A NEW APPROACH FOR DISTRIBUTED GIS

Yuxiang WANG, Chongjun YANG, Donglin LIU
China Beijing 100101
Institute of Remote sense Application, Chinese Academy Sciences

KEY WORDS: Distributed GIS, CORBA, ORB, Geo-Spatial Data

ABSTRACT

The research described in this paper explores a new approach towards distributing GIS tools across multiple CORBA ORBs. The main structure of a distributed GIS is introduced. How CORBA and GML are integrated is explained. Object migration is implemented among servers for load balance and security is considered. Then the main procedure of designing a CORBA distributed GIS is introduced.

1. INTRODUCTION

GIS has become an indispensable spatial data processing tool for government and research. With strengths in reliability, processing efficiency, resource sharing, parallel processing and system scalability. A distributed GIS is a collection of GIS sites. Each site is an autonomous GIS that contains both geographic and spatial data. It is a group of physically distributed GIS sites. They are logically united as one site. For user, geographic service can be accessed transparently.

- Compared with a centralized-server GIS, the major advantages of a distributed GIS are as follows:
  - Evenly load balance can be achieved by strong multiple GIS sites. When a site works wrong, other sites will finish the process left.
  - High speed can be achieved by distributing the whole job to multiple sites for parallel processing.
  - Data and processing interoperability are enhanced and system development cost is reduced.
  - In a well-planned distributed GIS, system is easily scalable.

A distributed GIS has advantages in data sharing, reliability and system growth. In recent years, distributed GISs have attracted increasing interest. For example, the Distributed Geographic Information System Project in Australia and in Norway, and the geodata modeling technique for distributed GISs at Berkeley. The National Center of Geographic Information Analysis(NCIGA) includes distributed GIS in its research plan. The University Consortium for Geographic Information Science included distributed computing and interoperability in its research priorities.

2. CORBA BRIEF

The Common Object Request Broker Architecture (CORBA) is made up of Object Request Brokers (ORBs) that can communicate with each other via the General Inter-ORB Protocol (GIOP), the Internet Inter-ORB (IOP) Protocol [2]. CORBA facilitates a middle-tier, object-to-object infrastructure that allows one to encapsulate data from multiple sources. The language and compiler used to create the server objects, as well as the location of distributed CORBA objects and the operating system they execute on, are totally transparent to clients. CORBA therefore provides an ideal mechanism for creating 3-tier (or n-tier) distributed applications that go beyond providing simple interoperability [2].

CORBA provides one with ability to perform dynamic discovery of objects and services, as CORBA objects are self-describing and introspective. CORBA's dynamic facilities, including Dynamic Invocation Interface (DII), and the Interface Repository allow the creation of extremely flexible systems that allow run-time discovery and late-binding [2]. This is especially useful in the Web environment where a user is able to discover new services and then make use of them transparently.

3. DISTRIBUTED GIS MODEL WITH CORBA

Distributed GIS is based on a distributed geospatial database. Clients can access GIS server remotely and concurrently. The whole structure of our distributed GIS is showed (Fig 1). Because our distributed web GIS servers include many functions, we divide our GIS functions into different ORBs. They are projection Transform ORB, Data Format Transform ORB, Spatial Database Storage ORB, Spatial Index ORB and Application ORBS (Fig 2). All the CORBA objects are communicated by IOP protocol. They are located and managed by Naming Service and Interface Repository. The naming service locates data objects by name. Each data object, for example, a digital map or a remote sensor image, has a descriptive, recognizable name used for query. The naming service maps the name into the address and the reference of the object.

![Distributed GIS Architecture](image)

Fig 1: Distributed GIS Architecture

GML Repository

<table>
<thead>
<tr>
<th>Distributed GIS</th>
</tr>
</thead>
<tbody>
<tr>
<td>CORBA ORB Group</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ORB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ms Access</td>
</tr>
<tr>
<td>SQL Server</td>
</tr>
<tr>
<td>ORACLE</td>
</tr>
</tbody>
</table>

CORBA Service

<table>
<thead>
<tr>
<th>Client</th>
</tr>
</thead>
<tbody>
<tr>
<td>IOP</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Client</th>
</tr>
</thead>
<tbody>
<tr>
<td>IOP</td>
</tr>
</tbody>
</table>

3.2 GIS Orbits
Fig 2  Distributed GIS CORBA ORB Server Group Figure

3.1 Projection transform ORB
According to the concept of spatial reference system of Geodata Model, attributes are tied with features by coordinate. The spatial reference system is function which associates locations in space to geometries of coordinate tuple in mathematical space, usually a real valued coordinate vector space.

The Conversion ORB's Interface is described in the file: ProjectedTransform.idl interface Projection
{
    attribute string ProjectionName;
    attribute long ProjectionNumber;
    attribute double originateX;
    attribute double originateY;
    void LLtoProjection(in double latitude, in double longitude, out double outX, out double outY);
    void ProjectiontoLL (out double latitude, out double longitude, in double x, in double y);
}

Fig 3  Projection Transform Model

Projection transform ORB supplies operation for coordinate transformation, and allows points to be transformed between any ellipsoid coordinate system and projection coordinate systems such as Gauss and UTM. It provides for a "domain of validity" for each coordinate system transformation, as most transformations are valid only over a portion of the Earth. Exceptions are raised when a transformation is invoked outside its domain of validity.

3.2 Format Transform ORB
Geographic data has been collected in digital form for more than 30 years. The overall rate of collection increases rapidly with advances in technologies such as high resolution satellite-borne imaging systems and global positioning systems. Geodata formats tend to be complex, more complex than other kinds of digital data formats, because of the range of information they must be able to represent. Usually, the complexity that begins with the underlying digital format imposed by a particular software application or acquisition method is incremented by the complexity of higher level descriptions, conventions, and rules imposed by the individuals, organizations, and disciplines using the software. Because spatial data exist in a wide range of incompatible and often vendor-proprietary forms, and geographic information systems (GIS) usually exist in organizations as isolated collections of data, software, and user expertise. So I developed the format transform ORB including interface for exchanging diverse geodata then applications can access remote spatial databases and spatial processing resources in real-time.

3.3 Spatial Database Storage ORB and Query ORB
Geospatial data are structured based on the Open Geodata Model of the OGIS4. Spatial entities are represented as features. Features are the primary units for access, management, manipulation and interchange. A feature is specified by geometric and attribute properties. Based on the Open Geodata Model, three basic spatial entity types are used to specify zero, one, two and three dimensional features. But spatial data in files are difficult for query. For quick query they are indexed in spatial databases. Our spatial data structure is based on spatial occupancy. R-tree is designed to organize a collection of arbitrary spatial objects. Each node in the tree corresponds to the smallest d-dimensional rectangle enclosing its son nodes. We developed the main spatial query methods including buffer, overlay and so on. They are listed in Table 1.

Table 1  Spatial query method for distributed GIS
<table>
<thead>
<tr>
<th>Function</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overlay(R1, R2)</td>
<td>Overlay two maps: R1, R2</td>
</tr>
<tr>
<td>Buffering(R1)</td>
<td>Buffer of line, polygon and point</td>
</tr>
<tr>
<td>Distance(A, A')</td>
<td>Distance of point A and point A'</td>
</tr>
<tr>
<td>Within(A, A')</td>
<td>If line, point or polygon A, within A', then return TRUE.</td>
</tr>
<tr>
<td>Intersect(A, A')</td>
<td>If line A and Line A', intersect, then return TRUE.</td>
</tr>
<tr>
<td>Contain(A, A')</td>
<td>If polygon A contain polygon line or point A', return TRUE; else FALSE.</td>
</tr>
<tr>
<td>Adjacent(A, A')</td>
<td>If polygon A and Polygon A', adjacent, then TRUE; else FALSE.</td>
</tr>
</tbody>
</table>

Because query in a distributed GIS involves a large number of spatial operations, it is more complicated than query in a normal distributed relational database. When query for remote data, data distribution and transformation costs must be considered.

3.4 Application ORBS
Application ORBS is the kernel of distributed GIS. Main geoprocessing is finished by application ORBS. They can be divided as two groups. One is the general application group...
including geospatial data visualization and map annotation and symbolization. The other is related with research domain. Models are changed with domains such as environment administration and land use and so on.

In our system, all sites have identical structures in the sense that each has a query server, and an ORB with several services objects that provide common services including naming services, persistence services, query services and so on. Since users may request data stored at remote sites, the query are distributed to multiple sites to execute. The query service is also responsible for initiating execution of the query. The ORB provides mechanisms for inter- and intra-site communications between objects. The naming service is used to locate named objects. The persistence service objects provide an object-oriented interface for the data persistently stored in the local service.

3.5 Integration of GML (Geographic Makeup Language) and CORBA
The emerging set of XML Standards is gaining widespread industry support in content management, delivery, and presentation at the web-based front end of today's Enterprise Systems. As spatial data will be encoded in an GML form and sent across the internet, a need for native processing of GML data under CORBA is required. Integrating CORBA and GML efficiently processed by CORBA systems providing data access and query services to the web; GML data to be sent back to web systems; CORBA Objects be accessed through XML based Messaging protocols. The Integration of CORBA and GML is a small piece of the overall solution to the provision of Enterprise Level software systems for today's emerging web-based GIS and traditional GIS migrating services to the internet.

![Diagram of GML and CORBA integration](image)

The diagram (Fig 3) illustrates the GML processing system implemented using a Java ORB client embedded in the Web Browser, with a Java ORB server processing the received document, storing relevant details in the order database, and generating an GML encoded receipt which it returns to the ORB client. Both the sent and received documents are serialized into text before they are sent or received, and require parsing into a manipulative form in order to be processed in GIS client.

4. LOAD BALANCE AND SECURITY
Object migration is the process of terminating an object implementation on one host, and then starting it on another host.

Object migration can be used to provide load balancing by moving objects from overloaded hosts to hosts that have more resources or processing power (there is no load balancing between servers registered with different ORB servers). Object migration can also be used to keep objects available when a host have to be shutdown for hardware or software maintenance. The migration of objects that do not maintain state is transparent to the client program. If a client is connected to an object implementation that has migrated, the ORB service will detect the loss of the connection and transparently reconnect the client to the new object on the new host.

When a distributed GIS based on CORBA is deployed over the Internet or Intranet, many security restrictions can apply to the system, including the following:
- Java sandbox security prevents unsigned Java applets from communicating with servers other than the ones running on the host machine from which the applets were downloaded.
- Server-side firewalls can prevent the client from accessing certain hosts.
- Client-side firewalls can prevent incoming connections or prohibit protocols other than HTTP.

When certain restrictions prevent the clients from connecting directly to the server, the client can choose to connect to the gateway if the server object reference has the necessary information. The clients can send messages to gateway; The gateway will forward the messages to the server.

When certain restrictions prevent the server from connecting back to the client to do callbacks, the server can choose to connect to the gateway if the callback object reference has the necessary information. The server can send callback messages to the gateway. The gateway will forward the messages to the client.

5. SYSTEM ANALYSIS AND DESIGN OF DISTRIBUTED GIS
Because of the Object-oriented characteristic of CORBA object, the main system analysis method is object-oriented. At first, skeleton structure of the system is necessary. Then IDL Interface file should be generated. After deep analysis of user demand, the main relationship of objects and operations can be abstracted IDL (Interface Definition Language) is adapted for describing the relationship. Because CORBA realizes the communication function, developers can just focus on developing your application. The main develop steps are as follows:
- analysis the system demands. The main demands include the main system function and the kind of the clients, the distribution of spatial data, the update rate of database and the amount of maximum clients.
- building the abstract model of the system. In this part, the relationship between system functions and decide how the functions are divided into the different ORBs.
- building the object model. According to the Object-Oriented design method, objects can be described with UML (Unified Modeling Language).
6. CONCLUSION

A distributed GIS is quite different from a stand-alone GIS. For interoperability, the new GIS platform need conform the specification of OpenGIS. So the spatial reference system and spatial data interchange is necessary. Spatial query and spatial data retrieve is important for domain application. GML is a new method for spatial data storage. The integration of CORBA and GML is the future of distributed GIS. Then the method of building a distributed GIS with CORBA is introduced.

REFERENCES


