Implementing Internet GIS with Java based Client/Server Environment

M.N.Babu,
Beach Sand and Offshore Investigations,
AMD, DAE, Thiruvananthapurm, INDIA.
Phone: 0471-2345185
e-mail: mnbabuin@yahoo.com

Key words: Internet GIS, Client/Server, Java, RDBMS

Abstract

Geographic information system (GIS) has become an indispensable tool for analyzing and managing spatial data. The rapid growth of Internet provides highly customised, accessible and interactive sources of information and has created huge demand for applications that use geographic information system. The broad use of fast, up-to-date, reliable maps on the Internet drive GIS industry for providing continually improving quality of information. Internet GIS technology provides dissemination, sharing, displaying and processing of spatial information over the Internet. Internet based geographical data services mainly involve sharing and management of spatial and non-spatial data. Data pertaining to spatial attributes in a GIS can be efficiently managed with any relational database management system (RDBMS). Databases offer great opportunities to improve individual and organizational productivity to enhance the quality of decision making. When building GIS application for the Internet, developers can choose any number of architectures, technologies and methods to deliver maximum information. Java's open, platform-independent, object-oriented design is an ideal platform for mapping and geographic information system (GIS) solution for developers. eXtensible Markup Language (XML) standards can be used to describe and exchange map data between applications. This paper describes the salient features of a spatial database that was developed by integrating thematic layers stored as feature datasets in MySQL-Relational Database Model into a Java based Client/Server environment. The map data is handled by a set of Java Servlets and user interactivity is made through a Map-Applet which can be downloaded to any client machine by making request to the map server. The system described here aims to provide information on implementing a Client/Server based Internet GIS through a Web-based platform.

Introduction

GIS is an emerging technology encompassing many disciplines namely geography, cartography, photogrammetry, remote sensing, surveying, GPS technology, statistics and other disciplines concerned with handling and analysing spatially referenced data. GIS is mainly comprised of data handling tools for storage, retrieval, management and analysis of spatial data as well as solving complex geographical problems. GIS can also be used for the generation of new information by the user-defined combination of several existing information. Because of these various distinguishing

features of GIS it is considered as an indispensable tool for conducting spatial searches and overlays and association of the spatial data with the non-spatial attribute data to eventually generate useful information.

The Internet is client/server based where the client sends a request for service and the server processes the request and returns information to the client. The Internet, unlike other Client/Server based networked applications, is constrained by the network size, speed and administration. The Internet is not just a computer network, but it is based on open standards that define low-level communication protocols such as transfer control protocol/Internet protocol (TCP/IP), state distribution protocols such as file transfer protocol (FTP) and hypertext transfer protocol (HTTP), document content standards like hypertext markup language (HTML), extensible markup language (XML) and image formats like GIF and JPEG.

The integration of GIS and Internet technologies is allowing easy access to information and data without using any specific GIS software. The growing capability of the Internet has created increasing demand for applications that use GIS. The Internet makes sense for an organization centralizing the maintenance of GIS software services and data. Using the Internet, software can be easily updated, and users can gain access to the applications and information they need for specific tasks.

Server and Client side Architecture

There are two basic approaches for deploying GIS or any other complex, data-driven application on the Internet; those are server-side and client-side applications. In a server-side Internet GIS application, a Web browser is used to generate server requests and display the results. An Internet GIS server usually combines a standard Web (HTTP) server and a GIS application server, and the GIS databases and functionality reside completely on the server(s). A server-side GIS application can be illustrated by a mapping application on any of the major Internet portals. Users type in the address they are looking for (the request), which is transferred to a Web server. The Web server passes the request to a GIS application server, which runs an address matching routine, generates a map graphic, convert the graphic to Web format, wraps the image in HTML and sends it back to the Web server, which then returns the response to the client as a standard Web page. Server–side application can comply with Internet standards, because all the large GIS databases remain on the server (Gifford, 1999). This is an advantage for simplified application development, deployment and maintenance of data. In server-side Internet GIS, usually the application requires proprietary software and the software stay on the server. Map data transmitted to a Web client are in standard HTML formats that can be accessible through any Web browser, creating significant positive implications for performance, reliability and size of user base.

In a client-side Internet GIS application, client-side applications require users to install a complete client application. In either case client-side application require software of some kind (other than browser) to be transferred to the user. In client-side Internet GIS, the client is enhanced to support

GIS operations. To implement client-side solutions of any kind, software must be transferred to the client. The primary advantages of client-side solutions are the abilities to enhance user interfaces, improve performance and implement solutions using vector data. Client-side solutions can be implemented with all the features and capabilities allowed by a modern GUI (graphical user interface). Multipoint feature selection, selecting an area by dragging a window and a modal cursor operations controlled by clicking on menu icons as capabilities available in client-side products that aren't available in server-side implementations. When client-side solutions use vector date structures, a large set of graphic and GIS operations can be performed locally. A significant performance advantage is that window display changes can happen without retransmitting a request to the server. Core GIS operations like editing, buffering, overlay analysis and route tracing are examples of operations that are enhanced by exploiting vector data structures on the client side.

An example of the client-side Internet GIS application is one that runs as a Java applet. The code or the applet is transferred to the Web browser as binary instructions that provide a GUI for the GIS application. Vector-based data is then transferred to the client enabling the complex GIS functions on the client. The architecture should not be confused with a similar server side GIS application that implements a Java applet to create a GUI for the GIS application. In this case, the applet is simply an interface and the complex GIS calculations and data remain on the server (Marshall, 2002).

Disadvantages of server-side solutions are primarily associated with limited user interface and poor performance. In GIS, there are many applications in which users need to drag windows or select multiple features before an action takes place. It is difficult or impossible to do this by using a standard Web interface. In addition, server-side application must retrieve all the application information from the server for every operation. For example, when users reset a display window by panning or zooming, turn a layer on or off, or take any other action, the request must be sent to the server and returned. The results in many requests, which can cause poor performance.

The disadvantages associated with client-side solutions relate to distributing software and data. By definition, when companies implement client-side Internet GIS they must somehow distribute software to users. Distributing software (Java, ActiveX or any other type) is problematic. Some potential users will be lost due to platform incompatibilities, some users will undoubtedly have difficulties loading software, and providing technical support is time consuming and expensive.

Standalone Internet GIS applications (ALOV, 2002) are not purely client-side Internet GIS applications in which the system is independent of the working platform, application software, server environment and browser software. Such systems are file based and easy to develop and can be deployed in any Java enabled Web server. The limitation of the system is its slow performance and lack of advanced features available with them.

Integration with Java and XML

Java™ is increasingly adopted by developers and IT professionals as the preferred software development/ deployment platform for Internet applications (ESRI, 2002). The idea of Java (write once, run anywhere) is incredibly powerful especially for Internet applications. Users must remember that Java is a proprietary language and not an Internet standard. Java is characterised in terms of a list of buzzwords: it is a simple, object-oriented, distributed, interpreted, robust, secure, architecture neutral, portable, high-performance, multithreaded and dynamic language. Java will be great but not until the kinks are worked out and is endorsed as an open standard by one of the appropriate standards organizations.

A Java application does not run on a particular machine; it runs in an environment known as Java Virtual Machine (JVM). Each platform has its own such environment, which translates the "independent" functionality of the Java application to machine specific function calls. Though the application can be written once and run it on many platforms, a Java application may run much more slowly than a native one. Fortunately, operating system developers and browser developers have created Just-In-Time (JIT) compilers to speed up Java applications by compiling them into native applications before they are executed. JIT compatibility is built into many browsers and operating systems.

The JDK 1.2 forms the basis of what Sun calls the Java 2 platform. This is Sun's first implementation of a complete set of Java programming building blocks, the Java Foundation Classes (JFC). JDK 1.2 is seen as the first functional release for Java, and it has attracted the notice of non-Java developers also. What has caught the attention of many people is Java's ability to generate applets, small programs, which can be downloaded from the Internet and run on any Java enabled Web browser. This has been used to add dynamic features to Web pages which would be quite impossible to achieve with previous technologies.

Java applets are small programs which can be incorporated into a Web page. The executable code of the applet is compact and is downloaded along with the text of the page. Once the code arrives from the server it is executed by the browser on the local machine. This allows the programmer to do many things which would be impossible in a normal Web page. Among other things the applet can display dynamic graphics on the Web page, it can allow the user to interact with it and it can connect back to the server to download further information or query a database. The infrastructure required to support the applet is built into the Web browser. Java applet will run on any platform: the environment provided by the browser insulates the applet from the underlying operating system and allows it to function on any platform on which a Java-enabled browser is available. This is obviously a key feature in allowing use over the Internet. All of the standard libraries for such things as the graphical user interface and network access are provided on the client platform. The only code which needs to be transferred across the Internet is that which is unique to the applet.

Java Servlets are dynamically loaded modules run inside the JVM and service requests from a server. Because the servlet is running on the server side, it does not depend on browser compatibility. Servlets can be used for any number of Web-related applications. A servlet can handle multiple requests concurrently and can synchronise requests. Also, it can forward requests to other servers and servlets. Thus servlets can be used to balance the load among several servers. Servlet can communicate a user through an applet interface embedded in any HTML page via HTTP. Figure 1 depicts the execution of Java servlets and communication with applet via HTTP.

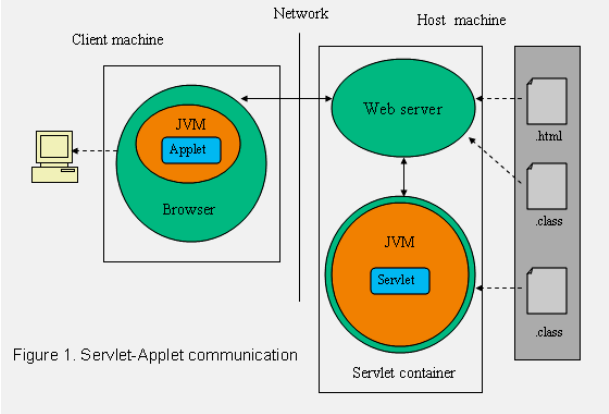

Figure 1. Servlet-Applet communication

For developing Web pages there are ever expanding numbers of hypertext markup language (HTML) tags that can be used to format the way items look on pages. The eXtensible Markup Language (XML) has been developed to compliment HTML, taking over from simply defining what data looks like to describing what it actually is. Whilst XML uses the same tag structure as HTML, an XML element can declare its associated data to be an address or post code, a point value with associated attribute data or any other desired data element. The intrinsic flexibility of XML can provide a number of tangle benefits to users. XML can allow integration of data from disparate sources, with middleware applications pulling data from different databases and translating it into neutral XML for client-side processing. An XML document can contain descriptions of data which can be used in multiple applications, each application using the specific tags, and delivering users appropriate view for their particular task.

Integration with Relational Database Management System (RDBMS)

Databases are becoming an increasingly important feature of GIS and an increasingly valuable asset. Relational Database Management System (RDBMS) provides number of functions critical to the effective operation of a GIS and it is developed to manage and share data in an orderly manner and to ensure that the integrity of the database is maintained. Thus the value of user's data can be increased greatly by the use of a Relational Database Management System (RDBMS). RDBMS enables fast storage and retrieval of large amounts of information. The interface that is used to insert, manipulate and extract data is called the Structured Query Language (SQL).

The database for the Internet GIS explained in this paper consists basically of two types of data: tabular data stored in a relational database (dbf) and ESRI shape (shp) files stored on disk. In this model, the map and attribute data are removed from the shape and dbf files and placed them in

MySQL relational database. Thus the map features are stored in a clearinghouse as SQL based database (MySQL) according to OpenGIS® standards (ALOV, 2002). Clearinghouse is a container which contains the list of datasets accessible via server side and their metadata. The clearinghouse can be either XML document or a set of tables in SQL database. This clearinghouse consists of tables that hold all the available attribute data and the attribute table includes fields that can be used to store binary data (BLOB). This model is more flexible and allows incremental downloading of data. The server spreads vector maps to client in the most efficient way and uses advanced caching to reduce network traffic. Other leading RDBMS like Oracle, MS-SQL Server, Interbase or MS-Access can also be used to store and manage layers and attribute data.

Design and implementation of low-cost Internet GIS

Designing and implementing Internet GIS with freely available software tools can be successfully done with little care. To obtain map data in standard formats we have to depend on commercial products like ArcView, MapInfo etc. Therefore, small investments are required for the generation of basic GIS data, but the cost will be almost nil by using by using products like ALOV Map (free Internet GIS software), the Java Development Kit (JDK), Tomcat Java Web Server, MySQL RDBMS etc for developing a Client/Server based Internet GIS. ArcView software is used to generate shp and dbf files and that is the only commercial product used here.

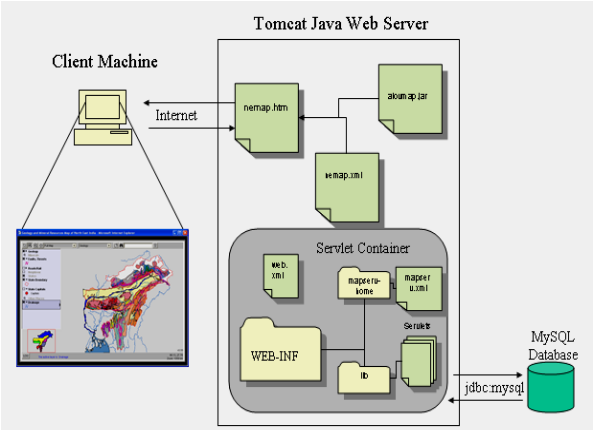

Figure 2. Internet GIS with Java Client/Server environment

ALOV Map is free portable Java® application for publishing vector and raster maps on Internet and interactive viewing through Web browsers. It supports the complex rendering architecture, the unlimited navigation and allows working with multiple layers, thematic maps, hyperlinked features and attribute data. Figure 2 illustrates the structure of the system comprised of ALOV Map, Tomcat Java Web Server and MySQL relational database. Map data transfer is controlled by a set of servlets in Tomcat Web Server and a map applet at the client end on request via HTTP. The servlets available in the Servlet container are: MapServlet - main servlet, ContextListener - allows to reload all objects on server, WMSServlet - implements Web Map Service and UploadServlet - upload shp and dbf files to SQL database and creates metadata and register the datasets. The following are the six simple steps to build a low-cost Internet GIS.

Table 1. Simple steps to build ALOV Map based Internet GIS

Step 1:	Creation of map layers (shp) and attribute tables (dbf)
Step 2:	Creation of database clearinghouse in MySQL
Step 3:	Creation of map folder in Tomcat \Webapps directory and placing the required files (Java .class & other files) of ALOV Map.
Step 4:	Pumping of map data (shp and dbf files) to map database in MySQL using UpLoadServlet. It creates a project xml file automatically.
Step 5:	Editing of the project xml file with required parameters
Step 6:	Testing the program

Map window, legend, buttons like zoom out, zoom in, pan, web link, feature selection, info, text boxes for selection domains, themes, query and help are available with the applet for user interactivity. Examples for query results are given in figure 3. Maps are shown with decimal geographic co-ordinates and a zoom scale is given at the right bottom end of the map panel in the applet. Web links have been given for some layers to be connected with related web sites. The flexibility of the system is that modification or updating of map data can be done at any point with the help of UploadServlet interface.

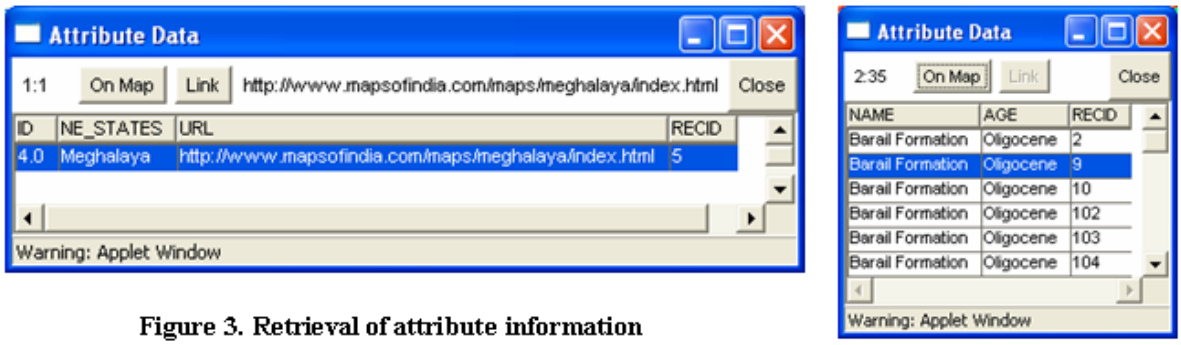

Figure 3. Retrieval of attribute information

Conclusion

The Internet is redefining the collection, management and dissemination of spatial information. In this new Internet computing environment, spatial applications are moving from the desktop to database servers where they can be professionally managed and accessed through simple Web browsers. The rapid growth of the Internet is having profound impact on GIS industry, bringing GIS out of its specialized niche into the broader information technology market. Internet offers increased dissemination and use of geographic information in all walks of life. The Java platform allows developers and business users to create exact application that run over networks. XML is a generalized way to describe arbitrary data types independent of how the data are used or presented to users. Databases are becoming an increasingly important feature of GIS since it allows many users to access large amount of data through computer networks or through the Internet. Small footprint

Internet GIS applications like ALOV Map and other free server based applications enables Client/Server based computing with less investment of money and other resources.

References

- ESRI, (2002),. Geographic Information System for the Java Platform, An ESRI® White Paper, November 2002, p1-8. http://www.esri.com
- Gifford, Fred (1999),. Internet GIS Architecture, Client v Server – Which Side is right for you, Mapping Awareness, V.13, No.7, pp 40-42.
- Marshall, Johnny., 2002. Developing Internet-Based GIS Applications, GIS India, V.11, No.1, pp 16-19.
- ALOV, (2002),. ALOV Map, Free Java GIS, http://www.alov.org.