RESEARCH ON DISTRIBUTED GIS BASED ON MOBILE AGENT

Min CHEN, Jihong GUAN, Xiaobin HUANG
Information School, Wuhan Technical University of Surveying and Mapping,
430079, Wuhan, P.R.China
chenmin@dus.wtusm.edu.cn

KEY WORDS: Data dissemination, Dynamic processes, GIS, On-line, Mobile agent.

ABSTRACT

With the emergence of more sophisticated distributed computing environment and wide spread acceptance and available of Internet and World Wide Web (WWW) technologies, a new generation of Geographic Information System (GIS) emphasizing on open architecture, distributed computing capability, mobility, interoperability, extensibility and intelligence has been receiving more attention over the past decade. Distributed computing mode based on mobile agent is a revolutionary mode that can overcome the innate limitations of the C/S or B/S mode. This paper presents an architecture of distributed GIS based on Mobile Agent to meet the new requirements of GIS applications under Internet environment with limited communication bandwidth and unstable connectivity. It analyzes the related technologies of distributed computing based on Mobile Agent: its architecture, accomplishment of agents' move, and technological support for collaboration. And last it discusses GIS applications based on mobile agent on Internet. Our research on distributed GIS integrates several existing technologies, most of which are Java-based, and we also take Concordia as the platform to build our distributed GIS based on mobile agent. We have accomplished mobile agents' move and collaboration based on Concordia. And the combination of Message Queue System, Proxy and Persistence Manager makes Concordia systems secure and reliable.

1 INTRODUCTION

With the rapid development of network technology and popularity of Internet and World Wide Web (WWW) in the past decade, GIS (Geographic Information System) has been seeing a switch from the original isolated and centralized information management model to an open and distributed architecture, so that the wealth of information stored in and managed by the variety of GISs locating over all the world can be publicly accessed through the Internet in almost every corner of the earth, which makes distributed GIS the fundamental platform of the ongoing digital earth project.

However, Internet has its innate open, distributed, heterogeneous and dynamic properties and its multifarious and disorderly resources, and the traditional approaches for information access in distributed GIS co-locate the data and the computation needed to process it by bringing the data to the computation, which unavoidably will increase the network burden considering that in a GIS there is large amount of data and the spatial data typically has an extremely complex and variable structure. Furthermore, the dominating Client/Server (C/S) architecture of current GISs makes their performance be prone to replying on wide communication bandwidth and stable network connectivity. Obviously, the problem lies in that the currently prevalent distributed computing paradigm and GIS architecture could neither deal with the Internet environment well nor meet the requirements of the new emerging application fields, such as active services,

intelligent search, mobile computing, etc. Such a situation calls for both new architecture and distributed computing technology for distributed GIS under Internet environment.

Mobile Agent is a recently developed computing paradigm, which combines agent technology and distributed computing technology and is well suited for open and dynamically changing environment. It may be an effective way to introduce mobile agent computing paradigm to distributed GIS in Internet environment with limited communication bandwidth and unstable connectivity. So in this paper we present a new architecture of distributed GIS based on Mobile Agent to meet the requirements of GIS applications under Internet environment. And it analyzes the related technologies of distributed computing based on Mobile Agent: its architecture, accomplishment of agents' move, and technological support for collaboration.

2 MOBILE AGENT TECHNOLOGY

2.1 Mobile Agent Concept and related programming language

Mobile Agent can be defined as an independent computer program that can continually execute in cross-platforms, has the abilities of self-control moving, imitating human's behavior and relationships, and providing certain Artificial Intelligent services. It can autonomously move in heterogeneous network according to some certain rules, searching for suitable computing resources, information resources or software resources. It applies the advantage of being at the same host or network with these resources to accomplish particular tasks on behalf of users, processing or applying these resources (Funfroken, 1997).

Introducing the thinking of distributed mobile computing into Agent technology makes Agent technology have many new characteristics: dynamic execute mode, asynchronous and autonomous computing mode, parallel finding the solution, intelligent route, strong fault tolerance.

At present time, there are many tools suitable for developing Mobile Agent Systems: C++, Script language (i.e. Tcl), Java, and Agent-oriented Programming Language, etc., and Java is especially the right language for Mobile Agent Systems. Our research on distributed GIS integrates several existing technologies, most of which are Java-based. We choose Java because it is simple, object-oriented, distributed, interpreted, robust, secure, architecture-neutral, portable, high-performance, multithreaded, and dynamic. Such features make it ideal for the Internet's distributed heterogeneous environment, architecture-neutral and portable applications. And we also choose Concordia as development platform for it is Java-based and is a complete framework for developing and managing effectively various Mobile Agent applications (ITA, 1998).

2.2 Development of network GIS

GIS software has gradually developed form single-computer-based or Mainframe/Terminal-based to local network-based since 1980s. After that more and more local networks connect to Internet, then Internet GIS emerges. Most distributed GIS applications are based on C/S mode. However the mode's innate limitations make it more and more incompetent to meet the various requests of complicated, dynamic distributed computing when Internet is growing so rapid and the distributed computing environment becomes much more distribute and dynamic.

First, to adopt C/S mode requires high quality and stable performance of the network connection. Only stable connection established and maintained can client's identity be identified, and the security of interactivity is ensured. The request for a server is usually an interactive process, so the client should be able to confirm some predictable response. What's more, in order to replicate and transfer data on the network smoothly, a proper bandwidth is required.

The second problem is that each client has to manage and maintain multi-connection to multiple servers, while server's control becomes more complicated and the performance are lowered with the growing of scale.

The third problem is that C/S mode cannot well support the direct group awareness among the clients' applications, so it is difficult for these applications to get full interactivity and to achieve high level cooperative. It results in incompetent of C/S mode in accomplishing complicated distributed applications, for interactivity and collaboration are critical to distributed applications.

3 DISTRIBUTED GIS BASED ON MOBILE AGENT

3.1 Architecture of Distributed GIS based on Mobile Agent

We choose Concordia as the platform to establish our distributed GIS based on mobile agent. Figure 1 shows the environment and architecture of distributed GIS based on mobile agent.

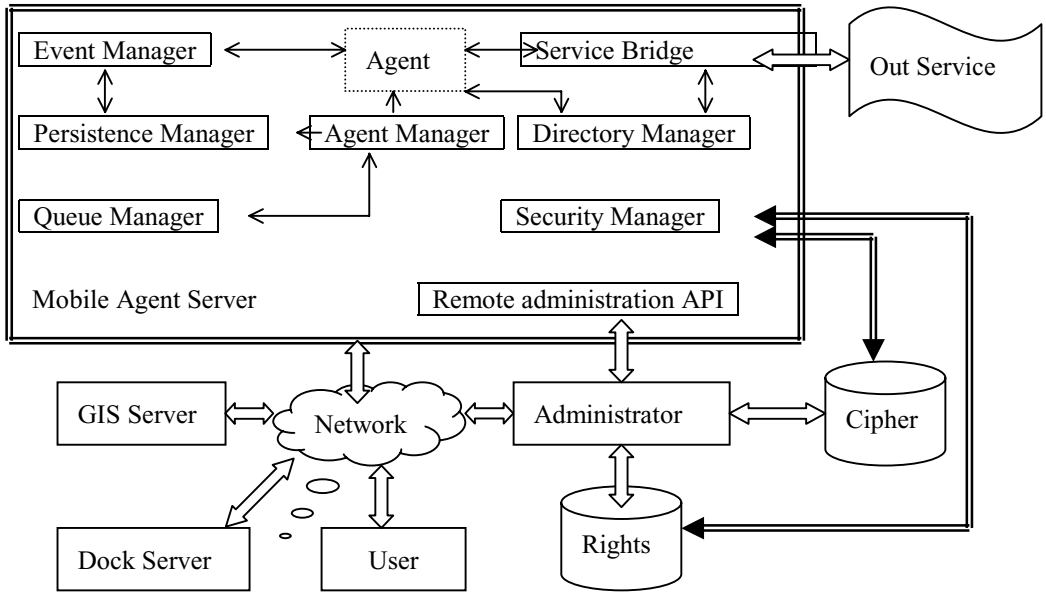

Figure 1. The environment and architecture of Distributed GIS based on Mobile Agent

In Figure 1, the users access the servers through network and are served by mobile agents stored at the corresponding servers. Mobile Agent Servers collect and deliver information by starting up its own mobile agents. Next section gives the detailed explanation of Mobile Agent Servers. The agent Dock Servers are machines that are permanently connected to the network and used as dock for those agents which are originated from the host that are disconnected at the time when they migrate back to it. Docking system is used to solve the congestion of network and make applications suitable for low reliable network. A GIS Server is composed of Graphical User Interface (GUI), Query Manager (QM), Object Manager (OM) and Service Manager (SM)(Choy, 1993). Figure 2 shows the structure of GIS server.

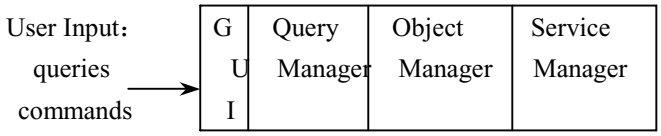

Figure 2. GIS Server of Distributed GIS based on Mobile Agent

3.2 Mobile Agent System and its server

A mobile agent system includes several parts and the Mobile Agent Server is the core part. Administrator manages services provided by Mobile Agent Server and can cooperatively work with the services. The operations on it are accomplished through user's interface.

In general, a Mobile Agent System has several Mobile Agent Servers simultaneously running on different nodes in the network. A Mobile Agent Server includes the following main components (Huang, 1999): (1) Agent Manager: provides the infrastructure for agents' transferring and receiving in the network, and manages agents' life cycle. (2) Event Manager: manages registration and notice events sent to or from agents and transfer the events to agents in other nodes and issue events in accordance with necessity along with the Mobile Agent Server. (3) Persistence Manager: keeps the status of agents moving in the network. (4) Queue Manager: gives the mechanism of queue and management for the execution of agents, maintains agents when they are waiting for their turns to accomplish tasks and keeps their persistent status when agents come into or go out of the system and makes necessary retry when the mobile agent system disconnected to the network. (5) Directory Manager: provides uniform name service, so agents can find the service needed in the network. (6) Security Manager: is responsible for distinguishing users and authenticating their agents, and protects servers' resources and ensures the security and integrity of the agents and their data objects when moving in the network. It also controls dynamic load needed of Java class library to agents. (7) Service Bridge: provides a mechanism for developers, which makes it possible to add some related services, and to access local services on the destination server. Remote management is executed through Remote Administration API.

3.3 Mobility of distributed GIS based on mobile agent

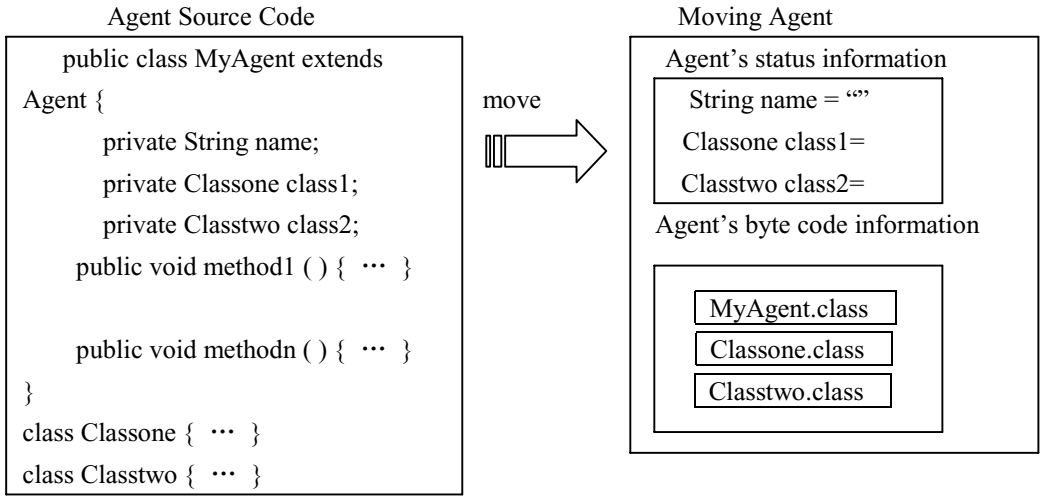

Figure 3. Mobile Agent's Move

Mobile Agents' move is quite different from distributed objects' interaction or Java Applet's download. In distributed object systems based on CORBA or DCOM, an object can call other remote object's methods through the network, but during the interactive process of two objects neither object can move. Applet's download in World Wide Web only provides the mechanism of code download from Web Server to Web Browser, not provides any mechanism for move of related status information, but only download code, for the code downloaded will create an object and once the Applet is created it can only stay at the Browser, but not move. While Mobile Agents' move is quite different, it can not only bring the corresponding code but also bring the related status information with it during the process of move (Huang, 1999). We can see this from Figure 3.

Moreover, Mobile Agent can move several times in the light of the user's request. In the process of Mobile Agent's move, the move plan is decided by itinerary. It is an independent data structure of Mobile Agent in Concordia. The Table 1 is an itinerary of a mobile agent:

Server Name	Method
Server1	method1
Server2	method2
Server3	method3

Table 1. An itinerary of a mobile agent

In accordance with this itinerary, the mobile agent will first arrive at Sever1 and execute method1 in this server. Then it will move to Server2 and execute method2, and at last it will move to Server3 and execute method3. We see that this mode provides a simple mechanism for the definition and trace of Mobile Agents' move. What's more, the system enables the mobile agent to change its itinerary during the move. This gives great flexibility for Mobile Agents' mobility.

4 KEY TECHNOLOGIES AND APPLICATIONS

4.1 Class definition and agent's migration

Agent is the base class, in which many essential services needed in the process of agent's move and execution i.e. itinerary plan, agents' transmission and events processing, etc., are encapsulated. In most cases, the mobile agents designed by developers receive the corresponding services by inheritance form the base class.

For example, if we want to know the prices of some pieces of land available in one city, we can design a mobile agent inherited from the base class Agent as RemoteQueryAgent. And let it move to the host of the city's Land Bureau to query the prices, then report the corresponding information to us. To accomplish the task, we need to at least design one member variable, two methods and one itinerary object:

• The member variable LandPrice: It stores the prices of certain pieces of land queried by the mobile agent.

• Method QueryPrice(): The mobile agent executes the method when it arrives at its destination. The method does queries to certain database through JDBC-ODBC Bridge, and puts the result in member variable LandPrice.

• Method PriceReport(): This method will be executed when the mobile agent returns to the user, and its function is to report the prices in accordance with the variable LandPrice.

• Itinerary: this object gives the move route of the mobile agent. If we use "Remote" to represent the host of the city's Land Bureau, "Local" to represent user's local host, then we can have the itinerary as Table 2.

Host Name	Methods Executed
Remote	QueryPrice()
Local	PriceReport()

Table 2. Itinerary of RemoteQueryAgent

The execution of RemoteQueryAgent is shown as Figure 4.

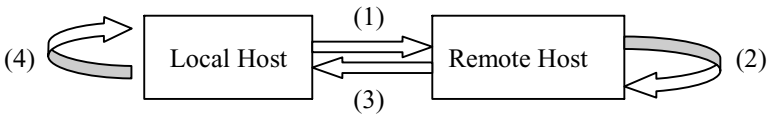

Figure 4. The execution of RemoteQueryAgent

(1) RemoteQueryAgent moves to destination Remote Host through the network; (2) RemoteQueryAgent executes QueryPrice() in Remote Host and puts the results in variable LandPrice; (3) RemoteQueryAgent returns to user's Local Host through the network; (4) RemoteQueryAgent executes method PriceReport() in Local Host and reports related result to the user in light of variable LandPrice.

4.2 Mobile Agents' Collaboration

The collaborative frame in distributed GIS based on mobile agent allows mutual coordination of multiple agents to solve complicate problems. Agents in an application can form several Agent Groups (Wang, 1997). The corresponding base classes for cooperative agents and Agent Group are CollaboratorAgent and AgentGroup. AgentGroup has a simple interface for cooperative CollaboratorAgents, while these CollaboratorAgents will apply Remote Method Invocation (RMI) of Java and use a remote invocation of an AgentGroup to access to corresponding AgentGroups and accomplish collaboration. AgentGroup collaboration is achieved through a software method "analyzeResults" and a collaborative point.

To each agent who arrives before the collaborative point from which collaboration starts, it will be sent in block status after its execution results sent to AgentGroup. When all agents in the group have arrived, AgentGroup will call the corresponding method "analyzeResults" to analyze and process the result set of the agents in the group. Every application should give its own analyzeResults. Each agent in the group will be awaked and have its analysis results, and the agents can adjust their behavior in accordance with the results.

To solve the complicated decision problem of site selection, we can use the collaborative mechanism:
(1) Create an Agent Group: SiteSelAgentGroup; (2) create several remote query agents: CollQueryAgent instances, and let the instances created belong to the group SiteSelAgentGroup; (3) set itineraries for each instance; send out CollQueryAgent instances; (4) and use SiteSelAgentGroup to synthetic process the results sent back from each agent in the group and find out the best solution.

SiteSelAgentGroup inherits the group base class AentGroup provided by Concordia, and includes one method analyzeResults( ). This class provides corresponding environment for agents' collaboration in the group, and through reloading the base class AgentGroup's method analyzeRestults( ) to do synthetic analysis and have the last results.
CollQueryAgent is similar to RemoteQueryAgent in section 3.2.2. It also includes member variable LandPrice, executive method QueryPrice( ) and PriceReport( ) and the Itinerary, but it inherits from base class CollaboratorAgent while not from Agent. Its construction function should have one parameter to point out which group it belongs. In the execution of method QueryPrice( ), agents not directly put the prices queried to member variable LandPrice, while achieve the final results through the collaboration of CollQueryAgent and its owner group SiteSelAgentGroup.

5 CONCLUSIONS

Internet is a global network of computers connected through communication devices to one another. It has been changing the ways of data access, sharing and dissemination; it is further changing the means of analysis and visualization in GIS. Internet-based distributed GIS are rapidly evolving as Internet and distributed computing technologies change. Distributed computing based on Mobile Agent is a new revolutionary computing-mode. Mobile Agent's properties of mobility, collaboration, high security and reliability bring new opportunities and new challenge to Internet-based applications, especially distributed applications as Internet-based distributed GIS. Mobile Agent has wide prospect in many fields, i.e., parallel distributed information retrieval and process, personal assistant and security intermediate, and active service, electronic commerce, knowledge mining, and intelligent communication etc. It is going forward to be standard, intelligent and practical.

REFERENCES

Choy M., Kwan M.P., Leong H.V.,1993. On Real-time Distributed Geographical Database Systems. http://www.cs.ucsb.edu/TRS/techreports/TRCS93-21.ps

Funfrocken S., 1997. Mobile agents as an architectural concept for internet-based distributed applications. http://www.informatik.th-darmstadt.de/vs/publikationen/fuenfrochen/papers/tr-vs-97-01.ps.

Huang X.B., 1999. Research of Distributed Applications Based on Mobile Agent, [Thesis] Wuhan Technical University of Surveying and Mapping.

ITA, 1998. Mobile Agent Computing—A White Paper, Concordia User Documentation, Mitsubishi Electric Information Technology Center America (ITA). http://www.meitca.com/HSL/Projects/Concordia/.

Wong D., Paciorek N., Walsh T., et al, 1997. Concordia: An Infrastructure for Collaborating Mobile Agents, In Mobile Agents: First International Workshop, Lecture Notes in Computer Science, Vol.1219, Springer-Verlag, Berlin, Germany.