Journal of Zhejiang University SCIENCE
ISSN 1009-3095
http://www.zju.edu.cn/jzus
E-mail: jzus@zju.edu.cn

A multi-Agent system architecture for geographic information gathering

GAO Gang-yi (高刚毅)[†1,2], WANG Shen-kang (王申康)[1]

([1]*College of Computer Science, Zhejiang University, Hangzhou 310027, China*)

([2]*Zhejiang University of Finance & Economics, Hangzhou 310012, China*)

[†]E-mail: ggy0222@sina.com

Received July 4, 2003; revision accepted Oct. 1, 2003

Abstract: World Wide Web (WWW) is a vast repository of information, including a great deal of geographic information. But the location and retrieval of geographic information will require a significant amount of time and effort. In addition, different users usually have different views and interests in the same information. To resolve such problems, this paper first proposed a model of geographic information gathering based on multi-Agent (MA) architecture. Then based on this model, we construct a prototype system with GML (Geography Markup Language). This system consists of three tiers–Client, Web Server and Data Resource. Finally, we expatiate on the process of Web Server.

Key words: GIS (Geographic Information System), Multi-Agent, Geographic information, GML
doi:10.1631/jzus.2004.1367 **Document code:** A **CLC number:** TP391

INTRODUCTION

With the rapid increase of geographic information on the Internet, WWW (World Wide Web) contains a great deal of information that can be geo-referenced. The National Academy of Sciences of America estimates that 80 percent of the information on the Web is composed of spatial information that includes coordinate information, such as longitude, latitude and their various kinds of projections, mailing addresses that can be geo-coded, relative distance and direction information. With the wide application of personal portable communication and computing devices, such as cell phones, GPS device and PDA, the construction and applications of geographic reference information (GRI) are undergoing significant changes. It is impossible for a centralized system to collect and manage so much GRI. It is often embedded into various forms of documents, such as HTML pages.

On the other hand, the application of GIS (Geographic Information System) cannot provide users with more and better geographic information services, for example, search, retrieval, on-line analysis, on-line processing, etc. Users do not know where the geographic spatial data is, what it is and how to use it. So how to find the needed data in the enormous data repository is a problem of each user.

To resolve these problems, it is important to retrieve the geographic information. Hence, we assure that the geographic information search engine over the Internet can extend the current text-based search engines, such as Google and Altavista. This paper proposes a geographic information gathering model based on multi-Agent (MA) architecture, which combined GIS and Agent technology. By means of this model, a kind of intelligent geographic information service can be provided comprehensively and efficiently on the Internet.

GEOGRAPHIC INFOEMATION AND AGENT

Geographic information (GI) is an important distributed information resource, including digital,

literal, graphic and image information which directly or indirectly relates to various quantity, quality, distribution features, spatial relation and rule in the geographical field. GI has three basic functions of describing phenomenon itself, recording the spatial location of the phenomenon and reflecting the transformation process of the phenomenon, and is basically the relating of all kinds of information in the real world and forming a synthesis information entity that is continually distributed in time and space.

The concept of Agent originated from Distributed Artificial Intelligent (DAI), and is a basic term of DAI. There is no uniform and specific definition yet for Agent until now. Patie Maes, the director of MIT media lab, deems that Agent is "a computer system, which locates in dynamic and complex environment, can autonomically sense the environment and act accordingly to complete its tasks or goals". Agent is a computing entity with four features of autonomy, reactivity, interaction and initiative. In the research on Agent in China and abroad, different researchers endowed Agent with different construction, context and capability in their own system so that they can conveniently work deep in a specific field.

Based on these features of Agent, it is natural to introduce Agent into the GIS system where it can be applied to such aspects as gathering of geographic information, downloading and transformation of geographic data, cooperation of geographic information service, system integration and individuation style design. Agent offers a new method for computation and problem solution, which has many advantages as follows (Luo and Wang, 2000).

Autonomy and reactivity: Agent can run without the direct intervention of humans or others; it has some kind of control over itself according to the internal state and outside environment. In addition, Agent not only can act by itself, but also can react to the environment. It can get feedback information from the environment, and then redirect its activities.

Interaction: Agents can cooperate with each other. With the development of GIS applications, the functions of GIS become more and more complicated. So many problems cannot be solved by a single GIS system except for cooperation in multi-system. The cooperation of multi-agents can better resolve the cooperation problem of geographic spatial information service functions and GIS application on the Internet and improve the capability and efficiency of GIS service.

Intelligence: Geographic information gathering consists of information searching and information filtrating. We can search and filtrate geographic information by means of Agent, which offers different services to different users, and can remain user's individuation characteristics and provide individuation style according to user's interests.

Distributing: Agent system loosens the restrictions of centralization, close and ordinal control, and offers distribution control, dynamic emergency processing and parallel processing. GIS is just this distributed system.

Software reuse: Agent system can decrease the cost of software and hardware and offer a quicker method of problem solution. Agents can be used repeatedly. Therefore, most codes are shared and the complexity and construction difficulty of GIS are reduced.

MA SYSTEM ARCHITECTURE

MA system architecture

Geographic information gathering system is a multi-Agent (MA) system, which contains a set of software Agents that may run on one computer, and may be distributed on different computers in the network.

To complete the GIS task, Agent needs to communicate and cooperate with other participants (such as users, other Agents or data sources). Within the system, different Agents act as different characters, having different functions and tasks (Tang and Xu, 2001; Eihadi *et al*., 2000).

The overall architecture of the MA system is shown in Fig.1.

Interface Agent

In this model, interface Agent is designed to achieve the interaction between system and users. It realized the generality of user interface. After reque-

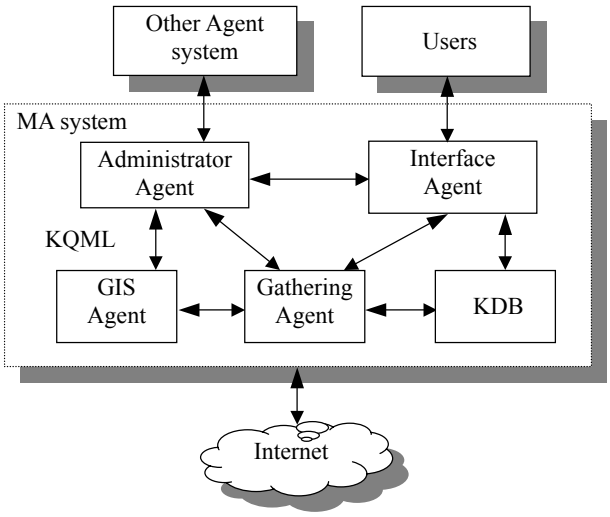

Fig.1 Proposed MA system architecture

sting searching command, what you need to do is just to wait easily but need not know where the data source is or which server to access. Agent will do everything for you. And the system will return the well-processed data to users. So it has some kind of intelligence (Park and Wu, 1999).

One advantage of interface Agent is that it is a friendly interface. You will learn to use it easily in a short time even if you are a rookie in GIS. The main functions of interface Agent include:

(1) Describing user's searching request in some kind of fixed format;

(2) Supporting vague-searching through communication with KDB (Knowledge Based DB);

(3) Communicating with information gathering Agent and GIS Agent, submitting the searching request and returning the resulting data to user;

(4) When user gives an incorrect request description (such as syntax errors), interface Agent will display error information and prompt user to correct it.

User's problem description can be easily described by GeoScript language, which is a kind of interpretative program development language, and is itself a distributed computing platform. It has unexampled advantages in integrating operating systems that are written by different languages and are distributed in different computers. But one of its disadvantages is slowness. In addition, script need to be extended depending on its specific application, for example, GeoScript language in GIS field.

Administrator Agent

Administrator Agent manages the whole system, processing the complex communication between the inside and outside of the system, coordinating all the Agents in the system, and distributes controls in the system. It is unique in the system. The main functions of administrator Agent include:

(1) The interface of the system with other systems or Agent systems;

(2) Administrating Agent registers the table of the system;

(3) Coordinating the interaction of Agents in the same system;

(4) Creating and administrating all the active Agent instance, including the status and life cycles of active Agents;

(5) Processing the communication of Agents.

The Agent register table includes all information that identifies the Agent, besides its ID, address, name, etc. It is an important part that records the specific method and the service function of the cooperation of Agents. According to this register table, administrator Agent creates Agent instances and uses them.

To create an instance, it is necessary to analyze the searching request, then try to find out what type of Agent need to be created and how to create it by accessing the register information of the administrator Agent. Finally, create it.

All the Agents in the system use a common communication language–KQML, which can be used not only as a message format, but also a message processing protocol of knowledge sharing system. KQML primitive is described by common LISP grammar.

In distributed geographic information system (DGIS), the balance of the network burden is an important problem, because:

(1) The data transmission of geographic spatial information is huge;

(2) The cost of the communication between Agents is too high.

Hence, to reduce the network burden, we must:

(1) Try to reduce the inter-network communication, especially the communication with large amount of data. A solution is to put the return data in the local database, if there is a similar request later,

just returns the data to users by accessing the local database;

(2) Try to reduce the amount of communication data among agents (this should be taken into account when designing the system);

(3) Using the mobile feature of Agent, we may move some work to idle network nodes to balance the network burden.

Information gathering Agent

Information gathering Agent is the Agent that searches and filters the geographic information of DGIS, acting as an important part of the system.

Information gathering Agent receives the client request from the interface Agent, and then searches on the web. After getting the searching result, information gathering Agent will filtrate the information according to the user's interests and strategies stored in KDB. In addition, it will modify the personal interests of users and individualize them for later searching to be referenced. Other work, such as the downloading of geographic information, is in the charge of GIS Agent.

GIS Agent

GIS Agent is the Agent with some specific function in the system, such as displaying Agent, analyzing Agent and downloading Agent.

The creating process of GIS Agent is:

(1) Interface Agent transfers the user's request to administrator Agent;

(2) Administrator Agent searches the Agent that needs to be created according to the description of the request;

(3) Administrator Agent creates the instance of GIS Agent.

According to different system demand, we can flexibly design different kinds of GIS Agent to complete specific missions. System can be extended flexibly based on the specific requirement.

KDB

KDB stores the knowledge embedded in the system and the rules defined by users. According to users' specific demand, they can create KDB to implement intelligent information service.

The main functions of KDB include:

(1) Communicating with interface Agent to support specific searching strategy;

(2) Storing the searching rules frequently used (such as AND, OR, NOT);

(3) Storing the information about the searching of spatial metadata DB (SMDB). This part can also be stored in its separate DB. Information gathering Agent accessing the corresponding spatial metadata DB is also through KDB;

(4) Storing user's personal information, such as user's name, key word of searching.

The traditional DBMS requires users to understand the mode of the database, and can only process definite searching request. The design of KDB has changed all of these. If the user's request is not very clear, the system can also process it with vague-searching through the strategy provided by KDB. Hence, the design of searching strategy is the key problem.

KEY TECHNIQUES FOR IMPLEMENTATION

Based on the model of MA system architecture above, the author proposed a construction method of the model system, and implemented a simple prototype system. Its architecture is shown in Fig.2.

The architecture system consists of three tiers–Client, Web Server and Data Resource. GML (Geography Markup Language) standard is adopted in data transmission between tiers. GML, which is an extension of XML, is proposed by OGC (OpenGIS Consortium) to solve the interoperability problem.

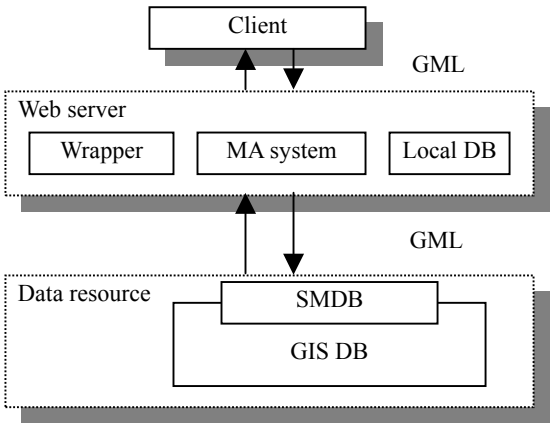

Fig.2 The architecture of application system

In May 2000, OGC proposed GML1.0 in its interoperability project, and then research was going on. Feb 2001, GML2.0 was born. GML 3.0 has been presented to the public. It is supported by many corporations such as Oracle, ESRI, Galdos and MapInfo. ESRI integrated XML in ArcIMS3.0z. ArcXML is a product on XML and its content is a super set of XML (OpenGIS Consortium, 2001).

Coding in GML has many advantages:

1. General component can be used repeatedly;
2. The interoperability of GIS can be done;
3. Consistency of web service;
4. Separating content from display, GML coding is easy to be transformed to SVG by XSLT;
5. GML encapsulates the Geographic information, attribute information, spatial geographic reference system and the main projection relations so that the extensibility and flexibility of distributed processing can be guaranteed;
6. GML is a convenient method for the distribution of geographic data, and is also an important method for distributed storing of geographic data.

GML generally use a data description language rule of Schema, similar to Document Type Definition (DTD). Citing a city schema, we describe the presentation of GML, in which we define our own type–RoadType, RiverType, MountainType. Limited by pages, here is only a part of all (Fig.3).

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- File: city.xsd  -->
<schema
targetNamespace="http://www.opengis.net/examples"
xmlns:ex="http://www.opengis.net/examples"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:gml="http://www.opengis.net/gml"
xmlns="http://www.w3.org/2000/10/XMLSchema"
      elementFormDefault="qualified"
version="1.0">
     <annotation>
        <appinfo>city.xsd v1.0 2002-04</appinfo>
        <documentation xml:lang="en">
         GML schema for the Cambridge example
        </documentation>
       </annotation>
        <!-- import constructs from the GML Feature and
Geometry schemas -->
       <import      namespace="http://www.opengis.net/
gml" schemaLocation="feature.xsd"/>
<!-- ============================
          global element declarations
     ============================== -->
<element name="CityModel" type="ex:CityModelType"
           substitutionGroup="gml:_FeatureCollection"
           />
<element  name="cityMember"  type="ex:CityMember-
Type"
         substitutionGroup="gml:featureMember"/>
<element name="Road" type="ex:RoadType"
         substitutionGroup="ex:_CityFeature"/>
<element name="River" type="ex:RiverType"
         substitutionGroup="ex:_CityFeature"/>
<element name="Mountain" type="ex:MountainType"
         substitutionGroup="gml:_Feature"/>
```

Fig.3 Part of GML schema

Web Server

Wrapper is an important part of server. It is in charge of the format transformation between geographic spatial data and GML so that is can make full use of the advantages of GML. Developed by Java Server, Wrapper accesses database (such as SQL Server and Oracle) through JDBC-ODBC Bridge or Oracle JDBC.

LDB (Local Database) temporarily stores the geographic information data that has just been found. Therefore, the amount of transformation data and the possibility of network jam are reduced. And if users send a similar request, data can be returned in time by accessing the LDB instead of searching in Internet.

Data Resource

The searching for data resource is mainly through SMDB (Spatial Meta DataBase).

SMDB stores spatial metadata. Spatial metadata is the description information about geographic spatial data and corresponding information source. It helps users locate, evaluate, compare, retrieve and use geographic information data efficiently through describing and illuminating the content, quality, condition, position and other features of geographic spatial data. Spatial metadata administrates data with multi-index so that users can locate the data quickly. With the development of Internet, metadata has been not only a method of data description and index, but

also a powerful and indispensable tool of data finding, data transformation, data administration and data use in the whole network information process.

Client

In client, the searching result is displayed in the form of web pages. Sharing of the geographic information data on the Internet involves many problems. So it is little possible to use these data free, while the searching result can be used as useful reference.

Now most sharing forms of geographic information are the map and the display of geographic information data that mostly adopts vector-graphics. Since GML separates the content of geographic data from the display of graphics, there are many tools to interpret its content, such as SVG (Scalable Vector Graphics), Microsoft VML (Vector Markup Language) and X3D. Office supports VML in output graphics format and Internet Explorer 5.0 supports its display. AutoDesk also supports VML output in AutoCAD Map. As W3C standard, SVG is widely applied, and is supported by many products. The GML data display of OGC actually adopts SVG to display the map information. According to the Map Style Sheet of GML data display processing, SVG, using XSLT (XMSL Transformations) criterion, transformed GML data into SVG format and displayed it. Nowadays most of the browsers such as Internet Explorer (IE) support SVG (World Wide Web Consortium, 2001).

PROCESS OF WEB SERVER

In the system construction, design of Web Server is the key to the system. The processing of web server is shown in Fig.4.

Web Server receives user's request, searches it, transforms it into GML format and returns the result to the user. If the result is already in LDB, Server accesses LDB directly and transforms the data into GML document then returns. In addition, LDB need to be updated after searching.

When submitting the user's searching request to database, we face two different kinds of database. One is the database to support spatial data searching; the other is the database not to support spatial data searching. SQL Server does not support spatial data type, nor support spatial data searching. So the geographic data result cannot be used directly. After the data is obtained, you need to add the records to GML documents. Oracle is a widely used and object-relation database. It allows users to store and retrieve geographic objects such as point, line and

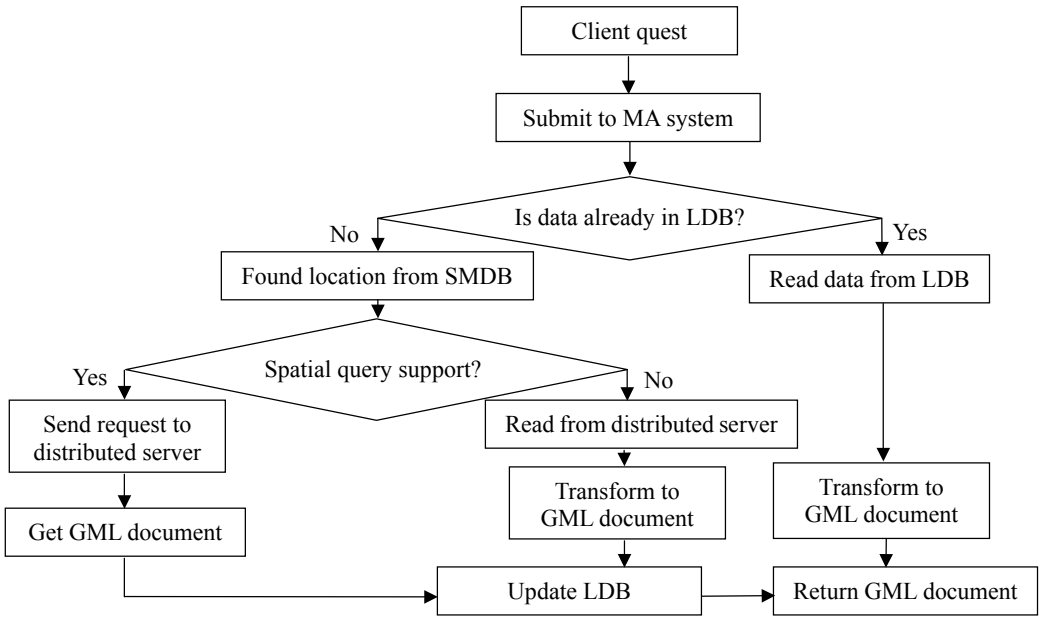

Fig.4 The process of Web Server

area. At the same time, Oracle provides spatial data access method (R-tree). Searching and analysis of spatial data are supported, and the result of user's request is returned in the form of GML document.

CONCLUSION AND FUTURE WORK

Internet is an important source for retrieving geographic information, and it is impossible for one integrated system to collect and administrate so large amount of GRI. However, Agent technology gives us a brand new method to solve the problem. After analyzing geographic information and Agent, this paper proposed a model of geographic information gathering based on multi-Agent (MA) architecture and discussed the system construction by means of GML. The prototype system with this architecture has the following features and advantages:

1. System construction is flexible and extensible. Agent with different functions can be customized.

2. Users can retrieve geographic information from Internet.

3. Agent provides different searching strategies according to user's interests, and keeps the personal characteristics of the user.

4. Improving the capability and efficiency of DGIS service.

Because of the limitation of system design and implementing method, the prototype system still needs to be improved. What is more, it needs further research for MA (Multi-Agent) system to be applied in practice. In addition, MA system lack of a standard of system architecture and organization, and efficient methods to construct and evaluate MA system.

References

Prak, S.Y., Wu, C.S., 1999. Intelligent Search Agent for Software Components. IEEE Computer, p.154-161.

OpenGIS Consortium, 2001. Geography Markup Language (GML) version 2.0. http://www.opengis.net/gml/01-029/GML2.html.

Tang, C., Xu, L.D., 2001. An agent-based geographical information system. *Knowledge-Based System*, **14**: 233-242.

Luo, Y.W., Wang, X.L., 2000. Research on the Geo-Agents. *Journal of Computer Research & Development*, **32**(12):1504-1511.

Eihadi, S., Hamada, G., Mohamed, K., 2000. A Multi-agent Architecture for Information Gathering. IEEE, p.732-736.

World Wide Web Consortium, 2001. Scalable Vector Graphics (SVG) version 1.0. http://www.w3.org/TR/SVG.