统的地理信息系统比较,WebGIS 它具有四个特点<sup>[3]</sup>:

1) 更广泛的客户访问范围。客户可以同时访问多个位于不同地方的服务器上的最新数据,而这一 Internet/Intranet 所特有的优势大大扩展了 GIS 的数据管理能力,增强了对空间数据管理的时效性。

2) 客户端平台独立性。无论客户机是何种操作系统,只要支持通用的 Web 浏览器,用户就可以访问 WebGIS 数据。

3) 更简单的操作。要推广 GIS,使 GIS 系统为广大的普通用户所接受,而不仅仅局限于少数受过专业培训的专业用户,就要降低对系统的操作难度。通用的 Web 浏览器无疑是降低操作复杂度的最好选择。

4) 平衡高效的计算负载。传统的 GIS 大都使用文件服务器结构的处理方式,其处理能力完全依赖于客户端,效率较低。而当今一些高级的 WebGIS 能充分利用网络资源,将复杂的处理交由服务器执行,而对简单的操作则由客户端直接完成。这种计算模式能灵活在服务器端和客户端之间合理分配处理任务,提高网络计算资源的利用效率。

网络地理信息系统涉及到在网络(Internet/Intranet)环境下,地理信息(包括与其相关的图像、图形和与此相关的文本数据)的模型、传输、管理、分析、应用的理论与技术。作为地理信息系统的一种新形式,WebGIS 无论是在理论研

究，还是在应用方面都具有非常重要的社会意义。

## 1.2 WebGIS 的现状和存在问题

目前构建 WebGIS 的主要技术包括基于服务器的技术、基于客户端的技术和基于服务器/客户端的混合技术<sup>[4]</sup>。其中基于服务器的技术包括 CGI, Server API(包括 ASP, JSP, Servlet 等)等方式, 基于客户端的技术包括 Java Applet, Plug-in 等方式。基于服务器/客户端的混合技术一般综合使用基于服务器和客户端的技术, 以发挥两者的长处, 弥补两者的缺点。

国内目前流行的 WebGIS 平台软件中 Supermap 采用的是服务器端的 ASP 技术, GeoSurf 采用的是客户端的 Java Applet 技术, 而 ArcIMS, GeoBeans 等采用的是基于服务器/客户端的混合技术。它们都是采用了 COM 或者 Java Beans 等组件方式来构建 WebGIS 软件<sup>[5-7]</sup>。随着 WebGIS 平台软件的不断发展和成熟, 基于这些平台软件之上的应用系统也蓬勃发展起来。目前, 可以提供地图发布、地图浏览和查询的网站已经数不胜数。如 National Geographic, USGS, Go2Map 等, 这些应用系统从浏览功能、使用方法、地图显示、功能、交互性、定制功能、特殊功能等几方面有着不同的特点及性能。

但是因为 Web 的种种限制和 GIS 本身的一些特点, 这些 WebGIS 的实现技术存在着很大的问题。主要表现在以下几个方面:

### 1) 无法实现异构空间数据互操作<sup>[8-13]</sup>

现有的 WebGIS 系统都是为某一特定的 GIS 数据及其应用而设计的, 如果用户同时需要查看其它空间数据库中的数据, 甚至想把这些数据整合起来, 都是非常困难的。因为这些 WebGIS 系统采用的空间数据技术基础决定了它们的封闭性。虽然网络上的空间信息资源在不断增长, 但由于行业管理和数据安全的原因, 这些空间信息资源大多是面向行业的、依赖于特定的支撑环境和运行环境。他们各自独立、相对封闭、无法互相沟通和协作, 形成了空间信息孤岛, 难以满足 Internet 上与空间信息相关的综合决策的需要。

### 2) 无法实现跨平台<sup>[14]</sup>

分布式的应用程序逻辑需要使用分布式的对象模型, 诸如: 微软的 DCOM, OMG 的 CORBA 或 Java RMI 等。通过使用这些基本结构, 开发人员可使用本地模型所提供的丰富资源, 并可将服务置于远程系统中。但是, 这些系统有一个共

同的缺陷，那就是它们要求服务的客户端与系统提供的服务本身之间必须进行紧密耦合，即要求一个同类基本结构。这样的系统往往十分脆弱，如果一端的执行机制发生变化，那么另一端便会崩溃。因此，使用这些平台构建的 WebGIS 系统将无法实现跨平台的数据访问。这就需要更通用的模型来将这些分布式对象模型概括抽象出来，以在更高的抽象层上实现跨平台。

### 3) 开发、调试和维护的困难

对于 Web 开发人员来说，要创建、测试和设置结构清晰、运行稳定的 WebGIS，目前还没有合适的工具和模式可以使用和借鉴。由于 Web 本身将内容的表现和运行逻辑结合在一起，所以，Web 软件很难实现模块化，也就无法实现软件复用，这加大了开发和调试的工作量和难度。由于常常是针对特定应用编写特定代码，如果应用需求和应用模式一旦改变，那么大部分的代码都必须重新修改以适应新的应用环境，从而增加了维护的困难。基于这种 Web 开发模式开发的 WebGIS 系统，从设计、开发、应用到维护，很难以连贯、有效的方式注重软件的整个生命周期。

## 1.3 研究课题的提出

近年来，GIS 的应用需求十分旺盛，但由于目前 GIS 的这些局限使得软件之间数据交换、数据共享以及软件复用极为困难<sup>[15]</sup>。为解决数据资源的共享问题提出了进行地理信息系统互操作，开发开放式的地理信息系统 (OpenGIS)<sup>[16][40]</sup>。OpenGIS 是指在计算机网络环境下，根据行业标准和接口所建立起来的 GIS，是为了使不同的地理信息系统之间具有良好的互操作性，以及在异构分布式数据库中实现信息共享的途径<sup>[17]</sup>，它将 GIS 技术、分布处理技术、面向对象方法、数据库设计及实时信息获取方法有效地结合起来，成为 GIS 的一个重要的发展趋势。

Open GIS 是通过 OGC 制定的一系列开放式地理空间数据互操作规范来实现的。它提供了地理数据和地理操作的交互性和开放性的软件开发规范。与传统的 GIS 相比，OGIS 建立起通用的技术基础以进行开放式的地理信息处理。

其特点是：

- 1) 互操作性：不同地理信息系统软件之间连接、信息交换没有障碍。
- 2) 可扩展性：硬件方面，可在不同软件、不同档次的计算机上运行，其性

能和硬件平台的性能成正比；软件方面增加新的地学空间数据和地学数据处理功能。

3) 技术公开性：开放的思想主要是对用户公开，公开源代码及规范说明是重要的途径之一。

4) 可移植性：独立于软件、硬件及网络环境，不需修改便可在不同的计算机上运行。

除此之外，还有诸如兼容性、可实现性、协同性等特点。

.NET 是微软公司提出的一种分布式运算的框架，以 XML 为基础，以 Web 服务为核心，辅以其他各种技术实现，意在利用 Internet 上强大的计算资源和丰富的带宽资源，提高工作效率。.NET 技术是围绕 Web 服务展开的。要理解 .NET，我们必须先了解 Web 服务。

传统分布式的应用程序逻辑需要使用分布式的对象模型，诸如分布式组件对象模型 (DCOM)、对象管理组织的公用对象请求代理程序体系结构 (CORBA) 或 Sun 的远程方法调用 (RMI)。这些系统有一个共同的缺陷，那就是它们无法扩展到互联网上。它们要求客户端与系统提供的服务本身之间必须进行紧密耦合，即要求一个同类基本结构。

要求提供紧密耦合的基本结构本无可厚非，但当各个公司需要相互合作、或信息技术提供商扩大业务范围时，便很难实现单一而统一的基本结构。我们根本无法保证希望与之进行远程通信的管道的另一端具备所有我们需要的基本结构，因为对于它使用的操作系统、对象模型或编程语言，我们可能一无所知。

相反，Web 服务彼此是松散耦合的。连接中的任何一方均可更改执行机制，却不影响应用程序的正常运行。通过将紧密耦合的、高效的 N 层计算技术与面向消息的、松散耦合的 Web 概念相结合来实现的。我们将这种计算方式称为 Web 服务。

### Web Service 与 GIS 的结合

Web Service 为分布式互操作的软件系统提供了很好的前景，面向 Web 服务的系统应用和系统集成，将成为下一代 Internet 应用的主流技术<sup>[49]</sup>。而 Web Service 是实现新一代(第四代)GIS 的重要手段。第四代 GIS 的目标是由以系统为中心向以数据为中心，实现空间数据共享与服务的转变，成为 OS、DBMS 之上的主要应用集成平台，Web Service 正是这种平台实现的基本方式。

## 1.4 课题在国内外的研究现状

国际上, 已有许多公司推出了 WebGIS 平台产品, 如 MapInfo 公司的 MapXtreme, ESRI 公司 ArcIMS, Autodesk 公司的 MapGuide, Intergraph 公司的 GeoMedia Web Map 等。国内也有公司和科研机构推出了自己的 WebGIS 产品, 如: GeoSurf, GeoBeans 等。对几个主要 WebGIS 产品的技术特征比较如表 1-1 所示。尽管 WebGIS 产品众多, 而且已经应用到大型的旅游、交通等工程项目当中, 但从总体上讲, 所能实现的功能还很有限, 不能完全满足用户的要求, WebGIS 在理论体系和实现技术方法上仍然有不断完善的必要。

表 1-1 WebGIS 产品的技术特征比较

WebGIS 产品	MapXtreme2004	GeoMedia Web Map	ArcIMS	MapGuide	GeoBeans
服务器端操作系统	Windows NT	Windows NT	Windows NT	Windows NT	Windows NT
Web 服务器	Microsoft IIS	Microsoft IIS	Microsoft IIS/ Netscape	支持 CGI 的 Web Server	Microsoft IIS
应用服务	ODBC	ODBC	ODBC Arc View MapObjects 应用	ODBC	ODBC JDBC MapServer
服务器端运行模式	CGI、NSAPI 或 ISAPI		CGI	CGI	CGI 或 ISAPI
客户端浏览器	支持 HTML 的任意浏览器	IE/Netscape	支持 HTML 的任意浏览器	IE/Netscape	IE/Netscape
客户端运行模式	可选择下载 JAVA 小程序	ActiveCGM 插件	Java Applet ActiveX 控件	需要安装 MapGuide Viewer 插件	Java Applet ActiveX 控件
客户端地理信息格式	栅格	栅格/矢量	栅格/矢量	矢量	栅格/矢量

### Web 服务在 WebGIS 方面的国内外研究状况

#### 1 Web 服务在 WebGIS 方面的国外研究和应用状况

##### 1) OpenGIS 的地理信息网络服务规范

OGC 作为全球最大的空间信息互操作规范的制订者和倡议者, 在参照

ISO/TC211 标准的基础上也制订了相应的地理信息服务规范, 目前主要有四种地理信息服务的相关规范: (1) Web Map Service; (2) Web Feature Service; (3) Web Coverage Service; (4) Web Register Service。为了研究 Web 服务对网络地理信息服务的影响, OGC 启动了 OWS (OGC Web Service) 计划。

#### 2) ESRI 的 ArcWeb Service

ESRI 的 ArcGIS 已经是 COM 组件式 GIS 系统的集大成者, 目前 ESRI 在其产品中开始加强 Java 的份量, 包括 ArcExplore, MO2J 等产品的开发, 为开发 Web 服务的 Java 版作着一些准备。在即将发布的 ArcGIS9 中将提供基于 Web Service 的服务体系, 该服务体系将同时支持在 .Net 和 J2EE 这两种体系平台上的开发和应用。

#### 3) 基于 .Net 的网络地理信息服务 MapPoint

微软的 MapPoint 可以提供以下的 LBS 服务: 基于地址、兴趣点、经纬度的位置服务, 位置相关背景(如地图和地址)服务, 路径选择服务, 邻近搜索服务, 距离计算服务等。Mapoint3.0 完全基于 VS.Net 开发, 任何网络用户都可以通过 SOAP 来存取 Mappoint 的 XML Web 服务接口。VS.NET 会自动为 MapPoint 服务产生代理类, 使得开发者可以非常方便地使用 MapPoint 的服务。

#### 4) 基于 .Net 平台的 Terra Server 影像数据服务

Microsoft 公司的 Terra server 是目前 Internet 上最大的地图服务器和卫星影像数据仓库。在微软的 .Net 平台推出后, Terra Server 也推出了基于 .Net 平台的 TerraServer .NET Web Service, 简称为 TerraService。它为用户提供了一个通过可编程的接口访问 Terra Server 影像数据的方式, 从而提供更灵活的访问形式。并且用户可以通过服务的接口来建立自己的应用系统。

### 2 Web 服务在 WebGIS 方面的国内研究和应用现状

在 WebGIS 应用系统的开发和使用方面, 目前能够提供网络地理信息服务的网站已经有数字北京(<http://www.digitalbeijing.gov.cn>), 图行天下(<http://www.go2map.com>), 数字福建(<http://www.szjf.gov.cn>)等。这些网站目前以提供电子地图服务为主, 对于高级的地理空间信息处理服务(如地理编码, 路径分析等)的功能还比较弱。

目前国内对 Web 服务的应用主要偏重于电子商务的应用方面, 基于 Web 服务的 WebGIS 系统的研究还处于起步阶段, 也没有出现比较成熟的基于 Web 服务的 WebGIS 应用系统和软件。



## 1.5 本文的主要工作和组织

本文从理论性和实用性两方面着手,通过研究现有 WebGIS 开发流行技术、所遇到的问题以及当前 .NET 技术中 Web 服务发展的新趋势,提出一种基于 .NET 的 WebGIS 体系结构,并将其应用到具体开发过程,设计出了基于 .Net 的电网配电线路和设备 WebGIS 管理系统,并详细阐述了其关键技术的实现方法、应用服务器端和客户端的设计与实现。

本文共分为六章,每一章节具体内容安排如下:

第 1 章主要研究了 WebGIS 的发展现状和存在问题,给出了 .NET 和 Web Service 的定义,总结了国内外在基于 .NET 的 Web Service 的 WebGIS 系统方面的研究进展。最后给出了论文主要工作和组织。

第 2 章主要研究 WebGIS 的基本特征、实现技术。然后在分析 WebGIS 发展历程和传统的 WebGIS 应用开发中存在的问题的基础上,指出基于 .NET 的 Web Services 开发模式对于解决传统 WebGIS 开发中遇到的问题有着重要的意义。

第 3 章主要研究 .NET 以及 Web Services 技术。研究了 .NET 的基本概念,详细分析了 .NET 的体系结构,并总结了 .NET 框架的新特性。分析了 Web Services 的定义特征、协议、体系结构及运行机制、开发工具。叙述了分布式计算的发展历程,对传统的分布式计算模型与 WEB 服务进行了比较,研究了 .NET 的开放标准 XML, SOAP, UDDI, WSDL, 最后详细阐述了 WebService 对 WebGIS 的作用和意义。

第 4 章主要研究了 OGC 制定的一系列 Web Service 相关的服务框架、协议栈和规范,这些规范主要用来解决网络环境下的分布式应用程序的交互问题,详细介绍了 WMS, WFS, WCS 等几个核心规范,阐述了 OGC 规范对实现 WebGIS 系统的作用和意义。

第 5 章以 Web 服务框架为核心,对系统开发环境及开发工具、系统地图功能服务、客户端和应用服务器端进行了设计,在此基础上,开发了一个基于 .NET 的电网配电线路和设备 WebGIS 管理系统,实现了本地应用和互联网上 GIS 服务应用的集成,检验系统的可行性,并对系统的优缺点进行了适当的分析。

第 6 章为总结与展望,首先对论文的内容进行总结,并对开发过程中遇到的问题进行了讨论。由于基于 .NET 的 WebGIS 技术的还未完全成熟,还有许多值得研究和探讨的问题,最后列举了一些需要进一步开展的工作。

## 第 2 章 WebGIS 技术

万维网地理信息系统 WebGIS 是 GIS 与 WWW 的有机结合, 是 GIS 在广域网环境下的一种应用, 最终目标是实现空间信息网络化<sup>[21]</sup>。GIS 通过 WWW 使功能得到了扩展(实际上是两者互为扩展), 真正成为了一种大众使用的信息工具。从 WWW 的任意一个节点, 人们可以浏览 WWW 上的各种分布式的、具有超媒体特性的地理空间数据及属性数据, 进行地理空间分析、查询, 以支持智能辅助决策<sup>[22]</sup>。

在 WebGIS 系统中引入 Web Services 技术, 必将改变 WebGIS 软件的体系结构, 从而改变 GIS 数据访问和功能互操作模式, 真正实现网络环境下空间信息共享和空间知识发现, 推进 WebGIS 的社会化应用<sup>[23]</sup>。

### 2.1 WebGIS 的基本特征

万维网地理信息系统是在 Internet 或 Intranet 网络环境下的一种兼容、存贮、处理、分析和显示与应用地理信息的计算机信息系统<sup>[24]</sup>。国际学术界称之为 WebGIS, 这主要是由于大多数的客户端应用采用了 WWW 协议。它的基本思想就是在互联网上提供地理信息, 让用户通过浏览器浏览和获得一个地理信息系统中的数据和功能服务。和传统的地理信息系统相比, WebGIS 具有以下几个特点<sup>[25]</sup>:

- 基于 Internet/Intranet 标准

WebGIS 采用 Internet/Intranet 标准, 以标准的 HTML 浏览器为客户端, 通过 TCP/IP 和 HTTP 协议, 可以访问任何地方的空间数据。

- 分布式体系结构

空间数据本身在空间上是分布的, WebGIS 采用分布式体系结构, 形成了客户端和服务端相互分离、协同工作的多层分布结构, 通过各种均衡策略有效平衡两者之间的负载。这种结构适应了空间数据分布的特征, 提高了网络计算资源和存储资源的利用率。

- 服务范围广

WebGIS 服务范围的广泛主要体系在两方面: WebGIS 可以通过网络为更为广阔范围内的用户提供空间信息服务; WebGIS 客户可以同时访问多个位于不同

地方服务器上的最新数据，而这一 Internet/Intranet 所特有的优势极大地方便了 GIS 的数据管理。

- 平台无关

一般来讲，WebGIS 的客户端采用的是通用浏览器，因此对客户端的软硬件没有特殊要求。在服务器端无论采用什么样的操作系统和 GIS 软件，由于通过网络将请求和处理结果发往客户端，WebGIS 服务器的处理方式对客户端而言是透明的，任一用户均可以通过通用浏览器访问任何得到许可的 WebGIS 服务器。这种特性使得远程异构数据共享成为可能，极大地提高了软硬件平台的独立性。

- 成本低廉、操作简单

在 WebGIS 的实现中，客户端往往只需要使用 Web 浏览器（有时可能会安装一些插件或处理图形数据），而数据和软件的管理与维护基本上由服务器完成，因此系统得成本比以往的全套专业 GIS 软件平台要少得多，客户端软件的简单性所节省的维护费用也是不容忽视的。

- 支持地理分布存储的多源数据

WebGIS 能充分利用已有的各种空间信息资源，支持地理上分布存储的多种来源和格式的空间数据，不仅有利于数据的维护和更新，而且有利于平衡系统负载，提高存取速度。

## 2.2 WebGIS 的实现技术及比较

WebGIS 的基本工作模式如下：用户启动客户端的交互程序(通过 WebGIS 浏览器)，由代理协议向服务器发送请求，并对服务器回送的数据进行显示、操作，使用其提供的 GIS 功能；如果用户操作的完成被定义在服务器一端，客户端交互程序还会将新的请求发回服务器，服务器则运行服务端的 GIS 应用，并再次将结果返回到客户端，由客户端交互程序负责解释并显示给用户。

目前，根据主要的图形属性数据所处的逻辑位置不同，构建 WebGIS 的主要技术可划分为基于服务器端的技术、基于客户端的技术和基于服务器/客户端的混合技术。

### 2.2.1 基于服务器端 WebGIS 的开发技术

基于服务器端技术的 WebGIS 由客户端浏览器向 Web 服务器发出服务请求，

服务器接受服务请求后调用有关的 GIS 服务程序，由服务器访问地理矢量数据、执行 GIS 功能，并将执行结果以静态 Web 页面的形式返回给客户端浏览器。在这种 WebGIS 中，GIS 数据和 GIS 处理功能都位于服务器端，客户端只负责向服务器发送请求和显示由服务器送回的相应结果。

基于服务器端技术的 WebGIS 开发的优点是服务器可以进行许多难于在客户端处理的复杂 GIS 操作，系统易于维护和更新。同时系统对客户端的要求较低，凡是可以浏览一般网页的客户均可以获得 GIS 信息，即使是非专业的普通用户也同样可以轻松的使用各种 GIS 功能。

基于服务器端技术的 WebGIS 开发的缺点是所有和 GIS 相关的操作都位于服务器端，因此客户端的每个请求都必须发送到服务器端处理，影响了响应的性能和速度，且系统和客户的交互性能差。

目前已经有多种不同的技术方法被应用于研制实现 WebGIS，基于服务器端的 WebGIS 开发技术主要有：CGI(Common Gateway Interface, 通用网关接口)、服务器端应用程序接口(Server API)等。

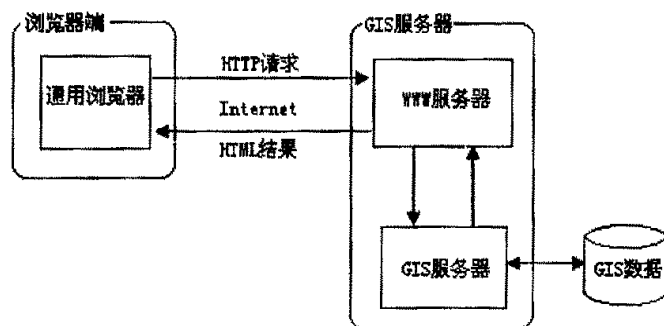


图 2-1 基于服务器模式的 WebGIS 体系结构

### 2.2.1.1 CGI (Common Gateway Interface) 技术

通用网关接口 CGI 就是互连网络服务器 (Web Server) 通过调用外部应用程序的接口扩展网络服务器的功能。客户端通过网络服务器激发 CGI 程序的响应实现具体的操作，读取超文本标识语言 (HTML) 文件，并将读取得数据信息或文件，通过服务器送往客户端<sup>[25]</sup>。

基于 CGI 的 WebGIS，在服务器端，GIS 软件(以应用服务器形式存在)通过 CGI 与万维网的 HTTP 服务器相连。在客户端，由万维网浏览器以 HTML 建立

用户界面，用户通过 HTML 表格和栅格图像输入指令，所有 GIS 操作和分析，都是在 GIS 服务器上完成的。服务器以图像和 HTML 文件的方式反馈<sup>[51]</sup>。

基于 CGI 的互联网地理信息系统的体系结构如图 2-2 所示。服务器端有两个服务进程 Web Server 和 GIS Server，这两者是通过 TCP/IP 协议进行通讯的，所以既可以运行在一台主机上也可以分别运行在两台计算机上。

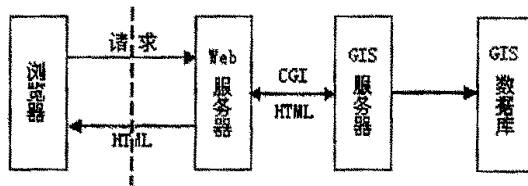


图 2-2 基于 CGI 的 WebGIS 体系结构

目前市场上推出的 WebGIS 系统软件，有一部分就是利用这一原理实现的，如 MapInfo 公司的 MapInfo MapXtreme。用这种技术方法构造 WebGIS 具有简单易行的特点(但可能具有带宽敏感问题，使系统的运行受到影响)，这种方法适用于对原有 GIS 系统的网络化改造<sup>[25]</sup>。

#### 2.2.1.2 服务器应用程序接口 (Server API) 方法

Server API (服务器应用程序接口) 技术是为克服 CGI 方法的低效率问题而研制的。这种方法的基本原理与 CGI 类似，所不同的是 CGI 程序是可以单独运行的程序，而基于服务器应用程序接口的程序必须在特定的服务器上运行<sup>[25]</sup>。如 ISAPI 依附于 IIS(Internet Information Server)，只能在 Windows 平台上运行，其可移植性较差。但是它的特点是速度要比 CGI 方法快得多，因为基于服务器 API 的动态连接模块启动后会一直处于运行状态，而不像 CGI 每次都要重新启动。目前，主要的 Server API 技术是 ASP 和 JSP/Servlet 技术。

##### 1) ASP 技术

ASP (Active Server Page) 是 IIS 的一个组件，利用它可以产生动态的、交互的、高性能的 Web 服务应用程序。ASP 属于 ActiveX 技术中的服务器端开发环境，与常见的在客户端实现动态主页的技术如 Java Applet, ActiveXControl, VB Script, JavaScript 等不同，ASP 中的命令和 Script 语句都是由服务器来解释执行

的, 执行结果产生动态生成的 Web 页面并送到浏览器端(用户可以在服务器端执行用 VBScript 或 JavaScript 编写的嵌入在 HTML 文件中的小程序, 而在以前这些小程序的都是客户端的浏览器上执行的); 而客户端技术的脚本语句则是由浏览器来解释执行的。由于 ASP 是在服务器端执行的, 开发者可以不必考虑与各种浏览器的兼容问题, 也可不必担心别人下载程序从而窃取编程逻辑。虽然在服务器上执行会增加服务器的负担, 但由于此时可以存取服务器的有关资源, 这些解释性语言编写的程序具有强大的交互能力[53]。

## 2) JSP 技术

Java Server Pages (JSP) 技术为我们提供了一种建立动态网页的简单方法, 而且也简化了构造 Web 程序的工作。作为 Java 技术的一部分, JSP 能够快速开发出基于 Web 的、独立于平台的应用程序。JSP 把用户界面从系统内容中分离开来, 使得设计人员能够在不改变底层动态内容的前提下改变整个网页布局。

简单的说, 一个 JSP 网页就是在 HTML 网页中包含了能够生成动态内容的可执行应用程序代码。此应用程序可能包含 JavaBean, JDBC 对象, Enterprise JavaBean (EJB) 和 Remote Method Invocation (RMI) 对象, 所有的部分都可以非常容易的从 JSP 网页上访问到。例如, 一个 JSP 网页可以包含 HTML 代码所显示的静态文本和图像, 也可以调用一个 JDBC 对象来访问数据库。当网页显示到用户界面上以后, 它将包含静态 HTML 内容和从数据库中找到相应的动态信息。

## 3) Servlet 技术

Servlet 意为服务器小程序, 与运行于客户端的 Applet(小程序)对应, 它们都是使用 Java 语言写的小型程序。Servlet 没有 main 方法, 只有一些特定的方法用于启动、执行和退出。它可以与运行于客户端的 Applet 进行交互, 也可以直接与客户端的 HTML 页交互。Servlet 是一个与协议无关、跨平台的服务器方构件, 如同 Applet 集成到 WWW 浏览器一样, 它被集成到服务器中, 可以实现网络上远程动态加载。

Servlet 与 CGI 的执行相比, 使用 Servlet 比使用 CGI 脚本来处理用户的请求要快。因为 Servlet 是在服务器上运行一个 Java 虚拟机, 因此在多次调用同一个 Servlet 时, 它只需加载一次。只有在 Servlet 发生了变化时, 才重新加载 Servlet。且加载一个修改的 Servlet 不需要重新启动服务器。Servlet 可以单独工作, 也可以把它们连接在一起, 可以让一个 Servlet 调用多个其他的 Servlet, 也可以让多

个 Servlet 按多个流水线方式工作<sup>[54]</sup>。

基于 Server API 技术的互联网地理信息系统体系结构如图 2-3 所示。

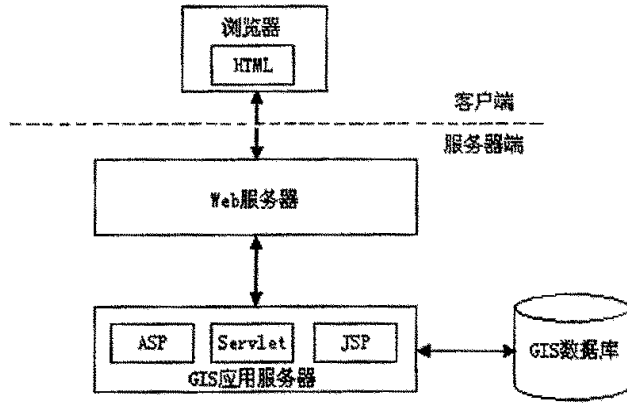


图 2-3 基于 Server API 技术的互联网地理信息系统体系结构

### 2.2.2 基于客户端 WebGIS 的开发技术

基于客户端技术的 WebGIS 允许 GIS 分析和 GIS 数据处理在客户端执行。这些 GIS 分析工具和 GIS 数据最初驻留在服务器上。用户通过浏览器向服务器发出需要 GIS 数据和 GIS 处理工具的请求，服务器将所需要的 GIS 数据和 GIS 处理工具传送给客户端，客户端接受所需要的 GIS 数据和 GIS 处理工具，按照用户的操作，进行 GIS 数据处理和分析；此时无须服务器的参与。由于所需要的 GIS 数据和 GIS 处理工具已经到达客户端，因而具有操作方便、灵活、速度快等优势<sup>[55]</sup>。

基于客户端的 WebGIS 开发技术的优点是增强了客户端处理能力，减少了服务器端处理的数据量和网络传输负荷，系统的交互性强。

基于客户端的 WebGIS 开发技术的缺点是对客户端计算机的性能要求高，系统难以维护，且用户通常需要进行必要的培训。

目前基于客户端的 WebGIS 开发技术主要包括 GIS Plug-in, GIS ActiveX 和 GIS Java Applet 等技术方法。

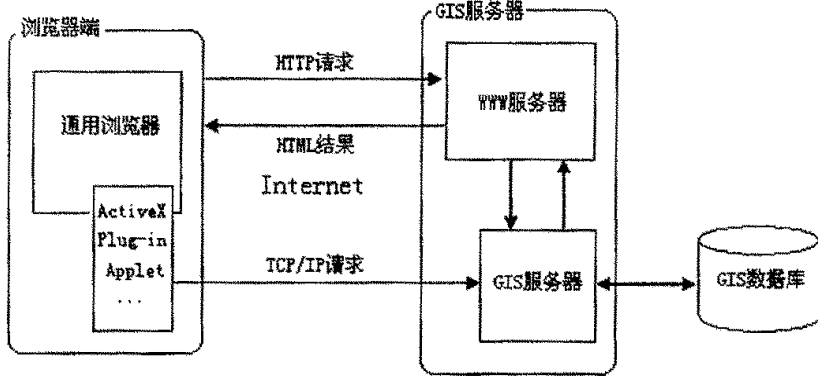


图 2-4 基于客户端的 WebGIS 体系结构

### 2.2.2.1 插件 (Plug-in) 技术

在普通的 Web 浏览器上安装能与网络浏览器交换信息并能执行的 GIS 软件，这种扩展 Web 浏览器功能的方法称为“插件技术”(Plug-in)。在 WebGIS 中，GIS Plug-in 可使 Web 浏览器支持特定格式的 GIS 数据处理，并为 Web 浏览器与 GIS 服务程序之间的通信提供条件，它能直接处理来自服务器的 GIS 矢量数据，并生成符合浏览器显示格式的数据，以供浏览器或其他 Plug-in 显示使用。

GIS 插件需要先安装才能使用，尽管它可以和浏览器一起来处理空间数据，但会增加客户端的负载。当客户端安装了许多插件后，如何管理这些插件就成为一个新的问题[25]。

目前，Autodesk 公司的 MapGuide 和 Intergraph 公司的 Geomedia Web Map 均采用了插件技术。

### 2.2.2.2 ActiveX 技术

ActiveX 是微软公司提出的一种建立在 OLE 标准之上的规范和公共框架。它同 Plug-in 一样，也用于扩展 Web 浏览器的功能。不过，Plug-in 技术与具体的浏览器有关，而 ActiveX 能使用在任何支持 OLE 标准的程序或应用系统中。ActiveX 由 HTML、Script 和 ActiveX 组件组成，其关键部分为 ActiveX 控件（也称为 ActiveX 组件）[25]。

ActiveX 控件目前只有 IE 全面支持，在 Netscape 中则必须有特制的 plug-in 才能运行，兼容性较差；并且只能运行与 MS-Windows 平台上；需要下载；占



有客户端机器的磁盘空间；由于可以进行磁盘操作，其安全性较差。其优点是：执行速度快：由于 ActiveX 可以用多种语言实现，这样就可以复用原有 GIS 软件的源代码，提高了软件开发效率。

### 2.2.2.3 Java Applet 技术

Java 技术主要是通过将 Java Applet 自动下载到客户端的浏览器上并利用 URL 对象来分布式地访问具有 URL 的数据对象。Java Applet 是一种运行在浏览器环境中的小程序，也可视为 Java 插件。Java Applet 通过 <applet> 标签被嵌入在 HTML 文件中，其执行代码同时被下载到浏览器上，并由浏览器负责解释执行。由于是自动进行的，只要服务器端对 Java Applet 做了更新，浏览器就会将最近版本的 Java Applet 文件下载到本地。

Java Applet 和 WebGIS 结合，形成了 GIS Java Applet，它是用 Java 开发的小应用程序，在程序运行时从服务器端自动下载到浏览器，与浏览器紧密结合，用以增强 Web 浏览器的空间信息处理功能<sup>[25]</sup>。不足之处是：对于叠置分析、资源分配及优化等空间分析功能的实现，GIS Java Applet 还比较薄弱；使用已有的 GIS 操作分析资源的能力弱，处理大型的 GIS 分析能力(空间分析等)的能力有限，无法与 CGI 模式相比；GIS 数据的保存、分析结果的存储和网络资源的使用能力受到限制<sup>[56]</sup>。

### 2.2.3 基于服务器/客户端的混合技术

前面所讲到的单纯的客户端模式和服务器模式都存在着明显的不足，可以将两种模式的优点结合构成一种混合模式是解决这些问题的理想方法<sup>[27]</sup>。

混合模式的特点在于<sup>[28]</sup>：当需要执行大数据的处理和分析时，可以在高性能的服务器上执行；当需要由用户来控制处理任务时，则可在客户端进行；在这种混合模式下，客户端和服务器共享彼此的功能，数据和应用程序可以根据需要由客户端来不断的请求，或者在客户端执行，或者在服务器端执行，从而使系统的执行效率达到最优化。在目前很多的主流 WebGIS 产品都提供了多种的技术方案供用户选择使用。用户可以根据需要灵活的配置服务器和客户端的功能，从而达到系统的最优化。

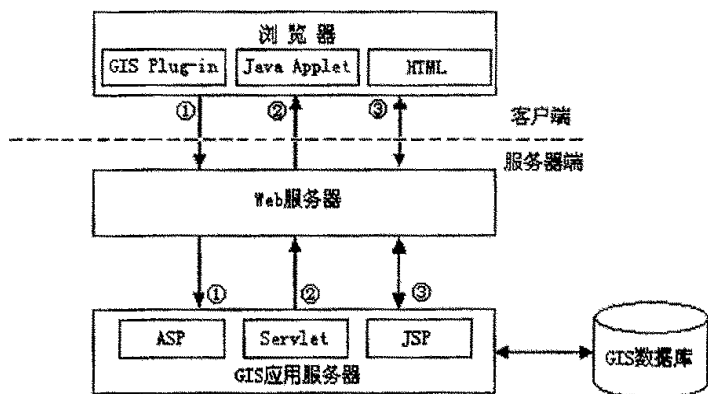


图 2-5 客户机—服务器混合模式体系

① 数据请求 ② 数据及小应用程序返回 ③ 持续的数据请求和传输

上面所述 5 种现有 Web GIS 的工作模式和主要运行环境等之间的比较如表 2-1 所示，用户可以根据不同的需要和网络技术条件来选择适合自己的模式。

表 2-1 Web GIS 常用技术比较

技术类型	优点	缺陷
CGI	客户端小，处理大型 GIS 操作分析的功能强，充分利用服务器现有资源。	网络传输和服务器的负担重，同步多请求问题，作为静态图像，JPEG 和 GIF 是客户端操作的唯一形式。
ServerAPI	不像 CGI 那样每次都要重新启动，其速度较 CGI 快得多。	需要依附于特定的 Web 服务器和计算机平台。
Plug-in	服务器和网络传输的负担轻，可直接操作 GIS 数据，速度快。	需要先下载安装到客户机上，与平台和操作系统相关，对于不同 GIS 数据类型，需要有相应的 GIS Plug-in 来支持。
ActiveX	执行速度快，具有动态可重用代码模块。	与操作系统相关，需要下载、安装，占用存储空间，安全性较差，对于不同的 GIS 数据类型，需要有相应的 GIS ActiveX 控件来支持。
Java Applet	与平台和操作系统无关，实时下载运行，无需预先安装，GIS 操作速度快，服务器和网络传输的负担轻。	GIS 数据的保存、分析结果的存储和网络资源的使用能力有限，处理较大的 GIS 分析任务的能力有限。

## 2.2.4 国内外主流的 Web 地图发布产品研究及比较 [1]

### 2.2.4.1 ESRI 公司的 Internet Map Server (ArcIMS) 平台

ESRI 于 1995 年开始研发用于 Internet 网上的 GIS 制图软件系统, 首先提供给用户的是两个产品平台上的 InternetGIS 制图服务软件: ArcView Internet Map Server 及 MapObjects Internet Map Server 扩展模块。2001 年 ESRI 推出了基于 Internet 和 Intranet 环境的新一代 GIS 应用和地图数据服务产品 ArcIMS。

ArcIMS 允许用户在 Internet 和 Intranet 环境下为浏览器端的客户提供 GIS 应用和地图数据。ArcIMS 的一个显著特点是在服务器和浏览器或其他客户端建立“数据流”。这一功能使得用户可以将本地数据与 Internet 上的数据结合起来, 从而为用户和数据发布者提供了一个更广阔的空间。其主要特点表现在以下几个方面:

- 简单的基于向导的界面。ArcIMS 在创建、设计、管理 GIS 站点上提出了一套简单易行的解决方案。它的便捷而强大的管理架构可以方便地帮助开发者完成创建地图服务、设置网页、发布和管理站点的工作。
- 强大的智能化客户端。ArcIMS 同时推出 HTML 和 Java 两种版本的强大客户端。ArcIMS Java 允许客户直接使用本地缓冲区中的数据, 客户端的操作不再必须由服务器来完成, 许多操作可以直接在客户端进行。
- 地图编辑和地图注释功能。ArcIMS 客户端支持动态编辑和标注地图信息, 允许用户将编辑结果返回给服务器, 地图注释功能还允许在地图上标注符合来进行提示。
- 方便的定制功能。ArcIMS 的客户端可以通过工业标准语言, 如 VBScript 和 JavaScript, 方便地定制以适用于不同的功能要求。
- 高质量的制图显示功能。由于 ArcIMS 支持矢量数据流, 可获得更清晰的显示效果。此外, 由于采用了新的影像边界平滑处理技术 (anti-aliasing), 提高了栅格数据的显示质量。
- 开放的可伸缩的结构。ArcIMS 不仅支持简单的 Intranet/Internet 应用, 而且支持一个或多个跨服务器的分布式 GIS 发布应用, 便于扩展原有的网上 GIS 应用, 无须重新构建。此外, ArcIMS 与其他流行的 Internet 技术兼容, 如 ColdFusion, Microsoft Transaction Server, Active Server Pages 等。

ArcIMS 的三层服务体系结构如图 2-6:

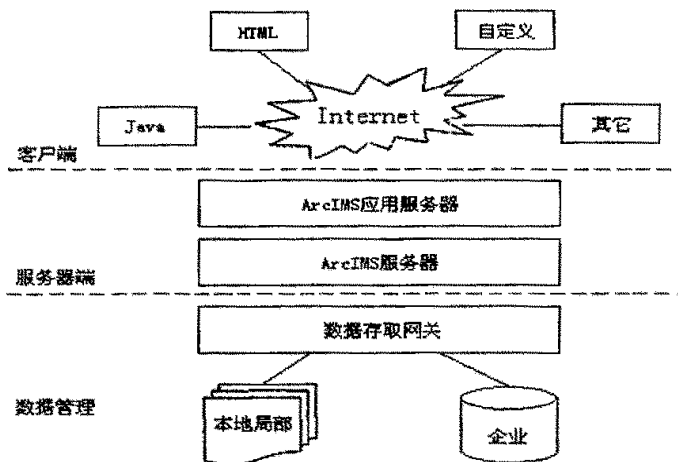


图 2-6 ArcIMS 的三层服务体系结构

#### 2.2.4.2 MapInfo 公司的 MapXtreme 平台

MapXtreme 是美国 MapInfo 公司开发的用于 Internet 或企业 Intranet/Extranet 的地图服务器产品。它采用标准的 TCP/IP 协议，通过 HTTP 进行文档和文件传输，在浏览器为标准的 HTML 语言，从而保证了与客户端浏览器的无关性。MapXtreme 在客户端提供了两种工作模式，一种是标准的 HTML 网页的模式，只要任何支持 HTML 的浏览器都可正常工作，例如 IE、Netscape、或 UNIX 平台的浏览器。推荐在 Internet 上采用这种工作模式。另一种是 Java Applet 插件，这种方式能够增强在浏览器端的交互性。但对网络速率要求较高，一般在 Intranet 上采用。在 ASP 环境下，MapXtreme 在服务器端的开发语言为 VBScript 或 JavaScript。在客户端可方便的扩展 HTML，Java 或 JavaScript 支持。2004 年 5 月 MapInfo 发布了新产品 MapXtreme 2004。MapXtreme 2004 是 MapInfo 为了支持 Microsoft 公司的 Windows.NET 框架，重新设计 MapX 和 MapXtreme for Windows 代码库体系结构的新产品。

MapXtreme 是采用瘦客户机模式三层体系结构的 WebGIS（客户端、服务器及数据管理层）如图 2-7 所示。

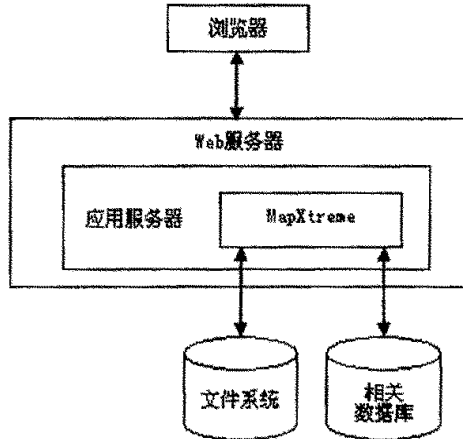


图 2-7 MapXtreme 的体系结构

MapXtreme 与 Web Server 的连接是通过应用服务器完成的，如 ASP。一个 Web Server 可以任意挂接多个 MapXtreme 地图应用服务器。MapXtreme 的 Server 可以自动维护和协调 Web Server 和多个 MapXtreme 之间的请求响应关系。MapXtreme 的技术特点主要表现在以下几个方面：

- 地图查询显示。用户可以在 Internet/IntranetWWW 上发布基于电子地图的应用系统。所以终端用户只需在本地机上安装浏览器（如 IE 或 Netscape）即可访问存放在服务器端的空间数据，可方便地对地图进行放大、缩小、漫游、查询、统计、图层控制、空间选择等操作。

- 稳定可靠的 GIS 高级管理。包括专题地图、缓冲区分析、地图目标查找、地图编辑、涂抹层、地理编码、扩展地图库、示例数据等。可直接读取 Lotus Notes、空间选择、访问各种数据源等。

- 支持分布式服务体系结构。当应用需求增加时只需增加 Server 来支持增多的用户数。完全与任何 Web Server 或 Web Browser 兼容。MapXtreme 的开放式结构可以在任何 Web Server 上工作，而且可以利用 ISAPI, NSAPI 以 CGI 各自的优势。而且，MapXtreme 不需要任何 plug-in，因此可以通过任何 PC 及 Unix 工作站的 Web Browser 访问。

- 以 MapX 为地图引擎。MapX 是一个 可编程的 OCX 控件，是可重复利用可编程对象，提供绝大部分 MapInfo Professional 支持的地图功能，可以利用编程平台所提供的数据库访问机制，也可利用自身提供的 ODBC 接口，并可进

行数据的智能绑定。

- 方便的开发方式。开发环境是由 Microsoft 公司提供的 Visual InterDev 开发工具。开发人员利用它可视化地创建并维护 HTML 文档，在 WWW 应用系统中集成高级应用逻辑，并管理整个 Web 应用开发过程。由 Visual InterDev 创建的 ASP 在 IIS 上运行时，能够自动为每一个客户端维持状态参数。其提供的开发语言 VB Script 早已为广大开发人员熟悉。

- MapXtreme 的优势。开发人员能集中地控制和维护地图和数据库数据，并集中实现应用程序功能，避免了以往系统维护、同步困难的问题，尤其适合信息量大，用户多的单位的实际情况。另外，由于使用 Web 浏览器作为客户端，更使开发人员可以将地图信息系统紧密地与其他系统结合，给用户提供一个综合信息系统。

#### 2.2.4.3 Autodesk 公司的 Autodesk MapGuide 平台

Autodesk MapGuide 包括下列四个软件产品：

- Autodesk MapGuide Viewer。这是安装在客户端的浏览器，可从 MapGuide 网站下载并安装到客户端的浏览器上，负责在浏览器中解释传输过来的矢量地图窗口文件。

- Autodesk MapGuide Author。这是在 Web 站点上创建、修改和发布基于矢量交互地图的工具。它提供“傻瓜式”面向对象的编程，用户通过设置对话框中的操作属性就可以完成所有制作。通过 Autodesk MapGuide Author 可以设置地图访问权限，增加安全措施。

- Autodesk MapGuide Server。这是运行于 Windows NT 上的与 Web 服务器进行交互信息的地图 CGI 服务器软件。它是 32 位多线程软件，可并发连接、访问位于站点上的多个地图文件和关系数据库。MapGuide 有自己的矢量地图格式，以 sdf 为后缀，其他格式如 mif、shp、csv、bna 等均需转换成这种格式。

- Cold Fusion。用于提供快速 Web 应用开发，包括用于向浏览器发布空间数据。

#### 2.2.4.4 Intergraph 公司的 GeoMedia Web Map 平台

GeoMedia Web Map 技术特点主要表现在：

- 直接访问多源数据。利用 Intergraph 独特的地理数据服务器和超图空间

数据仓库技术，能对多源数据进行无缝集成，可以访问并发布多种格式的 GIS 数据，无须转换。

- 动态发布 Active CGM 矢量图形，传送数据量小，传送速度快。允许使用栅格影像作背景，遵循 Web 工业开发标准，能将光栅图像、视频、音频与 GIS 较好地集成。
- 支持空间信息的实时发布和实时更新。
- 易于开发。使用标准的 Web 开发工具，如 Java Script、VB Script、Frontpage 和 ActiveX 等，GeoMedia Web Map 就可被用户化。
- 维护简单，客户端操作简便。由于 GeoMedia Web Map 直接发布数据库中随时更新的数据，发布时也不必转换为其他格式，所以对于所发布信息的更新和维护非常简单。在客户端，使用者只需浏览器即可完成对信息的浏览、查询和分析。

## 2.3 WebGIS 应用开发发展趋势

随着软件开发方法经历了结构化方法、面向对象方法到基于构件的软件开发的发展，从开发模式的角度来看，GIS 在其发展历史上，曾经经历了 GIS 功能包、集成式 GIS、模块化 GIS 和构件式 GIS 等阶段。ESRI 的 Arc/Info, Genasys 的 GenaMap 等为集成式 GIS 代表产品；模块化 GIS 代表软件有 Intergraph 的 MGE、中地公司的 MapGIS 等；分布式构件技术如 CORBA, DCOM、RMI 等。

随着网络和通信技术的发展，新的分布式计算要求软件实现跨空间、跨时间、跨设备、跨用户的共享，为适应软件的这种需求，面向服务软件体系结构产生了。面向服务软件体系结构(SOA)并不是一个全新的概念，而是面向对象技术和构件技术的合理化发展。但是它给 IT 业带来的好处和产生的影响远远超过了技术革新。与传统的软件开发模型相比，SOA 提供了一个更有力的开发和部署应用的软件开发模型。它带来了软件开发和使用方式的革新。一方面，软件和应用已经从以往要进行安装的程序，演变成了一种服务。另一方面，面向服务的软件开发，将应用分成多层，开发者可以专注于某一层的发展。同时可以将服务提供者提供的服务集成到自己的程序中，从而可以减少软件开发和部署的时间和投资。服务由服务的提供者进行维护和更新，使用者也可以自主选择提供者从而系统具有更好的灵活性、伸缩性、可用性和可维护性。

基于.NET的 Web Services 是面向服务的体系结构的一个实现,它通过将紧密耦合的、高效的 N 层计算技术与面向消息的、松散耦合的 Web 概念相结合,是一种新的分布式计算模式。Web Services 可以在更高的层次上解决构件式 GIS 开发中存在的问题,它通过 UDDI 在 Internet 上注册服务,采用基于 XML 的 WSDL 对所提供的服务进行详细描述,使用户可以方便地调用服务;其基于 Web 的网络运行环境,使服务不存在异构问题;SOAP 协议则很好地解决了远程对象调用的问题。从这一点来说,我们可以认为基于 Web Services 的 WebGIS 开发模式是软件体系结构发展的必须。

Web Services 由于借助 XML 和 SOAP,使得互操作和集成问题从层次上被简化。XML 提供了跨平台的数据编码和组织方法,而 SOAP 建立在 XML 之上,定义了一种跨平台的信息交换的简单包装方法。绑定于 HTTP 之上的 SOAP 协议,可以跨语言、跨操作系统进行远程过程调用(RPC),实现了编程语言和系统平台的无关性。如果我们将 Web Services 的设计思想和理念与 GIS 相结合,可以实现以下目标:

1) 使用一种通用的标准(XML)语言来描述数据,屏蔽数据格式的不同,实现空间地理信息的共享,而且独立于平台与开发语言。目前已开展了将 XML 应用到地理空间信息领域的研究,用来描述空间地理信息,即 GML<sup>[19]</sup>。

2) 使用一种标准的通信协议来实现应用系统的交互,屏蔽系统软、硬件的不同,实现不同 GIS 系统的互操作。

3) 通过 Web Services 提供的编程接口,给各应用系统提供无限的集成能力,使得空间地理系统间、空间地理信息系统与普通商务应用系统间可以无缝集成,实现空间地理信息应用的有效价值链。

综上所述,基于 Web Services 的 WebGIS 开发模式对于解决传统 GIS 开发中遇到的问题有着重要的意义。

## 2.4 本章小结

由于 Web 的种种限制和 GIS 本身的一些特点,无论是基于服务器的技术、基于客户端的技术,还是能较好的发挥两者的长处、弥补两者缺点的服务器/客户端混合技术,事实上,这些 WebGIS 实现技术都存在很大问题。首先,它们都不能实现异构空间数据互操作以及实现跨平台的问题。因而,也不是真正意义



上的开放的万维网地理信息系统。而且，由于 Web 本身将内容的表现和运行逻辑结合在一起，所以，对于一个 WebGIS 应用，从设计、开发、应用到维护，都很难以连贯、有效的方式注重软件的整个生命周期。然而，以上述方式和模型构建的 WebGIS 是紧密耦合的，对 Web 开发人员来说，系统的开发、调试、维护都很困难，要实现对系统的扩展更是困难重重。

上述的这些问题，有些是 GIS 软件特有的问题，有些是 Web 软件特有的问题。所以，必须从 GIS 和 Web 这两方面同时入手来解决 WebGIS 的这些问题。目前有两大类型的标准化组织在研究和解决这些问题：专注于 Web 方面的 W3C, UDDI 等组织和专注于地理信息共享和互操作方面的 ISO TC211, OGC 等组织。OGC 对于 Web Service 则更专注于它所表达的服务的这个概念，强调了在网络服务中的地理信息服务的地位和作用。也就是说，OGC 注重于 Web Service 的框架和概念层面。本文将在第 4 章详细的阐述 OGC 的 Web Service 的概念和体系。

由于客户端无需下载任何插件，地图应用易于扩充、方便快捷的开发方式等特性满足用户需求，本文第 5 章应用实践中系统采用的是基于服务器端 CGI 技术的 MapXtreme 平台，具体实现将在第 5 章详细阐述。

## 第 3 章 .Net 与 Web Service

当今的互联网在很大程度上是模仿过去的大型机模式。尽管带宽充裕，信息仍被锁定在中心数据库中，用户必须依靠 Web 服务器进行所有的操作，就像过去的分时共享模式那样。网站就像孤岛，用户无法以任何可行的方式进行通信。

所有人都相信 Web 会发展，然而如果想要使这种发展真正运用到开发人员、企业和用户身上，我们就需要高瞻远瞩。.NET 是 Microsoft 的用以创建 XML Web 服务（下一代软件）平台，该平台将信息、设备和人以一种统一的、个性化的方式联系起来。借助于 .NET 平台，可以创建和使用基于 XML 的应用程序、进程和 Web 站点以及服务，它们之间可以按设计、在任何平台或智能设备上共享和组合信息与功能，以向单位和个人提供定制好的解决方案。XML Web services 允许应用程序通过 Internet 进行通讯和共享数据，而不管所采用的是哪种操作系统、设备或编程语言。.NET 平台提供创建 XML Web services 并将这些服务集成在一起之所需。

### 3.1 新一代的互联网平台——.Net

.NET = 新平台 + 标准协议 + 统一开发工具，它代表了一个集合、一个环境、一个可以作为平台支持下一代 Internet 的可编程结构。在未来，我们可以在任何时间，任何地点，使用任何设备获取信息 (any time, any place and on any device.)。但是这当然不是 .NET 的全部，仅仅能够获取信息是不够的，通过 .NET，用户还将会获得由程序封装过的数据，也就是服务。可以认为，.NET 是一种分布式运算的框架，以 XML 为基础，以 Webservice 为核心，辅以其他各种技术实现，意在充分利用 Internet 上强大的计算资源和丰富的带宽资源，提高用户的工作效率<sup>[31]</sup>。在 .Net 平台上，客户端——服务器、服务器——服务器、客户端——客户端之间的信息的交换和传递都是以 XML 的形式来完成的，XML 构成了信息沟通的媒介，如图 3-1 所示<sup>[30]</sup>。

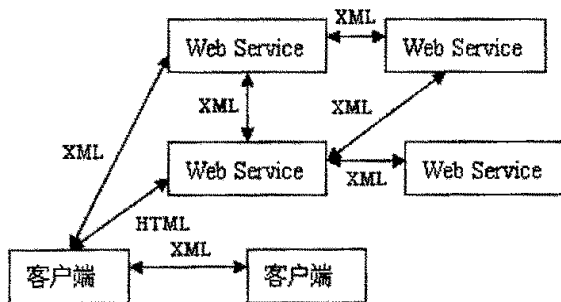


图 3-1 .Net 平台下的信息交换与传输

.NET 的网络服务应用模型体系结构如图 3-2 所示<sup>[26]</sup>。

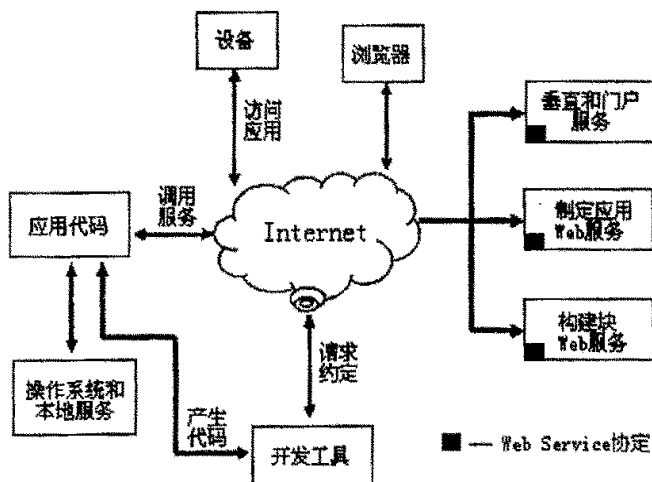


图 3-2 .NET 的网络服务应用模型体系结构

与其他平台相比，.NET 技术平台具有下述特点<sup>[25]</sup>：

1) 支持多种语言编程，可重用性好。用一种语言开发的组件可通过继承被其他组件重用。目前，.NET 支持二十多种语言。

2) 跨平台性能好。在不同的操作系统平台上运行 .NET 程序的过程是，先将用某种语言编写的程序编译成中间语言，执行时用即时 (Just In Time) 编译器将中间语言编译成本地操作系统平台所支持的代码，从而实现异构平台下对象的互操作。

3) 安全性高。.NET 通过通用语言运行环境 (Common Language Runtime)

来确保资源对象及类型安全。

4) 远程交互能力强。在 HTTP、XML、SOAP (Simple Object Access Protocol, 简单对象访问协议)、WSDL (Web Service Description Language, Web 服务描述语言) 等标准和规范的支持下, .NET 提供了在异构环境下连接远程设备、获取远程服务、交互远程应用的编程界面。

### 3.1.1 .Net 平台的基本组成

.Net 是一种将信息、用户、系统和设备连接在一起的软件技术。它提供了前所未有的集成技术, 通过 XML Web Services 将许多大大小小的、分散的应用通过互联网连接在一起。

微软的 .Net 平台是由下面的一些基本元素所构成的<sup>[27]</sup>:

1) 开发工具: .NET 框架和 Visual Studio.NET, 这些是开发人员用来生成 XML Web Service 的工具。.NET 框架是 .NET 平台核心中的一套编程接口; Visual Studio.NET 是一套多语言系列的编程工具, 可使开发者快速开发出符合 .NET 体系的应用软件服务, 并使各环节的数据传送更加容易。

2) 服务器: .NET 的服务器是一系列用于生成、发布和操作 XML WebService 的基础程序, 包括 Windows 和各种 .NET 企业服务器等等。这些服务器包括有:

- Application Center 2000, 用于 scale-out solutions;
- BizTalk Server 2000, 用于创建和管理基于 XML 的跨程序和服务的商务流程;
- Host Integration Server 2000, 用来访问主机上的数据和应用程序;
- Mobile Information 2001 Server, 使移动设备, 比如移动电话, 也能使用这些应用程序;
- SQL Server 2000 储存和检索结构化的 XML 数据;

3) 智能设备 (smart device): .NET 利用智能的客户端应用程序和操作系统使各种的智能设备, 诸如 PC、手提电脑、PDA 和其它移动设备等都能够通过 XML Web Services 在任何地点、任何时间获得信息。

4) 模块构建服务 XML Web Services: 是微软 .Net 平台的核心技术, XML Web Service 是在 Internet 上进行分布式计算的基本构造块。开放的标准以及对用户和应用程序之间的通信和协作的关注产生了这样一种环境, 在这种环境下, XML WebService 成为应用程序集成的平台。应用程序是通过使用多个不同来源

的 XMLWeb Service 构造而成的，这些服务相互协同工作，而不管它们位于何处或者如何实现。

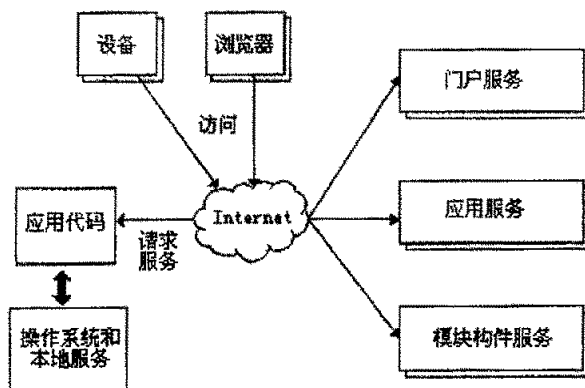


图 3-3 .Net 战略的总体描述

### 3.1.2 .Net 框架技术架构 (.Net Framework)

.Net 框架 (.Net Framework) 是用于构建和运行下一代软件应用程序和 XML Web 服务的 windows 组件。目前，它支持 20 多种不同的编程语言。.Net 框架包括几个主要组成部分，首先是整个开发框架的基础，即通用语言运行时 (Common Language Runtime, CLR) 以及它所提供的一组基础类库；在开发技术方面，.Net 提供了全新的数据库访问技术 ADO.NET、Web 应用开发技术 ASP.NET 和 Windows 编程技术 Windows Forms；在开发语言方面，.Net 提供了 VB、C++、C# 等多种语言支持；而 Visual Studio.NET 则是全面支持 .Net 的开发工具<sup>[32]</sup>。其体系结构如图 3-4：

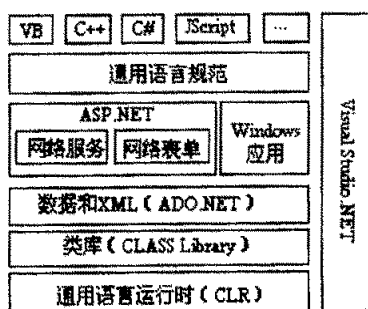


图 3-4 .Net 框架体系结构

- 通用语言运行时 (Common Language Runtime, CLR)

通用语言运行时处于 .NET 框架的最底层, 是这个框架的基础。它为各种语言提供了一个统一的运行环境; 另外, 它还提供了更多的功能和特性, 比如统一和简化的编程模型, 用户不必迷惑于 Win32API 和 COM; 避免了 DLL 的版本和更新问题 (常称为 DLL Hell), 从而大大简化了应用程序的发布和升级; 多种语言的交互, 例如开发人员可以在 VB 中使用 C++ 编写的类; 自动的内存和资源管理等。

- .NET 类库

.NET 框架提供了一个包括很多高度可重用的接口、类型的类库, 该类库是一个完全面向对象的类库, 为应用程序提供各种高级的组件和服务。下面是对这些组件和服务的一个概括:

- 1) 系统框架服务

系统框架服务包括一套在标准语言库中使用的基本类库, 例如: 集合、输入/输出、字符串及数据类。另外, 它还提供了访问操作系统和其他服务的类, 如网络、线程、全球化和加密的类。服务框架也包括数据访问类库及开发工具, 如调试和剖析服务能够使用的类。

- 2) ADO.NET 组件

为了提供对数据的访问, 服务框架包括 ADO.NET 类库。如同名字所暗示的那样, ADO.NET 由 ADO 发展而来。ADO.NET 为基于网络的应用程序和服务提供数据访问服务。下图 3-5 阐明了 ADO.NET 的体系结构, 表明任何数据, 不论这些数据实际上如何存储的, 都以 XML 或相关数据的格式被操作。

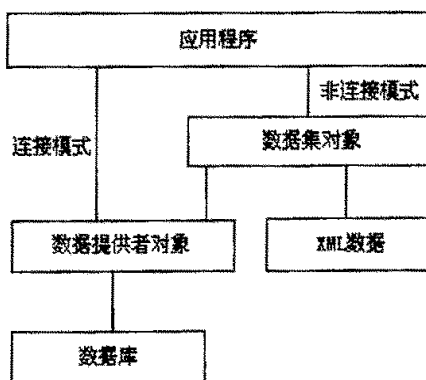


图 3-5 ADO.NET 体系结构

### 3) XML 数据组件

开发人员可以通过 XML 对应用程序中所使用的数据进行转换、传输和确认。NET 系统框架对 XML 数据提供了操作支持,也支持 ADO.NET 数据和 XML 数据之间的转换。

### 4) Windows 表单应用模板

从概念上讲,在服务框架的最上面是两个应用程序模板:Windows 表单应用模板和网络应用程序模板。编写 Windows 客户应用程序的开发人员可使用 Windows 表单应用程序模板以利用 Windows 丰富的用户接口特点,包括传统的 ActiveX 控件和 WindowsXP 的界面,如透明的、分层的浮动窗口等。开发人员会发现 Windows 表单可编程模板和对设计阶段的支持非常直观。

### 5) ASP.NET 应用服务

ASP.NET 的核心是高性能的用于处理基于 HTTP 请求的运行语言,编译运行方式大大提高了它的性能。ASP.NET 使用基于构件的 .Net 框架配制模板,因此它也具备 XCOPY 配制、构件并行配制、基于 XML 配制等 .NET 框架的优点。它支持应用程序的实时更新,提供高速缓存服务。

### 6) ASP.NET Web 表单

ASP.NET 网络表单支持传统的将 HTML 内容与脚本代码混合的 ASP 语法,但它提出了一种将应用程序代码和用户接口内容分离的更加结构化的方法 (Code Behind 方式)。ASP.NET 提供了一套映射传统的 HTML 用户接口部件(如列表框、文本框、按钮)的 ASP.NET Web 表单控件和一套更加复杂强大的网络应用控件。

## 3.2 .Net 的核心技术——Web Service

.NET 技术是围绕 Web Service 展开的。Web Service 是指由企业发布的完成其特别商务需求的在线应用服务,其他公司、合作伙伴的应用软件能够通过 Internet 来动态访问并使用这些在线服务。Web Service 是独立的、模块化的应用,能够通过因特网来描述、发布、定位以及调用。可将 Web Service 视作 Web 上的可编程组件。

### 3.2.1 Web Service 的定义和特征

那么到底什么是 Web 服务? Web service 平台是一套标准,它定义了应用程

序如何在 Web 上实现互操作性。你可以用任何你喜欢的语言，在任何你喜欢的平台上写 Web service，只要我们可以通过 Web service 标准对这些服务进行查询和访问。从表面上看，Web service 就是一个应用程序，它向外界暴露出一个能够通过 Web 进行调用的 API。就是说能够用编程的方法通过 Web 来调用这个应用程序。更精确的解释是：Web service 是建立可互操作的分布式应用程序的新平台。

从外部的使用者角度而言，Web Service 具有如下的一些特征<sup>[33]</sup>：

1) 完好的封装性。Web Service 是一种部署在 Web 上的对象，所以它具备对象的良好封装性，对于使用者而言，他能且仅能看到该对象提供的功能列表。

2) 松散耦合。这一特征也是源于对象/组件技术，当一个 Web Service 的实现发生变更的时候，调用者是不会感到这一点的，对于调用者来说，只要 Web Service 的调用界面不变，Web Service 的任何变更对他们来说都是透明的，甚至是当 Web Service 的实现平台从 J2EE 迁移到了 .NET 或者是相反的迁移流程，用户都可以对此一无所知。

3) 使用标准协议规范。作为 Web Service，其所有公共的协约完全使用开放的标准协议进行描述、传输和交换。这些标准协议具有完全免费的规范，以便由任意方进行实现。一般而言，绝大多数规范将最终由 W3C 或 OASIS 作为最终版本的发布方和维护方制定。

4) 高度可集成能力。由于 Web Service 采取简单的、易理解的标准 Web 协议作为组件界面描述和协同描述规范，完全屏蔽了不同软件平台的差异，无论是 CORBA、DCOM 还是 EJB 都可以通过这一种标准的协议(SOAP)进行互操作，实现了在当前环境下最高的可集成性。

5) 通用数据格式。通过用现有的开放式标准而不是专用的封闭通讯方法，任何支持同样的开放式标准的系统都能够理解 Web Services。利用自我描述的基于文本的消息使自治和完全不同的系统之间能够进行通讯。Web Services 使用 XML 实现此功能。

### 3.2.2 Web Service 的工作流程

Web Service 的使用过程如下：

1) 客户端向提供 UDDI 服务的服务中心请求查找服务，得到 WSDL 文档的 URL 地址。客户端如果知道 WSDL 文件的 URL 地址，可以跳过这一步。



- 2) 客户端根据 URL 向 Web Service 的提供者请求服务说明文件 WSDL。
- 3) 客户端根据服务说明，向 Web Service 的提供者请求这个服务，并得到服务的结果。

通过这一系列的交互，客户端完成了对服务的使用。

Web Service 的工作流程如下图 3-6 所示。

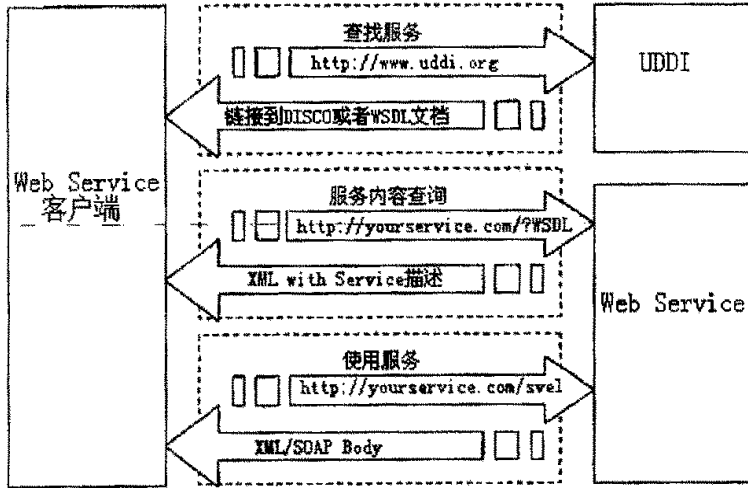


图 3-6 Web Service 的工作流程

### 3.2.3 Web Service 的体系架构

对于 Web 服务而言，其标准的体系构架如图 3-7 所示。

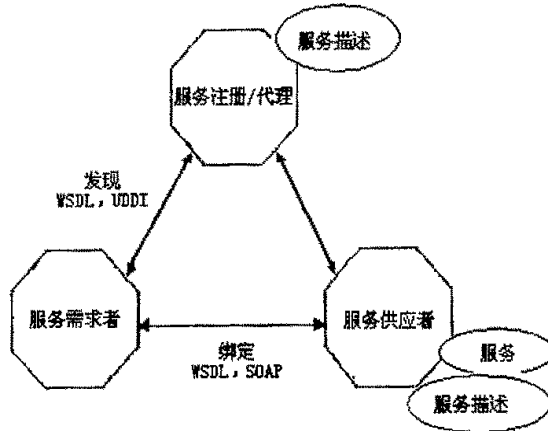


图 3-7 Web Service 的体系架构

### 1. Web 服务的组成角色:

1) 服务供应者: 从商业角度, 是服务的所有者; 从体系结构角度, 是拥有服务访问权的宿主平台。从技术角度, 主要功能是提供服务, 并维护注册表以使服务可用。

2) 服务请求者: 从商业角度看, 是需要某些功能满足某种需求的应用。从体系结构角度, 是查找并调用或初始化某项服务交互的应用。服务请求者的角色可以是人们使用的浏览器或者没有用户界面的一个程序(如另一个 Web 服务)。从技术角度, 主要功能是通过服务中介发现 Web 服务, 然后调用这些服务创建应用。

3) 服务注册库: 是可搜索的服务供应者发布的服务描述的注册库。服务请求找到服务并获得服务描述中的绑定信息, 包括服务开发过程中的静态绑定或服务执行过程中的动态绑定。服务注册库实际上是服务的注册交换中心, 扮演着服务供应者和服务请求者之间撮合匹配的服务中介角色。

### 2. Web 服务的交互操作

一个应用在使用 Web 服务时, 单次或重复发生三类基本操作:

1) 发布操作: 服务只有在发布了服务描述之后, 才能被服务请求者发现和调用。服务发布到哪里取决于应用需求。

2) 查找操作: 服务请求者直接或查询服务注册库获得所需服务的描述。查找操作在服务请求者的两个不同的时间段是不同的。在设计阶段为程序开发获取服务的接口描述, 在运行阶段为服务调用获取绑定和定位信息。

3) 绑定操作: 在绑定操作中, 服务请求者在运行阶段使用服务描述中的绑定细节来定位、访问和调用目标服务, 以调用或初始化一次服务交互。

### 3. Web 服务的内涵

1) 服务: Web 服务是由服务描述定义的接口。服务是部署在服务供应商所提供的网络可访问的平台上的软件模块, 是 Web 服务的具体实现。服务是为被调用或与服务请求者交互而存在的, 同时可作为服务请求者在其实现中使用其它的服务。

2) 服务描述: 说明服务接口和实现的细节, 包括数据类型、操作、绑定信息和网络位置, 也可以包括分类和其它元数据以方便服务请求者发现和使用服务。服务描述可以发布到服务请求者或服务注册者。

### 4. Web 服务的基本技术标准

Web 服务的三个基本操作包含了三个标准技术：发布操作使用“通用描述、发现和集成 (UDDI)；查找操作使用 UDDI 和 Web 服务描述语言 (WSDL) 的组合；绑定操作处理 WSDL 和简单对象访问协议 (SOAP)。从最基础的层次上看，绑定操作是三者中最重要。它包含服务的实际使用，这也是发生大多数互操作性问题的地方。简单地说，是服务提供者和服务请求者对 SOAP 规范的全力支持解决了这些问题，并实现了无缝互操作性。

### 3.2.4 Web Service 的关键技术

Web Service 是一种可编程的应用程序。它使用一系列标准协议来对各种请求做出响应。它对 UDDI 请求的响应是告诉请求者它可以提供什么服务。该服务由它所处理的消息的顺序和格式定义；这些消息的发送和接收都使用简单对象访问协议 (SOAP)，而此协议建立在 XML, HTTP 和 SMTP 的基础之上<sup>[94]</sup>。

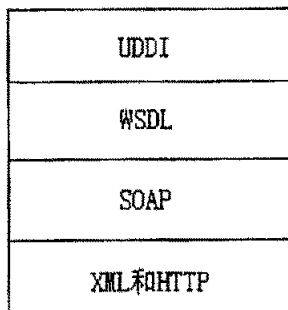


图 3-8 Web Service 的关键技术

#### 3.2.4.1 XML 可扩展标记语言

XML 全称是“可扩展标识语言” (eXtensible Markup Language, 简称 XML)<sup>[94]</sup>，同 HTML 一样，它也是网络上的通用语言。但 XML 并不是一个独立的，预定义的标识语言，它是一种元语言，是用来描述其它语言的语言。它允许用户定义自己的标识。XML 是 Web 服务的关键技术，SOAP、WSDL 和 UDDI 都是基于 XML 的。SOAP 使用 XML 作为其数据编码的格式。WSDL 使用 XML Schema 来描述 Web 服务的结构。UDDI 使用 XML Schema 定义注册表的结构，使用 SOAP 指定与注册表的通信。

XML 主要有三个要素：XML Schema, XSL (Extensible Stylesheet Language,

可扩展样式语言)和 XLL (Extensible Link Language, 可扩展链接语言)<sup>[36]</sup>。XML Schema 规定了 XML 文件的逻辑结构, 定义了 XML 文件中的元素、元素的属性以及元素和元素之间的关系, 它可以帮助 XML 分析程序校验 XML 文档是否有效和具有良好的格式。XSL<sup>[36]</sup>是用于规定 XML 文档样式的语言, 它能在客户端使 Web 浏览器改变文档的表示方法, 从而不需要再与服务器进行通信。XML 将进一步扩展目前 Web 上已有的简单链接。

XML 的特点如下<sup>[37]</sup>:

- 1) 可扩展性: 文档通过 DTD 或 XML Schema 来定义文档结构, 使其他信息系统自动了解文档的内容。
- 2) 结构性: HTML 不支持深层的结构描述; 而 XML 的文件结构嵌套可以复杂到任意程度, 能表示面向对象的等级层次。
- 3) 可校验性: 用户可以通过 DTD 或 XML Schema 来校验 XML 文档的格式是否满足 DTD 或 XML Schema 的约束。
- 4) 自描述性: 这个特性使差异性可以存在, 使计算机可以在没有人为干涉的情况下, 理解数据的含义。
- 5) 多样的样式表支持<sup>[38]</sup>: XML 把数据内容与他们的表现形式分开。这样既可以只关心数据的逻辑结构, 也可以通过样式表来格式化数据的表现。

#### 3.2.4.2 SOAP 简单对象访问协议

简单对象访问协议 SOAP (Simple Object Access Protocol) 是一种基于 XML 的、用于在松散的分布式环境中用于点对点之间交换结构化和类型信息的简单的轻量协议<sup>[41]</sup>。SOAP 是计算机之间交换信息的一个通信协议, 它与计算机的操作系统或编程环境无关。在 SOAP 中, XML 用于消息的格式化, HTTP 和其它的 Internet 协议用于消息的传送。

SOAP 为信息交换定义了一个消息协议。SOAP 的一部分说明了使用 XML 来描述数据的一些格式。SOAP 的另外一部分定义了一个可扩展的消息格式, 用于方便地使用 SOAP 消息格式描述远端程序 (RPC), 并且和 HTTP 协议进行捆绑 (SOAP 消息可以通过其他协议交换, 但是目前的说明仅仅定义了和 HTTP 协议捆绑的内容)。SOAP 已经成为 W3C 推荐的 Web Service 间交换的标准消息格式。

SOAP 消息是使用 SOAP 进行 Web 服务调用的基本消息载体, 由一个强制的 (就是必须出现的) SOAP envelope, 一个可选的 SOAP header 和一个强制的 SOAP

body 组成的 XML 文档<sup>[19,23]</sup>。SOAP 消息的结构如图 3-9 所示。

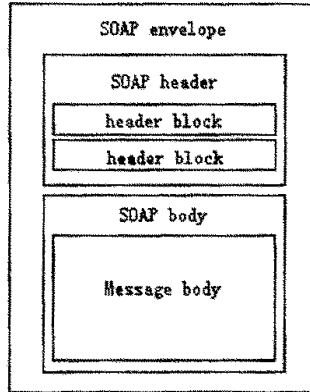


图 3-9 SOAP 消息的结构

SOAP envelope: Envelope 是表示该消息的 XML 文档的顶级元素。

SOAP header: 它是能够被 SOAP 消息传输路径中的任意的 SOAP 接受者结点处理的一组 SOAP 条目。

SOAP body: 它是能够被 SOAP 消息路径中的最终 SOAP 接受结点处理的一组 SOAP 条目。

SOAP 有以下的几个特点:

1) SOAP 是简单的, 客户端发送一个请求, 调用相应的对象, 然后服务器返回结果。这些消息是 XML 格式的, 并且封装成符合 HTTP 协议的消息。因此, 它符合任何路由器、防火墙或代理服务器的要求。

2) SOAP 不需要任何对象模型, 也不需要通过其他的通讯实体来使用对象模型。在避免对象模型的基础上, SOAP 将大部分对象功能(如初始化代码和垃圾堆积)留给客户端和服务端工作的底层, 同时其他功能(如信号编辑)则可以留给 SOAP 综合已有的应用程序和底层结构来完成。

3) SOAP 可以使用任何语言来完成, 只要客户端发送正确 SOAP 请求(也就是说, 传递一个合适的参数给一个实际的远端服务器)。SOAP 没有对象模型, 应用程序可以捆绑在任何对象模型中。

4) 在 SOAP 中, 双方使用 SOAP 消息来实现请求/响应通信。SOAP 消息是一种从一个发送者到一个接收者的单向传输, 所有消息都是 XML 文档, 它们具有自己的模式, 也包括对于所有元素和属性的正确的命名空间。

### 3.2.4.3 WSDL 网络服务描述语言

WSDL 是一种 XML Application，它将 Web 服务描述定义为一组服务访问点，客户端可以通过这些服务访问点对包含面向文档信息和面向过程调用的服务进行访问（类似远程过程调用）<sup>[42]</sup>。WSDL 首先对访问的操作和访问时使用的请求/响应消息进行抽象描述，然后将其绑定到具体的传输协议和消息格式上，最后定义具体部署的服务访问点。相关的具体部署的服务访问点通过组合就成为抽象的 Web 服务。

在 WSDL 中，由于服务访问点和消息的抽象定义已从具体的服务部署或数据格式绑定中分离出来，因此可以对抽象定义进行再次使用。其中消息 (Message) 是指对交换数据的抽象描述，而端口类型 (PortType) 指操作 (Operation) 的抽象集合。用于特定端口类型的具体协议和数据格式规范构成了可以再次使用的绑定 (Binding)。将 Web 访问地址与可再次使用的绑定相关联，可以定义一个端口 (Port)，而端口的集合则定义为服务 (Service)。WSDL 文档的结构组织如图 3-10 所示<sup>[47-49]</sup>。

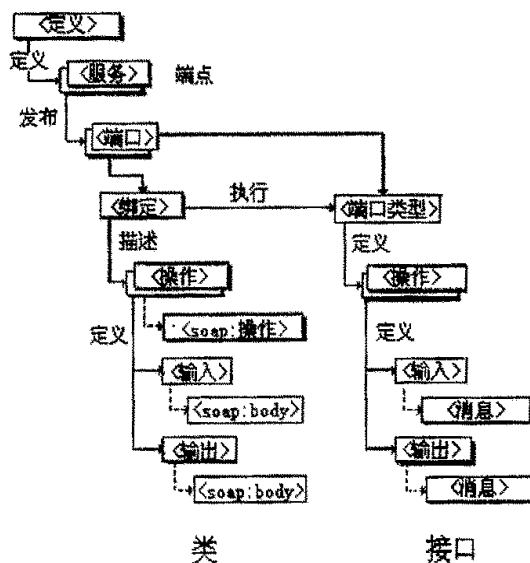


图 3-10 WSDL 文档的元素结构图

一个 WSDL 文档是由上述定义集组成，这些定义集定义了 WSDL 语言的核心要素，具体结构如下：

```

<wsdl:definitions name=" nmtoken" ? targetNamespace=" uri" ?>
  <import namespace=" uri" location=" uri" /*>
  <wsdl:documentation?/> ?
  <wsdl:types> ?
    <wsdl:documentation?/>?
    <xsd:schema?/>*
    <!--extensibility element-->*
  </wsdl:types>
  <wsdl:message name=" nmtoken">*
    <wsdl:documentation?/>?
    <part name=" nmtoken" element=" qname" ? type=" qname" ?/>*
  </wsdl:message>
  <wsdl:portType name=" nmtoken" >*
    <wsdl:documentation?/>?
    <wsdl: operation name=" nmtoken" >*
      <wsdl:documentation?/> ?
      <wsdl:input name=" nmtoken" ? message=" qname" >?
        <wsdl:documentation?/> ?
      </wsdl:input>
      <wsdl:output name=" nmtoken" ? message=" qname" >?
        <wsdl:documentation?/> ?
      </wsdl:output>
      <wsdl:fault name=" nmtoken" message=" qname" >*
        <wsdl:documentation?/> ?
      </wsdl:fault>
    </wsdl:operation>
  </wsdl:portType>
  <wsdl:binding name=" nmtoken" type=" qname" >*
    <wsdl:documentation?/>?
    <!--extensibility element-->*
    <wsdl:operation name=" nmtoken" >*

```

```

<wsdl:documentation?/> ?
  <!--extensibility element-->*
  <wsdl:input name=" nmtoken" ?> ?
    <wsdl:documentation?/> ?
    <!--extensibility element-->
  </wsdl:input>
  <wsdl:output name=" nmtoken" ?> ?
    <wsdl:documentation?/> ?
    <!--extensibility element-->*
  </wsdl:output>
  <wsdl:fault name=" nmtoken" >*
    <wsdl:documentation?/> ?
    <!--extensibility element-->*
  </wsdl:fault>
</wsdl:operation>
</wsdl:binding>
<wsdl:service name=" nmtoken" >*
  <wsdl:documentation?/?>?
  <wsdl:port, name=" nmtoken" binding=" qname" >*
    <wsdl:documentation?/> ?
    <!--extensibility element-->
  </wsdl:port>
  <!--extensibility element-->
</wsdl:service>
<!--extensibility element-->*
</wsdl:definitions>

```

“?” “\*” “+” 分别表示指定的元素或属性能够出现 0 次或 1 次，0 次或更多次，1 次或更多次。

WSDL 可指定一个可选的名字属性，即目标名字空间(TargetNamespace)，当采用此属性时，类型属性 URI 需要指定，但必须是一个相关的 URI。WSDL 允许采用一个文档定位输入语句指定一个名字空间(namespace)：



```
<definitions...>  
    <import namespace=" uri" location=" uri" /> *  
</definitions>
```

### 3.2.4.4 UDDI 统一描述、发现和集成协议

UDDI (Universal Description, Discovery and Integration) 规范定义一个发布和发现有关 Web services 的信息的标准方式<sup>[44-45]</sup>。UDDI 业务注册表使业务能够以编程方式定位有关其他单位公开的 Web services 的信息。

UDDI 是一套基于 Web 的、分布式的、为 Web 服务提供信息注册中心的标准规范，同时也包含一组使企业能将自身提供的、Web 服务注册以使得别的企业能够发现的访问协议的实现标准。Web 服务是下一代的 WWW，它允许在 Web 站点上放置可编程的元素，使得能进行基于 Web 的分布式计算和处理。UDDI 注册中心的创建目的就是为促进企业的 Web 服务的发展及为企业发现适当的 Web 服务。

#### 1) UDDI 基本概念

- UDDI 规范: UDDI 规范 V1 版包括两个规范文本, UDDI Programmer' s API V1.0 (UDDI 程序员 API 规范 1.0) 和 UDDI Data Structure Reference V1.0 (UDDI 数据结构参考 1.0)。前者定义了 UDDI Operator Site 能够支持的 API 接口, 而后者则描述了在 API 中具体 XML 描述的数据结构的具体定义。UDDI 规范是 UDDI Operator Site 的实现蓝本, 也是需要访问 UDDI Registry 的 Web 服务的参考规范。

- UDDI Registry (UDDI 注册中心): UDDI Registry 是所有提供公共 UDDI 注册服务的站点的通称。UDDI Registry 是一个逻辑上的统一体, 在物理上则是以分布式系统的架构实施的, 而不同站点之间是采用 P2P(对等网络)架构实施的, 因此访问其中任意一个站点就基本等于访问了 IIDDIT Registry。

- UDDI Operator Site (UDDI 注册中心操作入口站点, 简称 UDDI 操作入口): UDDI Operator Site 是 UDDI Registry 中每一个对等结点, 对于 UDDI Operator Site 的查询所获得的结果是覆盖全 UDDI Registry 中的信息的, 信息查询无需身份认证; 而在 UDDI Operator Site 上进行信息发布则必须使用该 UDDI Operator Site 自身的用户方能实施, 同时以后的更新、删除都必须通过这个 Operator Site, 并使用初始发布时使用的用户进行权限认证。

- Compatible UDDI Registry(兼容的 UDDI 注册中心): 所有兼容 UDDI 规范但并非属于提供公共服务的 UDDI Registry 的个别 UDDI 注册中心, 都称为兼容的 UDDI 注册中心。

2) UDDI——技术发现层

统一描述、发现和集成协议(UDDI)规范一个由 Web 服务所构成的逻辑上的云状服务, 同时也定义了一种编程接口, 这种编程接口提供了描述 Web 服务的简单框架。规范包括几份相关的文档和一份 XML Schema, 用来定义基于 SOAP 的注册和发现 Web 服务的协议。这些规范由来自多家业界主要公司的技术人员和管理人员花费了几个月的时间制定完成。这些公司也担负起实现第一批 UDDI 商业注册中心服务的任务, 这些服务将可以被所有人所访问, 同时其多个合作站点之间能够无缝地共享注册信息。

下图 3-11 描述了 UDDI 规范、XML Schema 和 UDDI 商业注册中心集群之间的关系, UDDI 商业注册中心集群能为 Web 服务提供“一次注册, 到处发布”的功能。UDDI 规范和调用模式用来在 Internet 上建立起发现服务。这些发现服务提供了一致的发布接口, 以使得能通过编程进行发现。

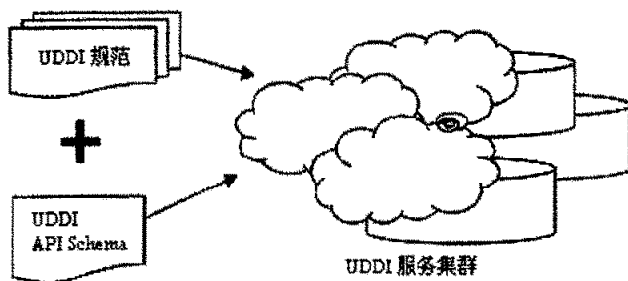


图 3-11 UDDI 规范、XML Schema 和 UDDI 商业注册中心集群之间的关系

通过使用 UDDI 的发现服务, 企业可以注册那些希望被别的企业发现的自身提供的 Web 服务。企业可以通过 UDDI 商业注册中心的 Web 界面, 或是使用实现了“UDDI Programmer’s API 标准”所描述的编程接口的工具, 来将信息加入到 UDDI 的商业注册中心。UDDI 商业注册中心在逻辑上是集中的, 在物理上是分布式的, 由多个根节点组成, 相互之间按一定规则进行数据同步。当一个企业在 UDDI 商业注册中心的一个实例中实施注册后, 其注册信息会被自动复制到其它 UDDI 根节点, 于是就能被任何希望发现这些 Web 服务的人所发现。

### 3.3 Web Service 对 Web GIS 系统的作用和意义

Web Services 是面向服务的体系结构的一个实现,其基本思想是利用平台标准(如 HTTP, XML),实现异构系统互操作,它具备平台独立,用户透明和轻松穿透防火墙等特点,是实现异构系统集成的理想计算模型。采用 .Net 技术的核心——Web Service 实现 WebGIS 系统,可以很好地解决目前 WebGIS 中存在的跨平台、软件复用等问题。

#### 1) 利用 XML 实现数据类型的标准化<sup>[57-59]</sup>

面向对象系统中,如何表示基本的对象是非常重要的问题。为了表示一个复杂对象如城市,需要这此简单几何体以组合和联合的方式来表示。但是在实际的 Web GIS 系统中,不仅对描述复杂对象的复合实体的表示方法不同,而且组成复合对象的基本数据类型的描述也有可能各不相同。对象之间错综复杂的各种关系更无法统一表示了。

XML 具有很强的扩展性。XML 提供了用户自定义对象的能力:提供了用继承、组成等关系描述信息结构的能力;而 XML Schema 支持语法解析器对信息流的自动处理。利用 XML 的可扩展性,可以使用不同的方式对数据类型的定义和解析,也可以使用这些基本类型组成复杂类型和复合对象。不同的 Web GIS 系统不需要再遵守同一套数据结构,而是可以自定义其数据类型,然后在 XML Schema 中给出这种数据类型的定义,其它系统就可以通过解析 Schema 文件来获得该 Web GIS 系统的特定数据类型。而且 XML 的命名空间保证了不同系统自定义的数据类型在重名的情况下仍不会发生名字冲突。

因为 XML 的巨大优势,OpenGIS 的简单特征规范就是使用 XML 来定义其数据类型。

#### 2) 利用 WSDL 实现软件复用

Web Service 可以简单地认为是网络上的组件。而且由于组件的功能是以 WSDL 文件的方式暴露给外界,本身的实现被完全隐藏。同时,由于采用了 XML/GML 定义数据类型,可以表示一些复杂的地理信息类型,所以 GIS 中的数据模型可以被完全表示出来,根据 XSLT 不同的数据模型之间也可以很方便地实现转换。对数据的处理与数据放在一起,不同的数据,其内部操作的方式不尽相同,但是因为都遵守一定的接口规范,所以外部调用的方式完全相同。正像面向对象里面的“多态”一样,相同的函数名可以由不同的对象来处理。在 Web Service

中也是一样，相同的函数，无论是在本机还是远端，无论是那种实现方式，都可以由同一个函数来调用。而针对同一个数据的操作构成的函数集，就可以看作是对此数据进行操作的一组函数，把这组函数统称为一个网络功能组件。可以供其它应用程序调用。

利用网络组件的方式，在应用程序之间流动的，只是命令和响应，数据和程序都不需要在网络上传输，节省了带宽和计算时间。

### 3) 利用 WSDL 和 SOAP 实现跨平台

对于跨平台的调用，有很多种方法可供选用，包括以前的 DCOM、CORBA、RMI 等方式。但是这此方法本身就有很多的对硬件，操作系统，开发语言，运行环境的限制，并不能完全达到透明地远程功能调用的目的，更不用说不同方法之间的调用了。

组件化软件的思想的引入，将一个组件的功能描述和其功能的实现完全分离，使得人们可以在不关心底层实现的情况下设计和构建一个大型的系统，在运行程序级实现了软件功能的复用。Web Service 也采用了这种思想，使用标准的协议和接口，用 WSDL 语言来描述功能接口，屏蔽了具体实现的方式（体系结构，包括硬件，操作系统，语言，运行方式）。由于 Web Service 的各个功能组件之间采用松散耦合的方式，系统一个节点的更新、崩溃、更换操作系统不会影响到整个 Web GIS 系统的其它节点。

DCOM、CORBA、RMI 这几种模型，更多地偏重于局域网环境下的跨平台。对于广域网环境下的跨平台，情况非常复杂，单就是两台机器之间的通信问题，就必须考虑很多种情况。在 Web Service 中，采用 SOAP/HTTP 标准协议实现服务的通信方式。这样就可以利用已有的成功的通信方式，不需要自己再设计一套完全不同的通信方式来进行客户端和服务器端之间的通信。

#### 4) 引用 UDDI 实现 Web GIS 组件的共享

在 Web Service 中，采用 UDDI 注册中心的方式来注册和发现服务，使得用户和服务提供者之间的沟通更加容易。服务的注册、发现和绑定模式，使得应用程序对服务不再是静态依赖关系。因为应用程序可以通过 UDDI 注册中心来动态更新有关服务的元数据，该元数据可以包括服务的版本号、更新时间、服务 URL 位置、服务功能等信息，从而供应用程序来选择和使用。所以服务的动态添加、删除和更新不会对已有的应用程序产生影响。可以利用不同组织、机构或者个人开发的不同 Web GIS 组件来构建 Web GIS 系统，从而实现了 Web GIS 组

件的共享。

### 3.4 本章小结

随着计算机软、硬件技术的发展特别是计算机网络的飞速发展,软件开发规模和复杂度不断增加,软件开发面临巨大挑战。.NET 作为一个全新的分布式计算框架,利用现有的网络基础设施、协议规范、Web 和数据库技术,为用户提供一体化的智能信息平台,创建了一种架构在 OS 和 Web 之上的基于 Internet 的新一代信息平台和软件基础设施,旨在消除“信息孤岛”,解决软件危机,实现网上计算资源、数据资源、信息资源、应用资源的智能共享。

.NET 技术是基于 XML, SOAP, WSDL, UDDI 等开放标准和协议的。

XML 是表达数据的标准方式的通常选择。大部分相关 Web 服务都使用 XML 作为数据表达方式,就像使用 XML 模式来描述数据类型一样。

简单对象访问协议 SOAP 为信息交换定义了一个精巧协议。SOAP 的一部分说明了使用 XML 来描述数据的一些格式。另外一部分 SOAP 定义了一个可扩展的消息格式,用于方便地使用 SOAP 消息格式描述远端程序 (RPC),并且和 HTTP 协议进行捆绑 (SOAP 消息可以通过其他协议交换)。

Web 服务描述语言 (WSDL) 是一种以 XML 为基础的协议语言,它提供了一种标准的方式来记录 Web 服务所接收和产生的消息。也就是说,记录 Web 服务协议。一套标准的机制有利于开发人员和开发工具创建和解释协议。

开发人员还需要一些方法来发现 Web 服务。UDDI 确定了一种机制,使得 Web 服务提供商可以为其 Web 服务做广告,也使得 Web 服务用户能找到其感兴趣的 Web 服务。

以上这些技术与 Web 服务共同构成了 .NET 的技术基础。

将 Web 服务引入 GIS,一方面可以很好地解决多年来 GIS 在互操作实现方面所面临的困难,另一方面,方便 GIS 融入其他应用系统和与其他应用系统集成。GIS Web Services 把 GIS 的数据和功能以服务的形式在网络中发布,服务使用者不需要了解服务的具体细节就可以直接在应用中使用网络服务。开发者可以调用分布在不同节点的网络服务,结合本地开发的功能,很快就能完成一个比较完整的 GIS 系统。因此,GIS Web Services 可以很好的解决传统 Web GIS 无法跨平台、无法实现异构空间数据互操作、开发调试困难以及资源共享等问题。

在第 5 章系统应用实践中,应用服务器端的设计将用到 Web Service 技术和 XML, SOAP, WSDL, UDDI 等开放标准和协议。

## 第 4 章 地理信息 Web 服务平台架构与体系结构

地理信息 Web 服务除了采用基本的 Web 服务技术协议外,还需要有关地理空间信息及处理的技术协议,目前主要由 OGC, ISO/TC 211 和 W3C 制订。OGC Web 服务 (OWS) 是典型的地理信息 Web 服务。

### 4.1 OWS (OpenGIS Web Service) 计划

“OGC Web 服务是一个基于标准的实现各种在线地理信息处理和位置服务无缝集成的框架。OGC Web 服务使得分布式地理信息处理系统能够使用 XML 和 HTTP 技术相互通信。OGC Web 服务提供了一个以服务(提供者)为中心的互操作框架,支持多种在线地理数据源、传感器产生的信息和地理信息处理能力的基于 Web 的发现、访问、集成、分析、利用和可视化。”为了更好地将地理信息服务的概念扩展到网络上,以 Web Service 的方式提供服务,OGC 建立了 OpenGIS Web Services (简称 OWS 研究) 研究计划。

OWS 是可以通过 Web 发布、查找和调用的自包含、自描述的模块化应用。OGC Web 服务能够用元数据(能力)来描述其实现的操作(或操作集),可以用 URL 和网络可访问的地址定义来寻址。OWS 是构建网络连接的地理信息处理应用或集成地理信息处理功能到其它信息应用(如客户关系管理 CRM 和企业资源规划 ERP) 的支撑框架。

### 4.2 OGC 的 GIS 服务框架

对于 Web 服务在空间信息领域的应用,OGC 表现出极大的关注<sup>[40]</sup>:

1) Web 服务具有广阔的应用前景,Web 服务代表了一个具有革命性的,基于标准的框架结构,它可以让各种在线的空间数据处理系统和基于位置的服务之间无缝地集成。它可以让分布式的空间数据处理系统使用目前广为流行的技术,例如 XML 和 HTTP 通过 Web 进行互相通讯。它提供了与厂商无关的、可互操作的框架结构来对多源、异构的空间数据进行基于 Web 的发现、数据处理、集成、分析、决策支持和可视化表现。

2) Web 服务是一个构建分布式系统的平台,是一个为空间数据处理应用建

2) Web 服务是一个构建分布式系统的平台,是一个为空间数据处理应用建

立网络连接的框架结构,或者是将空间数据处理功能与其他信息应用系统如 MIS 和 ERP 系统进行集成的平台。Web 服务的提供者即可以是提供空间数据处理功能的服务器,也可以是这些服务器的客户端。因此,Web 服务是一个提供可互操作的、开放的、动态链接的空间信息服务网络体系平台。

3) Web 服务将会使得未来的空间数据处理系统和位置服务等通过 Web 有机地联系在一起,它将是一个自我包含,自我描述,模块化的应用,可以用于数据的发布,访问,以及通过 Web 来调用。一个 Web 服务可以认为是一个“黑箱”,它屏蔽了操作细节,通过提供一系列的访问接口来提供空间数据的服务。它可以以元数据的形式来描述所执行的操作,因此可以通过 UDDI 来获得这些服务的相关信息。

OGC 定义的地理信息 Web 服务的框架结构如图 4-1。

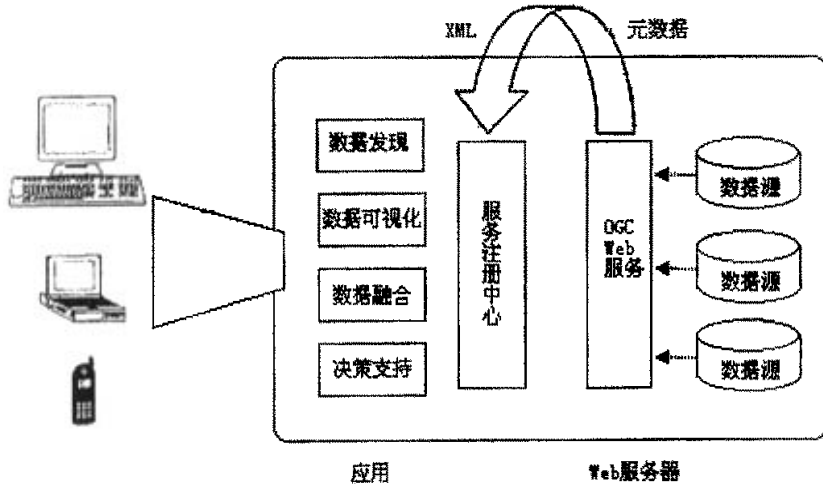


图 4-1 地理信息 Web 服务框架的结构

### 4.3 OGC Web 的服务协议栈

地理信息 Web 服务互操作协议栈如图 4-2,描述了支撑服务实现和部署的技术和标准的层次体系结构。最底层通过绑定、发送和接收消息实现软件组件的连接。更高层通过发布、发现、绑定机制实现互操作,使得软件组件能够透明地以集成和动态方式一起工作。其中,地理空间数据表现和编码、地理空间数据格式、模式和语义以及地理信息服务的接口和实现构成了 OGC 的 OpenGIS 服

务框架。这个协议栈规定了地理信息 Web 服务互操作、集成与共享的机制。

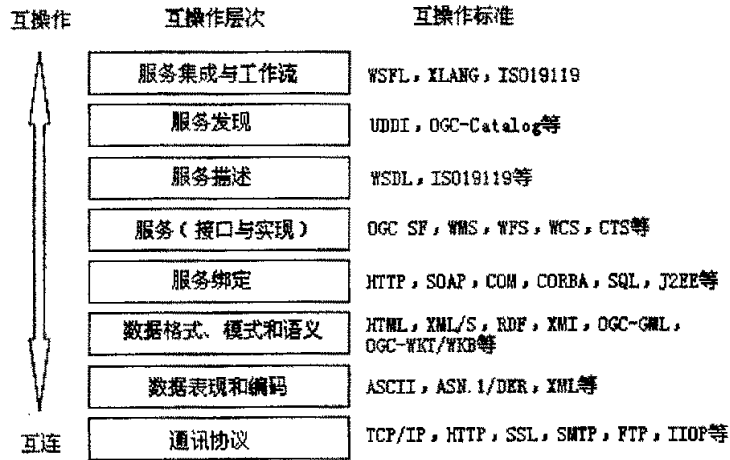


图 4-2 地理信息 Web 服务协议栈

图 4-2 描述了技术标准的分层体系结构。可以看出，这个栈基本上是基于 W3C 的 Web Service 协议栈的，只是在各层增加了许多 OpenGIS 的规范。如在数据格式层，不仅有 W3C 标准的 HTML, XML 等规范，也有 OGC 制订的 GML 编码规范等。在服务发现层，既可以使用 UDDI 注册中心，也可以使用 OGC 的 Catalog 规范。在绑定和服务层之间，增加了一个服务层，该层定义了 OGC 服务的一些基础规范如坐标变换、图层规范等，这些规范定义了基本的网络地图服务。

#### 4.4 OGC Web 服务的实现规范

OGC 网络服务规范包括三种基本地理参考信息访问服务：网络地图服务 WMS，网络图层服务 WCS 和网络特征服务 WFS。此外，还有其它服务像地理编码和地理解码能够返回空间参考。

OGC 将地理信息服务的实现规范按照服务分为以下几类<sup>[50]</sup>：

1) 核心服务 (Core Service)：它们是不考虑应用领域的通用接口，用以支持其它应用领域服务的。Open GIS 的核心服务规范包括：坐标转换规范，目录规范，服务注册规范等。

2) 网络制图服务 (Web Mapping Services)：这些服务规范使得遵循 Open GIS 规范的厂商或组织开发的服务器之间就 Web 上不同类型的空间信息可以进行动



态查询，存取，转换和综合等处理。目前 Open GIS 开发的规范包括网络地图服务规范(Web Map Server)，网络特征服务规范(Web Feature Server)和网络图层服务规范(Web Coverage Server)等。

OGC 下面将专注于制定服务注册和该注册的接口、XML 编码的传感器协议语言 SensorML 以及与传感器 Web 进行交互的接口规范等。

3) 位置服务 (Location Services): 该服务依赖网关服务。网关服务是将位置应用服务与公共的移动终端、无线平台、护平台、移动位置确定系统集成在一起。

地学空间融合服务 (Geospatial Fusion Services): 该服务将地址、地方名称、坐标、图像上的点、描述性方向等信息融合进一个信息管理框架，能够支持查找、发现和共享非地图格式的空间信息。目前包括地名服务接口 (Gazetteer Service Interface)，地理编码服务 (Geocoder Service)，地理解析器服务 (GeoParser Service)，位置组织器文件夹 (Location organizer Folder) 等规范，目前这些规范都处于讨论阶段。

下图 4-3 表示了一个概念上的 Web Services 相关的体系图<sup>[60]</sup>。

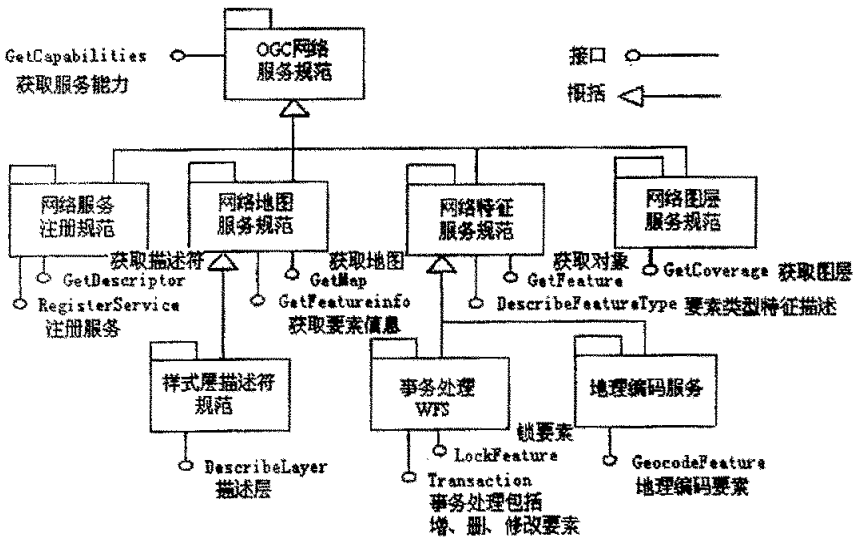


图 4-3 OGC 网络服务体系结构图

#### 4.4.1 GML——地理信息表示的基础

GML (Geography Markup Language) 是由 OGC 提出的<sup>[52]</sup>。GML 主要特点表现

在以下几个方面:

- 1) 表示地理空间现象, 不仅有简单二维现象特征, 也包括复杂的非线性的 3D 几何体, 带 2D 拓扑的 feature, 带时间属性的 feature, 动态 feature, 图层 (coverage) 和观测值 (Measurement)。
- 2) 提供更多对特征属性的支持, 以及其它取值非常复杂的对象。
- 3) 表示空间和时间参照系统, 测量单位和标准信息。
- 4) 在表示地理空间现象、观测结果和值时使用参考系统、单位和标准信息。
- 5) 表示特征和图层可视化时的缺省样式。

#### 4.4.2 网络地图服务规范 (Web Map Service)

网络地图服务规范 WMS (Web Map Service) 规范定义了客户端请求地图的方式的标准。地图 (map) 是地理数据的视觉表示, 地图不是数据本身。

一个 WMS 可以生成具有地理参考数据的地图。这些地图通常用 PNG、GIF 或者 JPEG 等栅格图形格式, 或者用 SVG 和 WebCGM 等矢量的图形格式来表现。该规范对客户端对地图的请求以及服务器端的服务描述加以标准化。WMS 定义了三个操作, 其中前两个操作是任何一种 WMS 所必须的。这些操作是<sup>[69]</sup>:

GetCapabilities (必须): 返回服务元数据, 这些元数据必须是用户或者机器可以识别的, 描述 WMS 的服务内容和可接受的参数。

GetMap (必须): 返回一幅具有正确的地理空间和维数参数的地图图像。

GetFeatureInfo (可选): 返回关于地图上特定特征的信息。

在 WMS 中 GetMap 是最重要的一个接口。GetMap 的参数有很多, 在调用时必须的参数主要有下面一些:

表 4-1 GetMap 请求的必须参数

请求参数	定义
VERSION=version	请求版本号
REQUEST=GetMap	请求的名称
LAYERS=layer_list	以逗号分隔的图层列表
STYLES=style_list	以逗号分隔的每层渲染样式列表
SRS=namespace:identifier	空间参考系统
BBOX=minx, miny, maxx, maxy	矩形框四角的 SRS 坐标
WIDTH=output_width	地图图像的宽度
HEIGHT=output_height	地图图像的高度
FORMAT=output_format	地图的输出格式

用户可以通过标准的浏览器向 WMS 发送地图服务请求, 请求的方式可以通过在 URL 中添加请求的类型参数来实现。WMS 的 GetMap 操作支持分布式的地图服务网络来提供给用户。

#### 4.4.3 网络特征服务规范 (Web Feature Service)

网络特征服务规范 WFS 支持地理特征的插入、更新、删除、查询和发现等功能。WFS 根据 HTTP 客户端的查询返回 GML 表示简单地理空间特征数据<sup>[60]</sup>。

为了支持交易和查询处理, 定义了如下的操作:

GetCapabilities: 一个 WFS 必须能够提供它所提供的服务功能, 也就是说它能提供那些特征类型的服务, 以及针对每一种特征类型的操作。

DescribeFeatureType: 一个 WFS 必须能够根据请求来描述任何可提供服务的特征类型的结构。

GetFeature: 一个 WFS 必须能够对请求进行服务来获得特征的实例。此外, 客户端能够指定获取那些特征属性, 以及对空间和非空间的查询进行约束。

Transaction: 一个 WFS 必须能够对交易的请求服务。一个交易的请求是对特征数据的操作, 包括创建, 更新, 删除等操作。

LockFeature: 一个 WFS 可以在交易期间处理对一个或者多个特征类型实例的锁定请求。这样就确保了交易的连续。

#### 4.4.4 网络图层服务规范 (Web Coverage Service)

网络图层服务规范 WCS 支持地理空间数据的网络交换, 用于交换的数据是包含地理位置或属性的图层。不像 WMS 向客户端返回在服务器端渲染的静态图片, WCS 提供了完整的未渲染的地理空间信息数据, 这些数据既可以在客户端进行渲染, 也可以作为地理模型和其它复杂的客户端的输入。

WCS 包括三个操作:

GetCapabilities: 返回服务的描述, 该服务的元素用来描述一个图层所请求的多维数据集。

GetCoverage: 返回地理位置的值或者属性, 这些值以一种已有的图层格式结合在一起。

DescribeCoverageType: 返回对图层结构的描述。

#### 4.4.5 样式层描述符规范 (Styled Layer Descriptors)

该规范应用于 WMS。一个基本 WMS 将其信息保存为图层并提供了预定义的显示格式。SLD 规范允许用户使用自定义的符号来表示特征数据。SLD 使得 WMS 可以从 WFS 上获取特征并使用用户定义的格式信息来渲染地图。

SLD WMS 给基本 WMS 增加了下面额外操作：

DescribeLayer 描述层

GetLegendGraphic 获取图例的图形

GetStyle 获取样式

PutStyle 设置样式

### 4.5 OGC 规范对 Web GIS 系统的作用和意义

#### 1) 利用抽象规范实现统一的数据抽象模型

OGC 的抽象规范综合了大部分的 GIS 系统模型的优点，给出了一个统一的标准的数据模型，遵从这个模型设计的 GIS 功能组件之间就可以互相操作对方的数据。

#### 2) 利用 GML 实现地理数据的编码

在 GML 中，实现了复杂对象、时间特征、空间参考等多种地理数据特有特性的表示，利用 GML 可以完全表示抽象规范的数据模型。GML 可以作为地理数据存储、传输或交换的基础。这样，在不同系统之间的交互的就可以以 GML 作为数据编码基础。同时，由于 GML 是以 XML 作为基础进行扩展的，所以能够对 XML 进行处理和解析的各种方法都可以用来处理 GML，使得以 GML 为基础的 Web GIS 可以有更多的灵活性。

#### 3) 利用地理信息网络服务实现互操作——建立对数据的操作集

采用 Web Service 的方式，在数据集上定义一组 Web 服务，客户端通过 Web 服务的方式来访问数据集。OGC 只规定了提供网络地理数据服务的几个基本功能，但是从这几个基本功能出发，完全可以获得服务器提供的所有数据集。

#### 4) 利用 Catalog 的方式实现地理信息网络服务的发现、查找和绑定

利用 Web Service 的方式，可以将一些基本的地学处理功能独立出来，以公共组件的方式进行使用。如投影系统的转换，不同的 GIS 使用的是相同的算法，所以将这些算法作为独立组件提供给每个 GIS 系统。任何需要进行投影转

换的应用程序输入 GML 数据，就可以得到处理后的结果。

为了有效地组织已有的数据和服务组件，类似于 UDDI 以及 ISO 的 Trader 的概念，在 OpenGIS 中，使用 Catalog 的方式。在目录服务中，数据和服务的提供者和消费者之间有一个公共的交换场所，这样能够更好地实现 WebGIS 组件的共享和交互。

## 4.6 本章小结

由于 Web Service 的迅速发展和巨大影响力，OGC 启动了 OWS 计划，力图将其已经制定的网络地理信息服务规范和 Web Service 技术结合起来，将更多的与地理信息网络服务有关的内容纳入到其研究和制定标准的范围来，这些包括 OGC 服务框架，OGC 服务协议栈，OGC Web 服务实现规范等。

本章首先介绍了 OGC 的 Web Service 研究计划 OWS (OGC Web Services)，然后 OGC 服务框架 OSF (OpenGIS Service Framework) 和 OSF 协议栈等框架性内容，然后介绍了 OGC 相关的一系列实现规范：基于 XML 和 OGC 的对象模型所产生的地理信息表示语言 GML，提供地图服务的网络地图服务规范 WMS，提供图形服务的网络特征规范 WFS 以及其它一些比较重要的实现规范。最后阐述了使用这些规范对开发标准的开放的 WebGIS 系统的重要意义。第 5 章在本章对 OGC 相关标准研究的基础上，提出了基于 .Net 的电网配电网线路和设备管理的 WebGIS 解决方案。

## 第 5 章 系统应用开发实践

### 5.1 系统概述和分析

城市配电网是一个纵横交错的巨大网络，具有十分复杂的空间和非空间属性。传统工作方式中，这些信息一般经图纸图表描述(例如，施工和竣工资料图表)，采用人工管理，长期以来，在管网资料的管理方面一直沿用这种做法。各种图档、资料的不齐全使管理人员在管理这样一个庞大的管网时感到力不从心。为了彻底改变这一现状，必须引入计算机技术——地理信息系统来管理我们的电网资料。要实现对供电网络图形数据资料的科学管理，只是依赖网络和数据库技术还显不够，必须采用最新的 GIS (地理信息系统) 技术。而供电网络中的设备信息绝大部分含有空间信息，而利用空间信息进行空间分析更是供电行业电网管理信息化的重要特点。如供电管网中的电杆、变压器等电力设备在城市中的分布。由于 MIS 缺乏对空间信息的管理，自然也就谈不上进行空间分析。管理人员只能进行统计报表处理、属性数据查询等工作，而不能立即建立对其所管理业务的空间概念。GIS 中的 GMIS (空间管理信息系统, Geographical Management Information System) 功能就解决了 MIS 所存在的这个问题。它的基本实现手段就是在 MIS 功能的基础上，能够同时对空间信息进行分析和管理，简而言之，就是在 MIS 上加了空间图形信息。例如在普通的 MIS 系统中，一根电缆只包含了诸如编号、型号材质、长度等属性数据，而在 GMIS 系统中，除了包含上述属性数据外，还包含了这根电缆在城市街区图中的走向分布以及与变压器等其它设备间的拓扑连接关系。本文所论述的正是基于 .Net 及重要技术平台 MapXtreme 来实现电网中配电线路和设备管理的 WebGIS 解决方案。

#### 系统管理的区域及主要使用对象

系统管理的区域为广西省北流市城区中的电力设备。实时监视范围为全市的主要线路、设备。配网 GIS 的主要使用对象为：配电工区、客户中心、市区所、计量所、电力工程公司及其它有关部门和领导。

### 5.2 系统基本配置及开发工具

#### 1) 服务器端配置

操作系统: Windows 2000 Advanced Server;

硬件环境: Intel Pentium III 处理器, 256MRAM, 40G 硬盘;

Web 服务器: Microsoft IIS5.0 和 Visual Studio.Net2003;

地图服务器: MapInfo MapXtreme2004;

## 2) 客户端配置

操作系统: Windows 98/NT/2000;

浏览器: IE 或 Netscape ;

## 3) 软件开发工具

地图编辑工具: MapInfo Professional 6.0;

界面设计: DreamWare;

数据库: Oracle, MapInfo 本身提供的数据库引擎。

## 5.3 系统的功能模块

### 5.3.1 主要功能

在任何时候用户都可以对地图进行放大、缩小、漫游、打开及关闭图层等地图浏览功能。除此之外, 系统还提供维护更新、查询分析、专线电网/变电器定位、距离测量等功能模块。

1. 维护更新功能: 用户若要维护子系统, 可按照维护向导很方便地登录, 选择专题, 填表, 增加、删除资料等。更新后的资料在系统中立即生成, 供各专题子系统使用。

2. 查询分析功能: 可以在地图上查到高低压线路、配变、开关箱等各种电力设施的位置, 以及该电力设施的各种技术参数等。

3. 专线电网、变电器定位功能: 可以按选定的要求, 查询如专线名称、配变型号而列出相应的线路或和配变的位置。

4. 距离测量功能: 用此功能, 在 GIS 上可轻易量出任意两点或三点之间线路长度和供电半径。

5. 鹰眼图功能: 可方便地在放大的主地图和小幅全局图之间互动式缩放并移动地图。

6. 打印功能: 通过线路层、地理层的随意叠加, 可以很清楚地得到线路、电缆等分布图, 并可打印。

### 5.3.2 功能体系结构

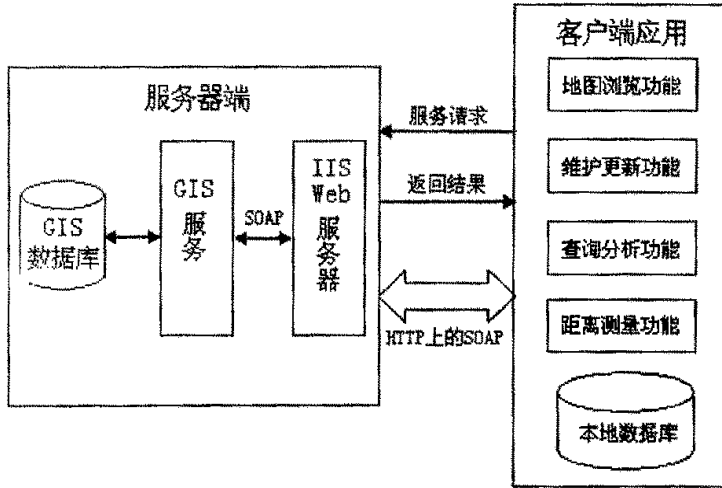


图 5-1 WebGIS 地理信息的功能体系结构

按照模型的功能要求，主要分为客户端应用开发和服务器端 GIS 服务的开发两个部分，在这种结构下，客户端完全通过 WWW 浏览器实现，一部分事务逻辑在前端实现，但是主要事务逻辑在服务器端实现，客户端只要有常用的 Web 浏览器即可，不需要安装任何插件，服务器端主要实现基本地图操作的功能，比如：浏览地图、放大、缩小、移动、点选等等。同时还需实现的 GIS 服务遵循 Web 服务技术和其相关规范，GIS 服务通过标准 Web 协议配置的，这些协议包括 HTTP, WSDL, UDDI, 和 XML。它们使用基于 XML 的简单对象访问协议进行通讯，使 GIS 服务能和当前大多数 Web 服务兼容。在客户端通过查询 UDDI 注册中心，发现所需 GIS 服务，分析其描述文档 WSDL，了解接口和类函数，嵌入到客户端应用程序中，其中使用 SOAP, HTTP 进行通信，使得 GIS 服务很容易穿透防火墙。

应用服务器和客户端采用 ASP.Net 来进行设计和实现。开发环境选择微软公司的 Visual Studio.Net2003，以 MapInfo 公司的 MapXtreme 做为地图平台，服务器端脚本选择微软公司针对 .Net 平台量身打造的 C# 语言，客户端以 JavaScript 为脚本语言，采用 B/S 体系结构，这种开发方式减少了客户端与服务器的通信，提高了程序运行效率，可以高效、快速、方便、灵活的实现 GIS 的基本功能。Web 服务出现为 GIS 的发展提供前所未有的动力和机会，我们可以



基于这一分布式平台建立一个异构的 GIS 网络。

## 5.4 系统 GIS 平台

目前市面上流行的 GIS 开发平台有 MapXtreme、ArcIMS、MapGuide 等，其中 MapXtreme 是美国 MapInfo 公司积累多年在公用事业、通讯领域实施 GIS 系统的经验，结合最新的开放图形和数据库技术开发的新一代的企业地理空间资源管理解决方案平台。强大的电力系统技术支持使其成为开发大型电力 GIS 的首选开发平台。该平台是 GIS 在面向 Internet/Intranet 应用的实现工具，在 WEB 方式时采用流行的三层结构，可通过浏览器，将所有的用户界面逻辑移至 Web 服务器，将业务处理和地理空间设施网络模型充分结合起来，解决分布数据访问，而且支持最新的 .Net 技术。

MapXtreme 技术系统主要由三个层次组成。顶层是 Internet 技术，在这一层中，有非常成熟的 TCP/IP、Web 等技术，并且有 CGI, HTML 等标准，保证了其标准性、开放性和先进性。底层是 MapInfo Professional 技术和基于 ActiveX (OCX) 技术的 MapX。MapInfo Professional 经过十多年的发展，已经相当稳定和完美，足以担当空间数据库内核；MapX 是一个性价比好、功能强大的 OCX 控件，提供了真正的对象联接与嵌入 (OLE) 组件。在这两者之间，是 MapXtreme 模块，其不仅是 MapX 在 Internet/Intranet 系统上的扩展，更代表了新一代的地图应用服务器模式，它采用了很多新技术，如内置开发工具、进程调度器分布应用模式等其应用模型如图 5-2 所示。

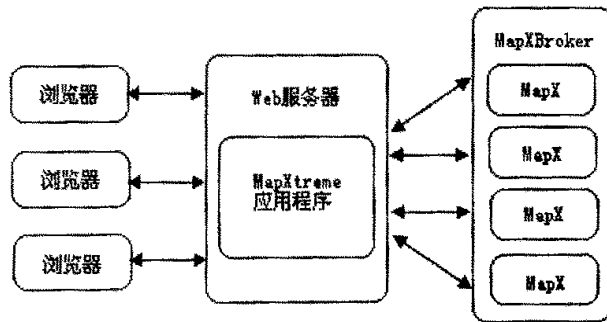


图 5-2 MapXtreme 应用模型

在编写、调试完成 MapXtreme 脚本以后，脚本被发布在 Web 服务器上，当被

客户端访问时,就会自动执行。比如,用户在页面上单击了一个按钮以后,在脚本文件中响应该动作的脚本会自动被执行,产生一个对地图引擎的调用。所谓地图引擎,实际上就是一个 MapXtreme 对象,它有一组对 MapX 进行访问的属性和方法。在脚本和地图引擎通信的过程中,地图引擎返回给脚本一个格式为 GIF 或 JPEG 的图像文件上,这个图像即为将被放到 HTML 页面中显示给用户的最终的地图图像。地图引擎提供了应用程序和 MapX 通信的渠道,它有三个重要组件: MapX, MapXtreme 和 MapXBroker。

#### 1) MapXtreme 的运行机制

Map Engine 为应用程序提供 MapX 对象。MapXtreme 功能的核心是 MapX。MapXCourier 可实现应用程序和地图引擎的沟通。MapX 是可编程的 OCX 控件。它除提供显示 MapInfo 格式地图,对地图放大、缩小、漫游、选择等操作外,还提供如专题地图制作、图层控制、动态图层和用户绘图图层控制以及生成和编辑地图对象、简单地理查询等强大的 GIS 支持功能,可利用编程平台所提供的数据库访问机制,也可利用自身提供的 ODBC 接口,绑定数据。MapXServer 是 MapX OCX 的包装,被 MapXCourier 或者 MapXBroker 启动。MapXBroker 用来预启动 MapXServer 实例。MapXCourier 能使应用程序通过 MapXServer 实例来达到访问 MapX 的目的。

#### 2) MapXtreme 应用程序的工作流程

当用户在客户端进行相应的地图操作后,客户端浏览器便向 Web Server 发出一个 HTTP 请求,递交 Form 表单。当应用程序请求 Map (地图) 服务时,地图引擎 (Map Engine) 将通过在 MapXtreme 中产生的一个 MapXCourier 对象使应用程序通过 MapXServer 实例访问 MapX 对象,并加载 MapX 对象所包含的 geoset 地图集。ASP 脚本语言应用程序则根据提交表单的信息,判断用户的地图操作请求类型,并对 MapXtreme 地图应用服务器的 Map 引擎进行访问,Map 引擎进行相应的空间分析和处理后,便将处理结果以位图 (.GIF, .JPG, .PNG 等) 的形式传递给 Web Server, Web Server 再将此位图文件嵌入到 HTML 页面中并返回到客户端的浏览器上。

MapXtreme 的这种 B/S 应用程序的工作过程,实际上就是用户浏览器与应用服务器之间通过静态或动态的 Web 页面进行信息交互的过程。这种浏览器/服务器 (Browser/Server) 式的优点是: 由于用户端所需软件只是一个简单的浏览器,用户基本上无需培训,用户端软件也无需维护;软件的升级与修改只在服务器

端进行，对用户透明；服务器与浏览器可处于不同的操作系统平台<sup>[66]</sup>。其工作的具体流程见图 5-3。

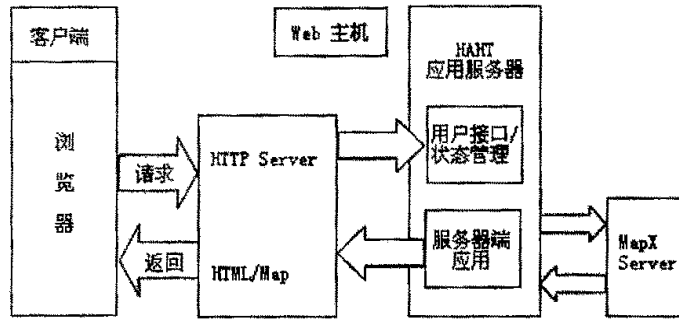


图 5-3 MapXtreme 的工作流程

由于 MapXtreme 是一个功能强大的 GIS 平台，平台并不直接提供表示电网的配电线路和设备信息，也不能直接进行 Web 发布，必须对平台进行二次开发来实现 Web 方式下在电网 GIS 中配电线路和设备显示管理。而且在实际调研过程发现，在 WebGIS 中，用户总是希望能够在 Web 页面实现对全网或者任意感兴趣的线路、变电站的结构信息进行浏览监测。因此基于 .Net 平台实现电网 GIS 中配电线路和设备显示管理的关键技术首先是如何实现地图以及地图数据的 web 发布，从而使得企业用户能够在任何时间、任何地点能够根据自己意愿查询、浏览电力网 GIS 中的相关地图以及相应的配电线路和设备数据。

## 5.5 客户端功能、设计和实现

### 5.5.1 客户端的界面设计

客户端的界面如下图所示 5-4 所示。

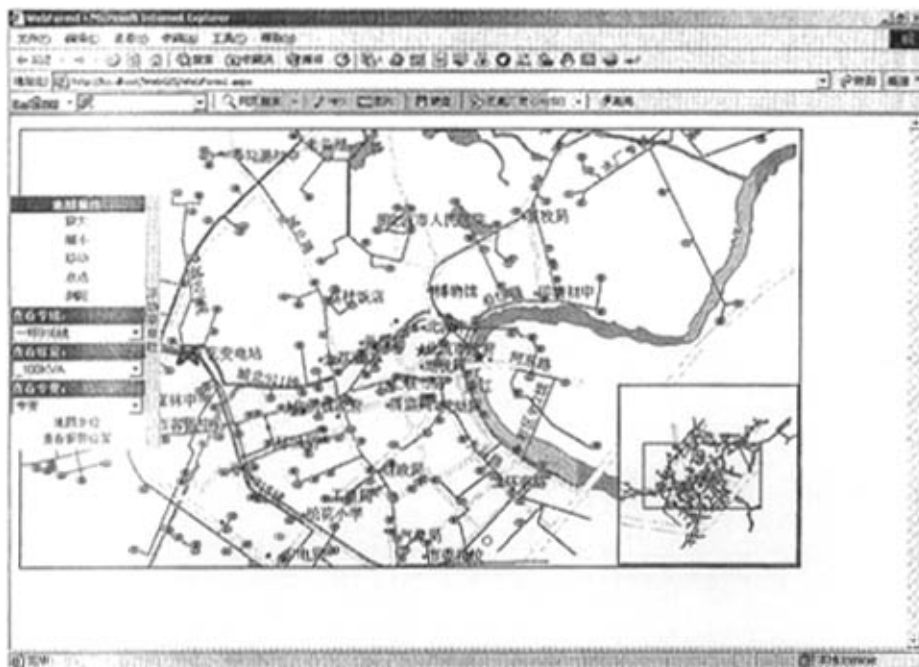


图 5-4 客户端的界面

### 5.5.2 客户端的功能和实现

由于主要的应用逻辑都在应用服务器端实现，所以客户端的程序非常简单，主要是界面的设计以及向应用服务器提交命令请求服务器端进行解析。

在客户端设置地图操作当前工具：

1) 先定义工具名变量：

```
var MapInfoWebZoomInTool = "MapInfoWebZoomIn";
var MapInfoWebZoomOutTool = "MapInfoWebZoomOut";
var MapInfoWebPanTool = "MapInfoWebPan";
var MapInfoWebCenterTool = "MapInfoWebCenter";
var MapInfoWebPointSelectionTool = "MapInfoWebPointSelection";
var MapInfoWebRadiusSelectionTool = "MapInfoWebRadiusSelection";
var MapInfoWebPolygonSelectionTool =
    "MapInfoWebPolygonSelection";
```

```
var MapInfoWebDistanceTool = "MapInfoWebDistance";
```

```
var MapInfoWebInfoTool = "MapInfoWebInfo";
```

2) 通过 ID 获得地图实例:

```
var obj = MapInfoWebGetMap('MapControl1');
```

3) 对获得的地图实例设置当前工具:

MapInfoWebSetCurrentTool(mapid, toolName) 例如设置当前鼠标为点击放大, 对其进行设置为:

```
MapInfoWebSetCurrentTool('MapControl1', MapInfoWebZoomInTool);
```

下面给出一个地图图层加载的代码:

MapXtreme 操作地图图层大体上可以分成以下 6 步, 完成图层的载入、显示。MapXtreme 的抽象分析可分成 Map、Catalog、ISession 三个模块如图 5-5 所示:

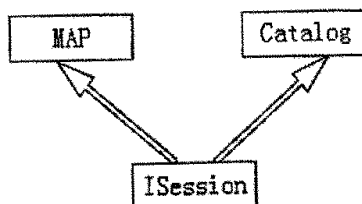


图5-5 MapXtreme的抽象模块

- i. 获得 ISession——程序的起点 ISession session = Session.Current;
- ii. 获得 Catalog——管理数据 Catalog catalog = Session.Catalog;
- iii. 获得 Map——管理地图 Map map = Session.MapFactory[0];
- iv. 打开表 Table t = catalog.OpenTable("Tab 表对应的路径");
- v. 创建图元图层 FeatureLayer lyr = new FeatureLayer(t);
- vi. 把图层加入到地图中 map.Layers.Insert(0, lyr);

地理信息系统的数据包括两部分: 属性数据和空间数据。这两种数据是有联系的。在设计数据库的结构及数据的分布时, 要尽量使数据库结构简单、冗余度小, 数据分布合理, 以提高系统的运行效率。在本系统中, 将属性数据和空间数据结合在一起, 存储在 Oracle 数据库中, 这样做比将空间数据文件和属性数据分开来存储方便查询、修改、和维护, 减少了不必要的麻烦, 并且这又是 GIS 发展的必然趋势。这种情况下, 直接从数据库加载空间数据和属性数据到 FeatureLayer 对象中, 程序执行时, 系统根据 FeatureLayer 自动将地图数据显

示出来，同时属性数据也已经加载到图层中了。

## 5.6 应用服务器端 GIS 功能、实现

GIS 服务的基本模式如下图 5-6 所示：

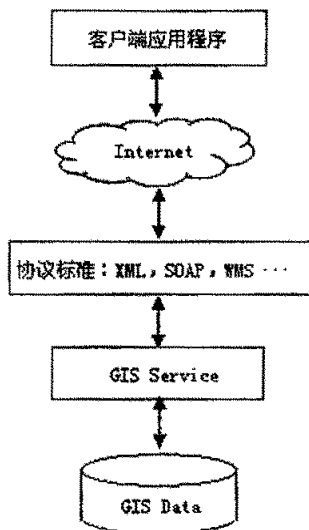


图 5-6 GIS 服务的结构图

根据前面研究基于 Web 服务 GIS 的模型，在服务器端设计实现 GIS 服务。首先设计地理基本功能服务，通过访问服务器端的地理信息数据，客户端可以借助功能服务和服务器端进行地理数据的交互或者直接享受数据服务。

### 5.6.1 GIS 服务生成与接口

Web 服务的开发周期包括对服务注册库、服务提供者、服务请求者每个角色的设计、部属和允许的需求，分为四个阶段：

#### 1) 开发阶段

包括 Web 服务实现的开发和测试、服务接口描述的定义和服务实现描述的定义。Web 服务的实现可通过三种途径来实现：创建新的 Web 服务，将现有的应用转换成 Web 服务，以及从其他 Web 服务和应用组合成新的 Web 服务。

以基于 Web 服务开发为例，首先是编辑生成 Web 服务实现的服务器程序或者把现有的应用程序改为服务器程序，编译生成可执行文件（.exe）或者动态链接

库文件 (.dll)之后, .NET 平台会自动生成描述 Web 服务的 WSDL 定义文件<sup>[67]</sup>。

#### 2) 部署阶段

包括发布服务接口描述和服务实现描述到服务请求者或服务注册者, 部署 Web 服务的可执行代码到运行环境中, 典型的应用环境就是 Web 应用服务器。具体的说首先在 Web 服务器中新建一虚目录, 配置好相关名称和端口及读取权限等等, 然后将开发阶段生成的 WSDL 文件、可执行文件或者动态链接库文件以及相关文件拷贝到 Web 服务发布的相应虚目录中。

#### 3) 允许阶段

在这一阶段, Web 服务是可用的, 即 Web 服务是从服务提供者的角度已部署就绪, 可操作和网络可访问性。服务请求者可以执行查找和绑定操作, 可以是静态和动态方式。

#### 4) 管理阶段

涉及 Web 服务应用的持续调控和管理, 安全性、可靠性、性能、服务质量和业务流程都需要管理。

### 5.6.2 Web 服务的调用模型

Web 服务的调用模型如图所示, 服务的客户端首先生成基于 SOAP 的客户 UDDI 的客户实例, 通过 UDDI 的 API 接口, 调用 UDDI 服务的存储操作和查找操作, 实现 Web 服务的注册和发现。客户端是独立允许的应用程序、小门户程序或嵌入网页的组件代码。客户端对 Web 服务的调用包括静态调用和动态调用两种模式:

1) 静态调用: 首先生成 Web 服务的客户端代理程序, 然后将该程序嵌入到客户端应用程序中, 由服务代理通过 SOAP 的 API 调用 Web 服务的有关操作。这种模式要求必须事先了解所调用的 Web 服务的接口, 才能部署到应用程序之中, 而且一旦部署到应用程序之后, 就不能更改所调用的服务。

2) 动态调用: 事先不知道所调用的 Web 服务及其接口, 而是在客户应用程序执行过程中, 动态发现所需要的 Web 服务, 动态创建 Web 服务客户实例, 直接通过 SOAP 的 API 来调用 Web 服务的有关操作。

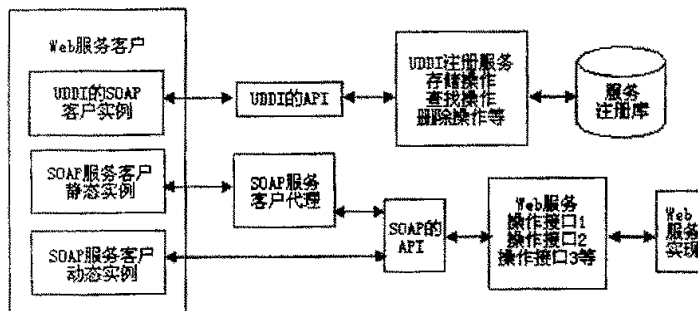


图 5-7 Web 服务调用 (运行) 基本模型

本文采用的调用模式是静态调用，首先在服务器端把 GIS 功能封装成一个服务，提供标准接口 (Interface)，定义类 (CLASS) 继承接口，然后设计所需的功能函数 FUNCTION 或者 PROCEDURE。GIS 服务是在 Web 服务器中实现的，在 Web 开发环境中编译服务，生成 GIS 服务的描述 (WSDL) 文件。在了解 WSDL 文档的 URL、接口、类信息的基础上，客户端应用程序通过引入一份定义服务的 WSDL 文档或者一份 XML 模式 Schema，解析 WSDL 文件中接口和继承类的信息，注册所需接口和类型的单元，将服务嵌入到本地客户端中。客户端一般对 Web 服务的远程应用产生可调用接口的静态链接，在该组件 URL 属性添加 Web 服务所在的虚拟目录。

### 5.6.3 实现举例

以距离测量为例说明服务器端函数的实现：

- 1) 先通过客户端脚本设置当前工具为 MapInfoWebDistance
- 2) 测量事件注册：

```
distance.DistanceComputed+=newMapInfo.Web.UI.WebControls.DistanceComputedEventHandl
er(distance_DistanceComputed);
```

- 3) 事件实现：

```
private void distance_DistanceComputed( object sender,MapInfo.Web.UI.WebControls.DistanceT
oolEventArgse)
{
string Msg = "测量距离为： "+ System.Convert.ToString(e.Distance*1.609) + "千米";
Response.Write("<script language = 'javascript'>"+ "alert("+ ""+Msg+ ""+ ") </script>");
}
```



实现界面如下图 5-8 及图 5-9:

1) 在界面上点选三点成一折线测量其间距离 (也可测量两点间距离)



图 5-8 测距点的选取

2) 客户端传回服务器端程序脚本调用服务, 计算结果显示如下:



图 5-9 测量结果

## 5.7 系统需要继续改进的地方

根据用户的需求及开发周期、工作量等方面因素的综合考量, 本系统完成了地图放大、缩小、漫游、打开及关闭图层操作等基本功能。除此之外, 系统还提供编辑维护、查询分析、专线电网/变电器定位、距离测量等功能模块。为用户提供了很大的方便。供电公司用电管理系统融入电力地理信息系统后, 通过 WEB 方式发布, 可使各部门的数据共享有一个质的飞跃, 使信息产业与管理工作的完美结合程度得到很大的提高。若作为一个大型的 WebGIS 系统, 还需要在以

下各方面进行加强。由于该系统依照标准搭建，具有极强的扩展性和可复用性，所以这些功能的添加并不影响系统的总体框架，只是具体实现方面的一些增补。

1) GML 数据的输出。系统目前向客户端输出的是 JPG 图片形式，这种方式应该说是适合所有能够使用浏览器的系统的。在一些需要客户端处理小量数据的情况下，需要把数据传送到客户端。不应该传送自己定义的数据格式然后再对其进行解析，而是传送标准的 GML 数据文件，GML 的输出应该参照 WFS 规范来实现。

2) 增加地理信息处理服务。地理信息处理服务包括空间分析功能、属性数据的查询功能以及独立的通用的地理信息网络处理功能(如投影变换、位置查询)等服务。

3) 基于 UDDI 的注册、发布。使得功能的使用不再局限于这个系统，而是所有的其它应用程序都可以根据自己的需要，查找到自己需要的功能，而不需要自己再实现这些功能。同时，系统的有些功能也可以从 UDDI 中查找并且使用。

4) 基于 Web Service 的应用程序的管理。包括数据的增加，删除，修改，编辑等功能。也就是实现 WCS 规范。

## 5.8 本章小结

本章探讨的是如何实现基于 .NET 的电网配电线路和设备的 WebGIS 管理，从而将电力 GIS 提升为一个在线的系统，赋予 GIS 系统新的生命力。该系统使得用户不但可以上网查询、浏览电网结构以及电力设备的有关属性信息，还可以随时随地根据需要对电网专线进行监视，可以方便、直观地监控全局运行情况，实现总体控制。GIS 数据的 Web 显示技术的突破，为电力系统计算机管理自动、准确、及时地提供更为详尽、丰富的数据，对于提高供电质量、用电可靠性和提高电力企业自身水平必将产生较大影响，同时也为基于电力网 GIS 在线分析计算提供了一种新的思路，奠定了一定的基础。

本章设计一个应用开发实例“基于 .NET 的电网配电线路和设备 WebGIS 管理系统”。研究了系统配置、开发工具、功能模块及 GIS 开发平台，系统主要包括两部分的设计：客户端的设计和服务器端的设计。本文采用的是瘦客户端的方式，客户端主要是界面设计以及与服务器交互两部分。应用服务器端则根据 Web Service 技术和 OGC 的实现规范，向客户端提供各种服务功能。

最后在总结系统的价值和意义的基础上，给出了这个实用系统可以继续加强和改进的地方，以便能够继续后面的研究。

## 第 6 章 总结与展望

### 6.1 本文总结

本文系统地分析了 WebGIS 的基本特征及实现技术, 研究 .NET 基本组成、技术构架和 Web Services 的定义、体系结构及运行机制。然后在分析 WebGIS 发展历程和传统的 WebGIS 应用开发中存在的问题的基础上, 指出基于 .NET 的 WebGIS 开发模式是软件体系结构发展的必须和这种开发模式对于解决传统 GIS 开发中遇到的问题有着重要的意义。

针对现有的 WebGIS 在分布式计算模式下的不足, 本文引入了 Web Services 的概念, 结合 OpenGIS 规范和 OGC 地理信息服务, 提出了基于 .NET 的地图服务平台的解决方案: 包括系统的基本配置和开发工具、功能模块划分, 地图服务平台的特点、优势和体系结构, 利用基于 .Net 技术和 MapInfo 的 MapXtreme 地图服务平台, 设计并实现了基于 .NET 的电网配电线路和设备 WebGIS 管理系统。本文主要的工作如下:

1. 分析了 WebGIS 的技术特点、WebGIS 的发展历程、WebGIS 在 WebServices 下的应用和现在 WebGIS 开发存在的问题, 指出 .NET 技术对解决 WebGIS 开发中遇到的问题有着重要的意义。

2. 分析了 .NET 的核心——Web Services 技术, 结合 OpenGIS 规范和 OGC 地理信息服务来提出一个基于 .NET 平台的 WebGIS 的解决方案。

3. 在 .NET 技术框架和 MapXtreme 地图服务平台的支撑上, 完成基于 .NET 的网络地理信息系统的设计、开发和测试, 并对该系统做出了适当的评价。

### 6.2 进一步的展望

正如计算机科学技术在不断地发展一样, 地理信息技术也在经历一个不断发展, 走向成熟的历程。本文对基于 .NET 的 WebGIS 系统进行了一定的研究和实践, 但还有许多值得研究和探讨的问题:

- 1、目前的系统设计实现了一些基本的地图功能服务, 今后应继续系统设计的开发工作, 使其功能更加完整。

- 2、地图服务平台的服务安全性、服务生命周期的控制、服务质量的管理等

方面也需做的进一步的工作。

3、目前基于 Web Service 开发的系统，一般都是在服务的地址和类型已知情况下的服务的静态链接，UDDI 正是为了解决服务共享共用而提出来的标准，目前新兴的网格服务 (Grid Service) 由于其本身就假定在资源动态变化的环境下的如何实现稳定的服务，所以在服务的动态生成、发现、动态使用方面，可能更适合网络环境。

## 参考文献

- [1] 刘南、刘仁义, 地理信息系统, 高等教育出版社, 2002, 10
- [2] 宋关福, 钟耳顺, 王尔琪. WebGIS—基于 Internet 的地理信息系统. 中国图像图形学 1998.3
- [3] 赵森生. Internet GIS 领域软件体系结构研究中科院遥感所博士论文, 2000.6
- [4] 赵森生, 杨崇俊. Web-GIS 的设计与实现. 中国图象图形学报, 2000, 5(1)
- [5] 宋关福. 组件式地理信息系统研究. 中科院地理所博士论文, 1998
- [6] 袁相儒, 陈莉丽, 龚建雅. Internet GIS 的部件化结构. 测绘学报, 1998, 27(4)
- [7] 罗英伟, 汪小林, 许卓群. 分布式 WebGIS 构件化研究. 计算机辅助设计与图形学学报, 2002, 14(8)
- [8] Bishr Y A, Pundt H. Probing the concept of information communities— A road towards semantic interoperability [EB/OL]. <http://www.ncgia.edu>. 1997
- [9] 黄裕霞, Cliff Kottman. 可互操作的 GIS 研究. 中国图像图形学报, 2001, 6
- [10] 黄裕霞. GIS 语义共享. 中科院地理所博士论文, 2000
- [11] 韩海洋, 龚建雅, 袁相儒. Internet 环境下用 Java/JDBC 实现地理信息的互操作与分布式管理及处理. 测绘学报, 1999, 2(28)
- [12] 高登凤, 杨冬青, 唐世渭. 元数据管理和空间信息共享, 中国地理信息元数据标准研究. 北京: 科学出版社, 1999
- [13] 丁力. Semantic Web 中空 1 } J 信息互操作框架研究. 北京大学硕士论, 2001, 60
- [14] 苏森. 面向对象的互操作技术. 电子科技大学博士论文, 1998, 6
- [15] 袁天凤. 论地理信息系统及其发展展望. 内江师范学院学报, 2002, 17 (6)
- [16] Open GIS Consortium Inc. Web Map Service Implementation Specification. <http://www.opengis.org/>
- [17] Web Feature Service Implementation Specification. Open GIS Consortium Inc. <http://www.opengis.org/>.
- [18] Microsoft 白皮书, Microsoft .NET 让新一代因特网变成现实. <http://www.yesky.com/200109201198001.shtml>
- [19] 李琦, 陈爱军, GeoML-地理空间信息共享和互操作模式研究[J], 测绘科学, 2000, 25
- [20] 蔡晓兵, 数据共享和互操作的新思路. 地理信息世界, 2003, 4, 1 (2)
- [21] 程永生, WebGIS—互联网地理信息系统的研究和实现, 大连理工大学硕士论文, 2002.3
- [22] 李慕华, 分布式 WebGIS 构件化的研究和实现, 北京大学硕士论文, 2000.5

- [23] 宋亚超, 阚国年, 张宏, 基于 Web Service 的 Internet GIS 集成与应用, 地理信息科学, 2004.3
- [24] 赵珺, 张明, WebGIS 实现技术分析及其操作模型, 计算机应用研究, 2004.6
- [25] 孟令奎, 史文中, 张鹏林, 网络地理信息系统原理与技术, 科学出版社, 2005.3
- [26] Mary Kirtland, 可编程网络: 网络服务提供了建立 Microsoft .NET 的基本模块, <http://www.microsoft.com/china/net/developer/programming.asp>
- [27] Kenneth E. Foote, Anthony P. Kirvan. NCGIA Core Curriculum in Geographic Information Science. [http://www.ncgia.ucsb.edu/giscclunits/u133/u133\\_fhtml](http://www.ncgia.ucsb.edu/giscclunits/u133/u133_fhtml)
- [28] 王兴玲、杨崇俊, XML 与新一代 WebGIS 系统的构建, 计算机工程与应用, 2002
- [29] Microsoft, Defining the Basic Elements of .NET, <http://www.microsoft.com/net/defined/whatis.asp>
- [30] 秦钢, 认识.NET My Services(二), [www.pcworld.com.cn/2001/back-issues/2124/2432c.asp](http://www.pcworld.com.cn/2001/back-issues/2124/2432c.asp)
- [31] .NET 全面接触, 微电脑世界, [www.pcworld.com.cn/2001/back-issues/2104/0405.asp](http://www.pcworld.com.cn/2001/back-issues/2104/0405.asp)
- [32] 邹建峰, 李律松, 李晓栓, ASP.NET 开发技术详解, 人民邮电出版社, 2005.1
- [33] 柴晓路. 架构 Web Service. <http://www.ibm.com/developerWorks/cn/webservices>
- [34] 微软技术教育大会—Teched 2001 技术文档, <http://www-microsoft.com/china/teched/ppt/default.asp>
- [35] Tim Bray, Jean Paoli et al., Extensible Markup Language (XML) 1.0 (Second Edition), W3C Recommendation 6 October 2000, <http://www.w3.org/TR/xml>
- [36] James Clark, XSL Transformations(XSLT) Version 1.0, W3C Recommendation 16 November 1999, <http://www.w3.org/TR/xslt>
- [37] [美] Jake Sturm, 开发 XML 解决方案, 北京: 北京大学出版社, 2002, 15~24
- [38] Ashish Banerjee, Aravind Corera, C# Web Services-Building Web Services With .Net Remoting and ASP.NET
- [39] Marck Birbeck 著, 裴剑峰等译.XML 高级编程.北京:机械工业出版社
- [40] XML 1.0 REC. <http://www.w3.org/TR/REC-xml>
- [41] SOAP 规范. <http://www.w3.org/TR/SOAP>
- [42] Erik Christensen, Francisco Curbera et al., Web Services Description Language (WSDL) 1.1, W3C Note 15 March 2001, <http://www.w3.org/TR/wsdl>
- [43] OpenGIS Geography Markup Language (GML) Implementation Specification. Open GIS Consortium Inc. <http://www.opengis.org/>.
- [44] 柴晓路, UDDI 服务实施的体系架构. <http://www.uddi-china.org/research/paper/uddi>

- [45] UDDI 规范. <http://www.uddi.org>
- [46] OGC Web Services Thread Set 2 (OWS1.2)-Architecture, March 2002. [http://opengis.org/OWS1.2/docs/020314\\_OWS1.2\\_Annex\\_B.doc](http://opengis.org/OWS1.2/docs/020314_OWS1.2_Annex_B.doc)
- [47] Roberto Chinnici, Markin Gudginetal, Web Services Description Language (WSDL) Version 1.2. W3C Working Draft 9 July 2002. <http://www.w3.org/TR/wsd112/>
- [48] Jean-Jacques Moreau, Jeffrey Schlimmer, Web Services Description Language (WSDL)Version 1.2: Bindings, W3C Working Draft 9 July 2002, <http://www.w3.org/TR/wsd112-bindings/>
- [49] John Miller. Web Services: What, Why, and How. <http://www.microsoft.com>
- [50] Open GIS Services Architecture. <http://www.opengis.org/>
- [51] 陈常松, 何建邦, 面向 GIS 数据共享的概念模型设计研究, 遥感学报, 2003.9
- [52] OpenGIS Consortium. GML3.0. <http://www.opengis.org/techno/documents/02-023r4.doc>
- [53] 张梨等, 互联网时代的地理信息系统, 测绘学报, 1998, 27 (1)
- [54] 赵雷生、杨崇俊, 基于网络环境的地理信息系统整合和知识发现, 中国图象图形学报, 1999, 4 (11)
- [55] 赵雷生、杨崇俊, Web-GIS 的设计与实现, 中国图象图形学报, 2000, 5 (1)
- [56] 郭杰华等, 基于 Internet 的地理信息系统 (WebGIS) 的研究和开发, 中国图象图形学报 2000.7
- [57] 徐寿成, 王行风, XML 与 WebGIS 的空间数据管理技术, 计算机应用研究, 2001, 18(12)
- [58] 张颖, 基于 XML 的地理信息元数据系统, 北京大学硕士学位论文, 2001.6
- [59] Open GIS Consortium. Open GIS Reference Model. 2002
- [60] Open GIS. Web Map Service. <http://www.opengis.org/techno/specs/01-068r3.pdf>
- [61] <http://terraserver.homeadvisor.msn.com/about.asp?W=0>
- [62] <http://www.dotnetmap.com>
- [63] 李琦, 杨超伟, 陈爱军., Web GIS 中的地理关系数据库模型研究, 中国图象图形学报 2000. 5(2)
- [64] FGDC. Content Standard for Digital Geospatial Metadata, Metadata Ad Hoc Working Group, Federal Geographic Data Committee. 1998, <http://www.fgdc.gov/>
- [65] USGS. SDTS(Spatial Data Transfer Standard), <http://mcmweb.er.usgs.gov/sdts/>
- [66] 刘南, 刘仁义, Web GIS 原理及其应用—主要 Web GIS 平台开发实例[M].北京:科学出版社, 2002
- [67] 柴晓路, 梁宇奇 编著, Web Services 技术、架构和应用, 电子工业出版社, 2003.1

- [68] 王毅, ASP.Net 1.0 高级编程, 清华大学出版社, 2002
- [69] 丁超, 张鹏, ASP.Net/XML 深入编程技术, 北京希望电子出版社 2002
- [70] Thuan L. Thai, Hoang Lam. .Net Framework Essentials. O'Reilly 出版社, 2002



## 致 谢

光阴荏苒，岁月如梭，转眼间三年的硕士研究生求学生涯已近尾声。回顾三年来的学习生活，无限感慨，借此机会向在我学习和生活中给予关心、帮助的老师、同学、家人表示衷心的感谢。

首先，我要深深感谢我的导师杨春金副教授在我个人发展上的指导、关心、支持。无论是在学习上、还是在生活上，杨老师对我进行细心的教导和至诚的关怀，无倦的教诲使我终身收益、永生难忘。论文的选题到各个阶段都得到了杨老师的精心指导和大力支持。在此谨向杨老师致以由衷的感激和深深的谢意！

此外，我也衷心地感谢王虹、杨杰、阙大顺、沈维聪，孟哲、肖攸安、胡君平、阮军老师这三年来在学习上、生活上对我的鼓励、支持和帮助。

感谢 03 级硕士研究生同学，郭玖琳、李景松、王平、柯栋、郑国栋、李佐斌、朱群英、倪福银的热心帮助，文中的许多思想来源于平时与他们的交流和讨论。还要感谢 04 级、05 级的师弟师妹们和我们共同营造了一个严肃活泼、团结向上的学术氛围。有了大家的同心协力，我们才能顺利完成研究工作。

深深感谢养育我的父母，感谢他们长期以来对我的关心和鼓励。感谢他们支持我、督促我继续学习和研究。

再次衷心感谢所有关心和帮助过我的人。

詹琼

二零零六年四月

## 攻读硕士学位期间发表的论文和参与的项目

### 1. 发表论文

- [1] 杨春金, 詹琼, 基于 GIS 平台的高速公路机电设备管理系统设计, 智能交通, 2005.5
- [2] 杨春金, 詹琼, VPN 技术在高速公路信息管理系统中的应用, 中国水运, 理论版, 2005.9

### 2. 参研项目

- [1] 襄十高速公路车辆查询系统, 负责项目调试工作。
- [2] 黄黄高速公路机电设备管理系统, 负责项目开发, 调试工作。
- [3] 北流供电系统配电线路和设备管理系统, 负责项目开发, 调试工作。