iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 2 Issue: VII Month of publication: July 2014

DOI:

ISSN: 2321-9653

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

Web GIS Architecture Based on JNLP & some Performance Improving Techniques

Barkha Narang [1], Meeshika Arora [2]

1. Assistant Professor , Jagannath International Management School, New Delhi-110019
2. M.Tech Student , Banasthali University, Jaipur

Abstract- ***WebGIS (also known as web-based GIS and Internet GIS) denotes a type of Geographic Information System (GIS), whose client is implemented in a Web browser. By analyzing the defects of two tier and three tier client/server Web GIS architecture, this paper presents four tier client/server Web GIS architecture based on JNLP. This kind of Web GIS architecture can solve the problem of client side software publish and maintenance, and improve efficiency of Web GIS.***

WebGIS have been developed and used extensively in real-world applications. However, when such a complex web-based system involves the dissemination of large volumes of data and/or massive user interactions, its performance can become an issue. This paper also presents several major potential performance problems with WebGIS and several possible techniques to improve the performance. These techniques include the use of pyramids and hash indices on the server side to handle large images. To resolve server-side conflicts originating from concurrent massive access and user interactions, we suggest clustering and multithreading techniques. Multithreading is also used to break down the long sequential, layer-based data access to concurrent data access on the client side. Caching is suggested as a means to enhance concurrent data access for the same datasets on both the server and the client sides.

Keywords- ***Client/Server; Web GIS; JNLP; performance; pyramid; hash index; multithread; cluster; cache; dynamic data request.***

I. INTRODUCTION

Rapid progress of Internet, application of data warehouse technology and the combination of GIS and Internet have paved the ways for the web distribution of GIS spatial data, it also makes the GIS space data management and application appear the characteristic of multi-user, distribution and web. Internet has already been one of the important developing platforms of GIS, the function of GIS has been extended by Internet, and furthermore it becomes a kind of new GIS architecture.

As regards for the web architecture of Web GIS, it is mainly divided into two tier and three tier client/server mode in recent years. Two tier client/server models has a lot of defects, with the large-scale development of GIS application system and the users' demand for system capacity, these defects and weakness confine the development of GIS application, but three tier client/server mode has been applied step by step depending on its distinctive advantage. This paper presents four tier client/server Web GIS architecture based on JNLP .

While Web GIS is gaining in popularity, dissemination of voluminous and heterogeneous data becomes a challenge, as the Internet bandwidth is not limitless. To handle this challenge, two important issues should be considered: (1) share and interoperate the heterogeneous data among different systems,

Vol. 2 Issue VII, July 2014
ISSN: 2321-9653
INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

different communities, and different users and (2) improve the system performance so that data are delivered to the users within a reasonable time span.

This paper is organized into several sections. In section 2, we review architecture of web GIS . In section 3, we discuss several performance improving techniques like- Pyramid and Hash Indices, Multithread and Cluster Techniques.

II. FOUR TIER CLIENT/SERVER WEB GIS ARCHITECTURE BASED ON JNLP PROTOCOL

A. JNLP Protocol Introduction

JNLP (Java Network Load Protocol) is a kind of program distribution technology, platform- spanned, safe and strong application , newly distributed by SUN corporation. JNLP can distribute application program with full function to client side, meanwhile, guarantee these program to be newest version. JNLP is realized by Java Web Start (figure 1). Java Web Start is installed at client side, but server still is standard Web server. Java Web Start accounts for the installation and maintenance of client side software. When client side needs for GIS application system, Java Web Start automatically checks if he has executed the program before. If the program has already been in the system Cache, then executes it directly. If not or newer version of this program exists, Java Web start downloads the newest version of it in order to client side software is the newest version. Although the initial download speed is little slow, all files slowly store on local disk of computer after downloading. Despite the fist activation cost of application program is more than that of HTML web page, nevertheless running application program is nearly prosecute immediately, for resources we needed are in local computers. From this point of view, application program not only get rid of web browser so as to run directly, but also cooperate with the web page start-up, it is so called get numerous results at one stroke.

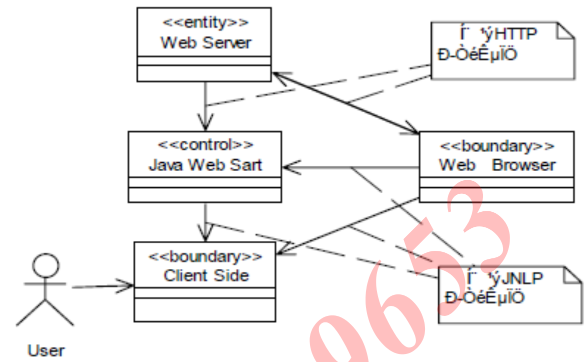


Figure 1 JNLP Java Web Start architecture UML Class diagram

Java Web Start is a kind of solving scheme facing Web application program. Using GIS client side application on program substitute for client software based on HTML, it makes user interface with high interaction at client side, it can compare with traditional application program (eg Word and Excel etc.), it demands low web bandwidth; GIS client side program need not join with Web server when click every time, it can slowly store the information downloaded. Thus Java Web Start can realize better interactivity with low connection and bear usage of offline. At the same time, client side software web security design is very consummate, following sandbox security. It can prosecute resource access with authorized permission once pass through security authentication, nevertheless virus problem does not exist.

B. High Efficiency Web GIS Architecture Design Based on JNLP Protocol

Four tier client/server Web GIS architecture structure based on JNLP protocol is mainly designed for solving the defects of three tier architecture, it consists of client level, top Web service level, GIS transaction level and database level. Compared with three tie client/server architecture, it adds first level top web service level. This top web service level accounts for application program maintenance at client side and the slow storage of static data. Its architecture structure shows as figure 2

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

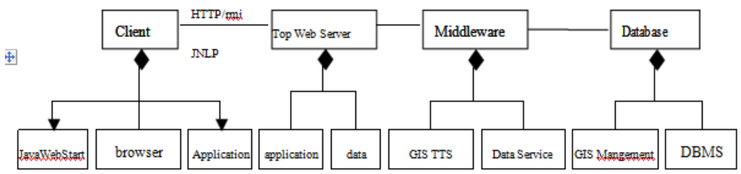


Figure. 2 Four tier client/server Web GIS architecture UML Class diagram

the protocols used between client level and top web service level are http, JNLP and RMI three kinds. By Java Web Start protocol, client level downloads and installs the Java application program, which stored at the top web service level. After acquiring Java application program at Web GIS, client joins with GIS transaction logic level server by TCP/IP protocol, server depends on the requirement of client, acquires data from database level, then transmit the data to client side application program, client side extends class package by JNLP protocol and makes the acquired data down in local Web Start cache, so as to make the client side use the Web GIS data content he concerns in the state of offline.

As regards as the architecture structure, top Web server level of four tier client/server Web GIS is in the state of activity only when the client side has not installed GIS client side software, in case client acquires GIS client side software, top web server level is not in the state of activity, four tie client/server architecture dynamically switches into three tie client/server architecture. Because of adopting separation of application program and GIS transaction logic function level, the Java Web Start pattern management at client side, it makes the client side software have the ability of self-installation, updating and repairing, the complicated computing work and display function are settled at client side, thereby the server and client side balance the load, client side may run at the state of offline and browse the spatial data content which has been displayed, reducing the unnecessary web transmission quantity.

III. PERFORMANCE IMPROVING TECHNIQUES

A. *Pyramid Technique & Hash Index*

Pyramid and hash indices used for handling large image data management and transmission problems. On the client side of a WebGIS, hardware specifications often restrict the size of an image viewed by users. For example, modern desktops and laptops use 10246768 or 128061024 resolution with 15–21 inches of display. In order to provide faster access to different parts of the entire image from the restricted window, we propose using both the pyramid and cut-and-hash indexing technique.

A pyramid, also called a hierarchy, is constructed by generating different abstraction levels of the original data through resampling. A pyramid consists of multi-scale replicated images of the original one. When a client requests data at a given scale, the server will search the required data from the level which has a scale closest to that requested, instead of searching the original high-resolution data directly.

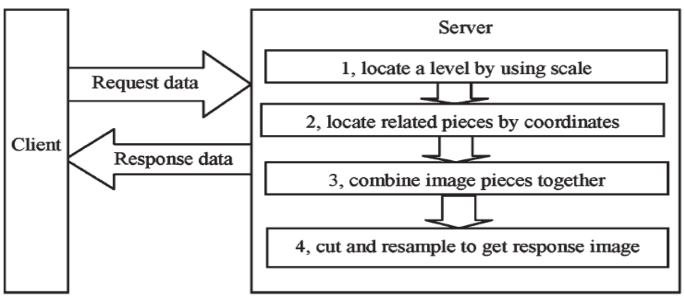
The use of the pyramid technique in this manner can reduce access time by extracting data from a smaller cartographic scale image already stored in the pyramid instead of the original one, especially when there is a large difference between the requested scale level and the original scale. When more detailed data are needed, or when it becomes necessary to access the original image, a better access speed can be achieved by accessing the smaller piece of the original data, if the original data are cut into smaller pieces. In this circumstance, a restricted area of the image, instead of the entire image, is accessed. If the smaller piece images are managed and accessed efficiently with effective strategies, the performance of the system will be improved tremendously.

The procedure for accessing a pyramid is critical. Given that a pyramid is built for an image, four steps are involved in accessing the data. The first step is to determine the target level at which the data will be extracted by comparing the requested scale with different Performance-improving techniques in web-based GIS 325 scales already stored in the pyramid. The second step is to find related pieces at the target level by using coordinate information. The third step is to combine these related pieces to form an image, which may have a spatial extent

Vol. 2 Issue VII, July 2014
ISSN: 2321-9653

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

larger than the requested area. The final step is to cut the combined image to match the requested area and, if necessary, resample the cut image to the requested scale level. These steps are illustrated in figure 3.

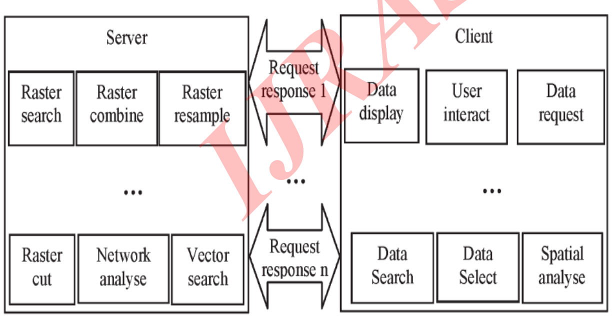


Procedure for data access using the pyramid and cut method.

B. Cluster & Multithread

Cluster & Multithread are used to deal with the problem of concurrent access and to improve client-side interactive performance. A WebGIS should provide an efficient environment for clients and servers to communicate intensively such that services requested by the users can be completed within a reasonable time. The communication between servers and clients, and related GIS function components in each side of the network are illustrated in figure 4.

Traditionally, the procedure for processing a user's request involving multiple data layers, raster and/or vector, takes the following steps: (1) When the client



Processing functions in WebGIS.

receives a request from the user, the client performs a spatial search on the client side to identify which spatial datasets are required. (2) Within the selected data extent, a thorough search by data layer is performed to check whether the client side has the requested data layer cached. If the client side does have the data layer, then there is no need to fetch the data from the server. (3) If the client side does not have the requested data layer, the client will then submit a data request to the server, asking for the requested data. (4) The server will conduct a spatial search to locate the data. (5) After identifying the data layer, the server sends the data back to the client. 6) The system returns to the second step until all layers are found and transmitted. If a traditional single-thread process is adopted, a user cannot initiate a subsequent layer-data request before the former-layer data are received. If the request involves many searches and data layers, the process will become very time-consuming and may not be completed within a reasonable response time. This in turn will be a great frustration to users. Furthermore, when many clients access the same server simultaneously, the sequential process described above will be too slow to respond adequately to requests from multiple clients. The increasing

In addition, each server can use multithread techniques to process concurrent client Requests. When the client sends requests to the port server connected to the Internet, it channels all the process requests and responses. The port server can respond to more than one request at a time by allocating one thread to one request with multiple threads handling multiple requests. The port server will redirect requests to another multithread server for processing when the first multithread server reaches its capacity. A WebGIS server should be equipped with flexible capacity to serve concurrent requests by using cluster and multithread system configurations on the server side. Given this system architecture, performance of the WebGIS can be maintained or enhanced by upgrading the computers or adding a number of servers to meet the growth of concurrent accesses as needed. Therefore, this is a scalable solution that can accommodate the future growth in demand.

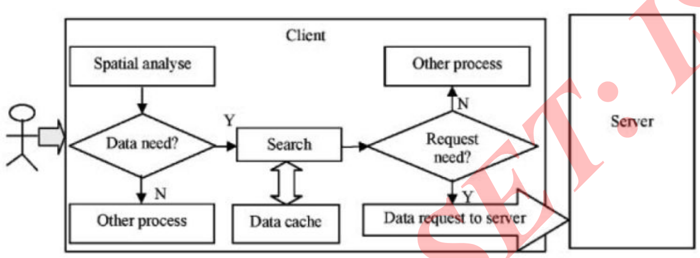
Vol. 2 Issue VII, July 2014
ISSN: 2321-9653

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

On the client side, a user often issues a request that involves many processes and accesses, and they can be accomplished sequentially. However, some of these processes or accesses could be performed independently without affecting each other. Then, the multithread technique can also be used to improve the performance on the client side. Using the multithreading technique can reduce the access time to possibly 1/10th of the required sequential access time in this specific example. Therefore, we adopt multithreading technique to send requests for different raster and vector layers simultaneously to the server.

C. *Caching & Dynamic Data Request*

Caching and dynamic transmission methods are used to improve the interactive performance. As illustrated in figure 6, when a request is issued, a WebGIS client will first examine whether the requested data are on the client. Then, the data request will be sent to the server only if the data are not on the client side. If the data are on the client side, the access will, of course, be much faster compared with the access across the



Cache and dynamic data request.

Internet from the server. Having some data residing temporarily on the client may also meet the need of subsequent data requests. Therefore, it will increase the efficiency of data access if some data are kept on the client by caching them in order to reduce the number of client requests sent to the server. Whenever data are needed, the client will first check whether the data are already available on the client side before sending a new request to the server. In a dynamic request and caching system, two issues need to be considered: what content should be cached and for how long? Caching data on the client side can reduce the load on network transmission and server processing time, and thus improve the overall system performance.

Ideally, data are required to be sent only once if all the data are duplicated on the client side. But this is impossible in most cases because the data sets may be too large to be completely cached on the client side. Therefore, based upon the user's interest or the frequently accessed data as reflected by coordinates, spatial data defined by these coordinates are cached for a certain duration before they are replaced by other data, which meet the more current needs of the user. Frequently needed data should be cached to allow the system to respond as quickly as possible. Whether this technique will be effective or not is largely dependent on the specific applications. In general, basic data layers with a relatively small volume, such as the county boundary data for a state, should always be cached as a static layer. It should be loaded first and only once. Some frequently used data which are larger in volume, such as the district boundary data of the entire world, could be cached as needed. Some data which are not frequently used and have a small volume, such as the school data inside a county, could be cached once they are loaded. If a data set is not important (which has to be defined by the specific application environment) and with a large volume such as image background data, these data should not be cached. These scenarios are summarized in table 1 and can be implemented as options in a flexible function in the setup of a WebGIS. The system administrator or developer can customize the caching strategies based upon the data volume and characteristics of the data in the specific applications.

IV. CONCLUSION

The four tie client/server Web GIS architecture base on JNLP protocol solves the defects of three tie client/server Web GIS architecture, offers the new technology of spatial client side application program, makes full use of hardware resources of server and client side, enhances the interaction of system, reduces the web data transmission. This architecture structure will become the one new direction of Web GIS architecture.

ISSN: 2321-9653

INTERNATIONAL JOURNAL FOR RESEARCH IN APPLIED SCIENCE AND ENGINEERING TECHNOLOGY (IJRASET)

This paper discussed the use of several methods in enhancing the performance of WebGIS. We also demonstrated that using these techniques increases system performance substantially. The pyramid-hash index improves the efficiency of publishing large images. Cluster and multithread methods can increase the efficiency to handle massive concurrent accesses to the servers. Cache and dynamic data management are used to improve client-side interactive performance.

REFERENCES

[1] Chaowei (Phil) Yang Corresponding authora, David W. Wonga, Ruixin Yanga, Menas Kafatosa & Qi Lic International Journal of Geographical Information Science, Volume 19, Issue 3, 2005

[2] Li Luqun[1], Li Jian[2],Tian Yu[1,] Computer Center of Shandong University of Science and Technology, Tai'an, China.

[3] Li Luqun[1], Li Jian[2],Tian Yu[1] Computer Department Xian JiaoTong University, Xian, China.

[4] Wulun.The theory, Methodology and Application of GIS[M] Beijing: China Science Publishing House,2001 [2] Xiong hanjiang,Gong jianya.The Design of Three Tier Client/Server Web GIS[J].Journal of Wuhan University.

[5] BUEHLER, K. and MCKEE, L., 1998, The OpenGIS Guide, 3rd ed., http://www.opengis.org/ techno/guide/guide/Guide980629.rtf.