Online GIServices

C. Vincent TAO

Canada Research Chair in Geomatics
Director of Geospatial Information and Communication Technology (Geo-ICT) Lab
York University
4700 Keele Street, Toronto, Ontario, Canada M3J 1P3
Tel: (416) 736-5221, Fax: (416) 736 5817, E-mail: tao@york.ca

Abstract

GIServices means geodata services and geoprocessing services. Most on-line GIS service systems provide centralized geodata services based on a client/server computing architecture. These systems allow users to access geodata sets from remote data repositories. With the increasing availability of geodata sources, an interesting issue is raised: can we provide users with a system allowing them to access and integrate geoprocessing functional tools over the network? This paper introduces a new technology "GIServices" and reports recent progress made toward this research direction. A distributed geoprocessing service model is designed based upon an open object architecture and a prototype system, "*GeoServNet*", has been developed. It allows users to assemble, access or "rent" geoprocessing components that are distributed across a network via any standard browsers. With this model, both distributed data and GIS processing recourses can be integrated in a way that users are able to access GIServices on the Internet.

1 GIServices vs. GISystems

An alternative development approach to GISystems would be a service-oriented approach that allows users to access, assemble, and "rent" geoprocessing components that are distributed across a network via standard Internet browsers (Tao and Yuan, 2000a, 2000b). In such a scenario, users make their data available to a GIService agent, and pick and choose the right tools to process their data sets. Users also have a full control over the entire data process. The processing can be done either remotely on a host server or locally on the user's machine. The GIService agent provides a common interface with which the geoprocessing functional components can be registered and made available. With this architecture, both data and geoprocessing resources distributed across the network can be accessed, assembled and even integrated by network users.

Primarily, there are three reasons for developing GIServices rather than GISystems:

From the Technological Point of View

The impact of the network-centric computing on GIS is unprecedented. From the technological point of view, GIS architecture has evolved: (a) From mainframe GIS to desktop GIS; (b) From desktop GIS to network-based (Internet/Intranet) client/server GIS, and (c) From client/server GIS to ubiquitous distributed GIS.

With the increasing availability of abundant and cheap ubiquitous computing devices, wireless/mobile/nomadic network access, and autonomous and intelligent software proxy agents, ubiquitous geocomputing in a heterogeneous network environment defines the trends of the next generation of GISes. Therefore, it is of great value to research the new models that support collaborative computing and electronic persistent presence (EPP).

Internet based geocomputing is just getting started. It will enable geoprocessing services delivered over the Internet or an Intranet. The existing web-based GISes that mainly support geodata services can not provide the users with spatial processing services such as spatial data processing, spatial analysis and modeling.

Journal of Geospatial Engineering, Vol. 3, No. 2, (December 2001), pp. 135-143.

From the Software Engineering Point of View

It is a fact that the majority of GIS users only use a small fraction of the functionality offered by GISes. However, users still need to pay full licensing fees in order to get the system installed in their office. On one hand, GIS vendors are developing more and more functions to make their systems powerful and versatile. While on the other hand, users are asked to pay the full cost for covering the vendors' investment even though their processes only need to make use of a small portion of the system functionality. It has been estimated by Microsoft that 90% of Excel's functionalities are used by only 10% of its users worldwide.

With regards to the GIS community, it is even worse due to the high price of the software licensing as well as the considerable cost of GIS training and software maintenance. It is also very common, in reality, that if one software package does not work well for one function, another software package needs to be purchased to complete a job adequately. This is understandable, since one pair of shoes can not fit all size. Each vendor has its own strength and targeted applications, and therefore, can not always work well in all GIS arenas.

By looking at this business model, one may ask whether there is a way of changing this situation where the cost of purchasing software by the GIS community is substantial as is the investment by vendors to create packages that attempt to solve all problems. With the advent of software component technology, it is possible that a large proprietary GIS can be broken into many interoperable functional components. These components can be assembled and integrated by users to meet their own requirements. The key concept of Web-based **GIServices** is that the multi-function software offered by a traditional GIS vendor can be delivered through the assemblage of many interoperable geoprocessing components over a network.

From the Users' Point of View

With the geoprocessing services available on the Internet, users are able to choose and use the geoprocessing components based on their own needs. The software tools can be "rented" easily through the Web. This will also allow the user to build a customized system by assembling only the necessary components. Various e-commerce arrangements such as registration for access, software metering, etc. can be used. On one hand, this benefits the users in term of cost savings (paying per needs), and on the other hand, it will generate more profits to the vendors due to the massive access and purchase of services as well as the reduced overheads with respect to maintenance and customer support.

A simple case of the use of these geoprocessing services is the manipulation of data sets that are located at a client's site. If the user needs a geoprocessing tool, for instance, a data conversion tool, to process the data, then there is no reason for the user to purchase a full license for a GIS package since the user only needs a few modules of the complete system. Utilizing an online geoprocessing services system, the user could login to the URL and "rent" the related modules to process the data and then store the results locally.

2 System Architectures for GIServices

The technological basis for the development of GIS service systems is the networked communication infrastructure, as well as distributed object computing technologies. The general architecture of a GIS service system is illustrated in Figure 1. From the client/server computing point of view, there are three approaches to develop geoprocessing services.

Thin Client Architecture

The client site only provides a user interface that allows the user to send a service request. All the computing work is done on servers elsewhere in the network. This model supports various "network" devices, such as desktop PCs, laptops, palmtops, PDAs (personal digital assistants), cell–phones and other information appliances. This architecture suits applications where the working environment is mobile, the communication methods are mainly wireless or require relatively narrow bandwidth.

Thick Client Architecture

The client site becomes the primary resource for computing, but the geoprocessing components come from servers elsewhere in the network. The client site provides a workplace that supports local processing of data sets. This architecture works particularly well in an environment where data sets are mainly stored at the local site. Due to the fact that the size of spatial data sets is normally high, it is not efficient for the data to be transferred to the server for processing. Therefore, transferring functions to the local site for processing local data would have a clear advantage.

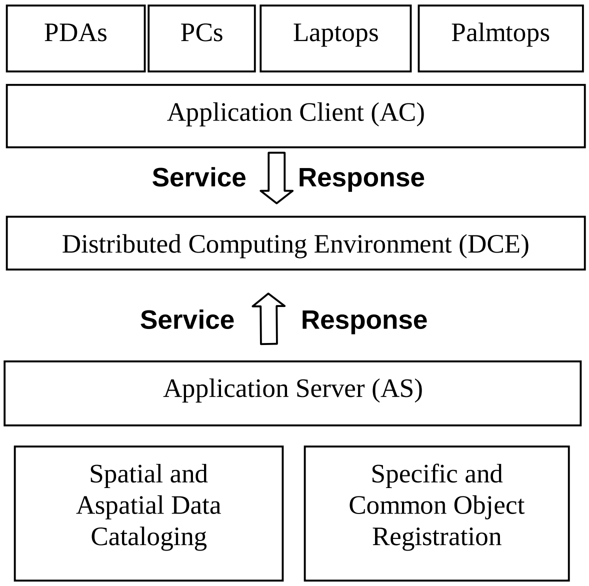


Figure 1 A DCE-based architecture for GIServices

Smart Client Architecture

With the advent of the distributed object computing technology, an optimized configuration of computing resources can be achieved. This results in a balanced thin/thick client or smart client architecture. In this architecture, the role of a client or a server is changeable. A client can be a server and a server can become a client. Depending on the bandwidth as well as the location of the data to be processed, the geoprocessing components can be assembled either on the client side or on the server side. Computing can be performed on both sides in a collaborative manner. This is an advanced distributed computing architecture that requires a more careful design.

3 Geoprocessing Data Models for GIServices

In a distributed environment, the geodata objects will provide clients and servers with the basic understanding of the geodata during the communication, as the geo-objects are passed in the network as parameters. The following conceptual data model defines the basic geospatial types and their relationships, as illustrated in Figure 2.

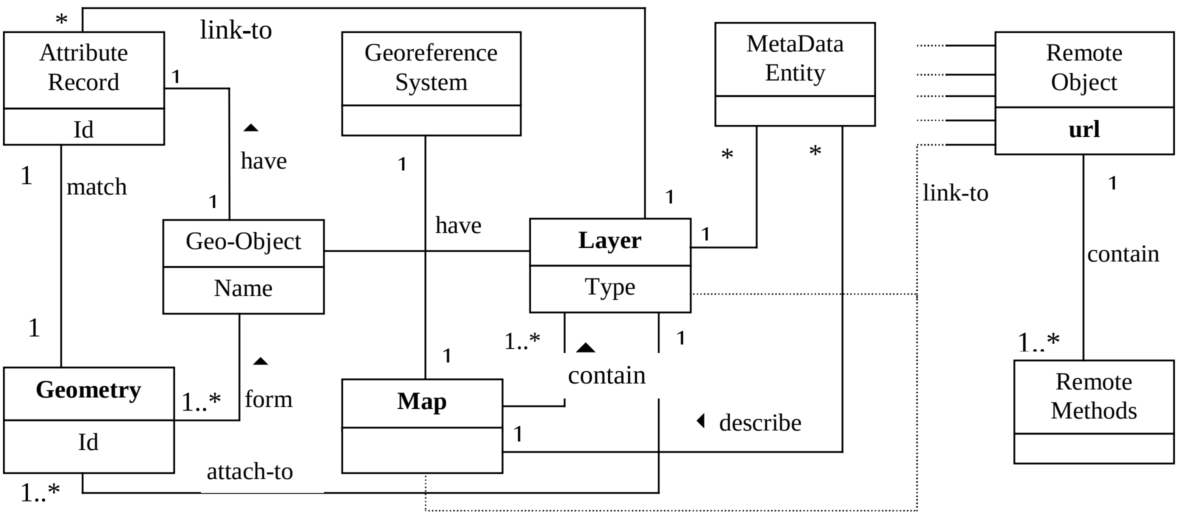


Figure 2 A partial abstract geospatial data model for the prototype distributed GIS

In this model, a Map/Layer/Geo-Object structure is used and basic geodata types are defined, such as geospatial geometry objects (Point, Polyline, Polygon, etc.), geospatial reference system objects (Datum, Ellipsoid, Projection, etc.), Metadata entities (Accuracy, etc.), and so on. A Geo-Object type is defined to integrate the geometry and the corresponding attribute records into single geospatial objects, such as a river which contains many sub-rivers or a city that may contain a lot of sub-objects. The geospatial reference system is applied to Map only since layers in one map usually have a unified reference system. Both Map and Layer objects may have Metadata entities to describe the data quality and other meta information of the map or layers.

Unlike a desktop GIS data model, remote objects are included in the data model in order to support remote communications. Remote objects that link to the internal geo-objects (such as Map or Layer) provide remote methods to clients. The dotted lines that link to the remote object in Figure 2 are only examples of the relationships between remote objects and internal objects. In fact, any internal object may connect to the remote objects if the remote methods are required.

Spatial data models define the way that the spatial data is organized and manipulated. There are many GIS data models in use, among which the geometry-centric geo-relational GIS model is the most popular one (Gardels, 2000). In this model, the geometry features, such as points, lines, and polygons, are stored in file-based systems while the associated attributes are maintained in a relational database. Data integrity has been the major concern of this model due to the separated management and organization of spatial data elements and attribute data elements.

A clear trend is to develop an object-based data model, in which a geographic object is defined by geometry features, attributes as well as its behaviors. Figure 3 shows a model that has been implemented in our prototype system (Tao and Yuan, 2000c). The model is designed based on the OpenGIS Abstract and Implementation Specifications (Open GIS, 1998). In the model, a map-layer-geometry structure is adopted. The attributes that are stored in an external relational database are bounded to a GeoRecord object that aggregates into a collection object, GeoRecordSet. The model also defines the reference system object (datum, ellipsoid, projection, etc.), metadata entity (accuracy, source, etc.) as well as object methods (behaviors).

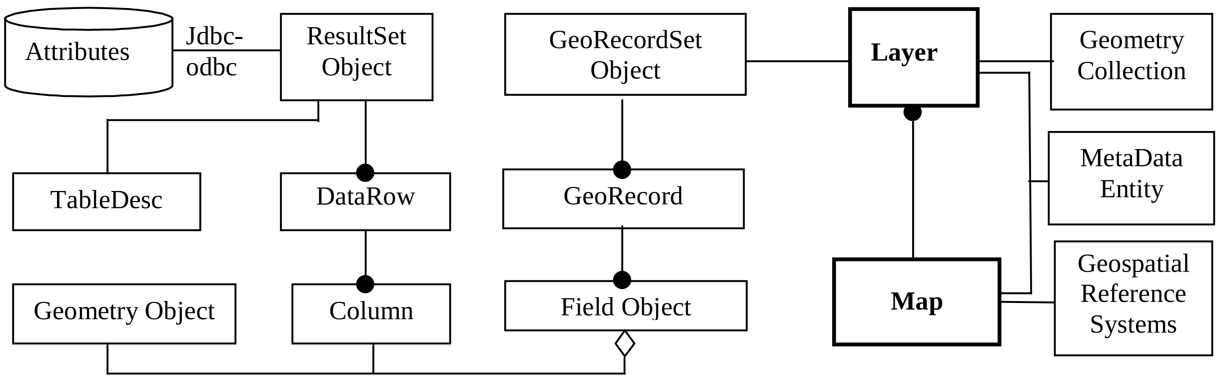


Figure 3: An integrated geospatial and attribute data model

4 Implementation of GeoServNet – A System for Online GIService

Geoprocessing Services – A Java Implementation

Following a comparative study of existing distributed object computing technologies, Javasoft's Java Remote Method Invocation (Java RMI) along with JavaBeans and Java Applet methods were used in the implementation of the system. On the server side, the system consists of a web server, a component registration server, a data catalog server and many geoprocessing servers. With this model, geoprocessing components can be distributed anywhere on the Internet. Any machine connected to the Internet, that can provide geoprocessing components, is treated as a server. However, the geoprocessing components need to first be registered with the registration server.

The registration server provides the registration information to the applet client through RMI, and then the applet client can make a remote call to the geoprocessing servers.

The component registration is a core part of the model. It allows users to choose and integrate the available geoprocessing functions (components). Without this registration, the remote method invocation will be static and fixed by the programming in advance.

Component Registration Model

Similar to the data cataloging, geoprocessing component registration also needs component meta-information to support distributed function access, such as functionality description, version, location (IP, URL), port number, remote geodata connection (database URL), provider's information, component credit, etc. Java servlet techniques provide a solution to the component registration model. Figure 4 illustrates a component registration model.

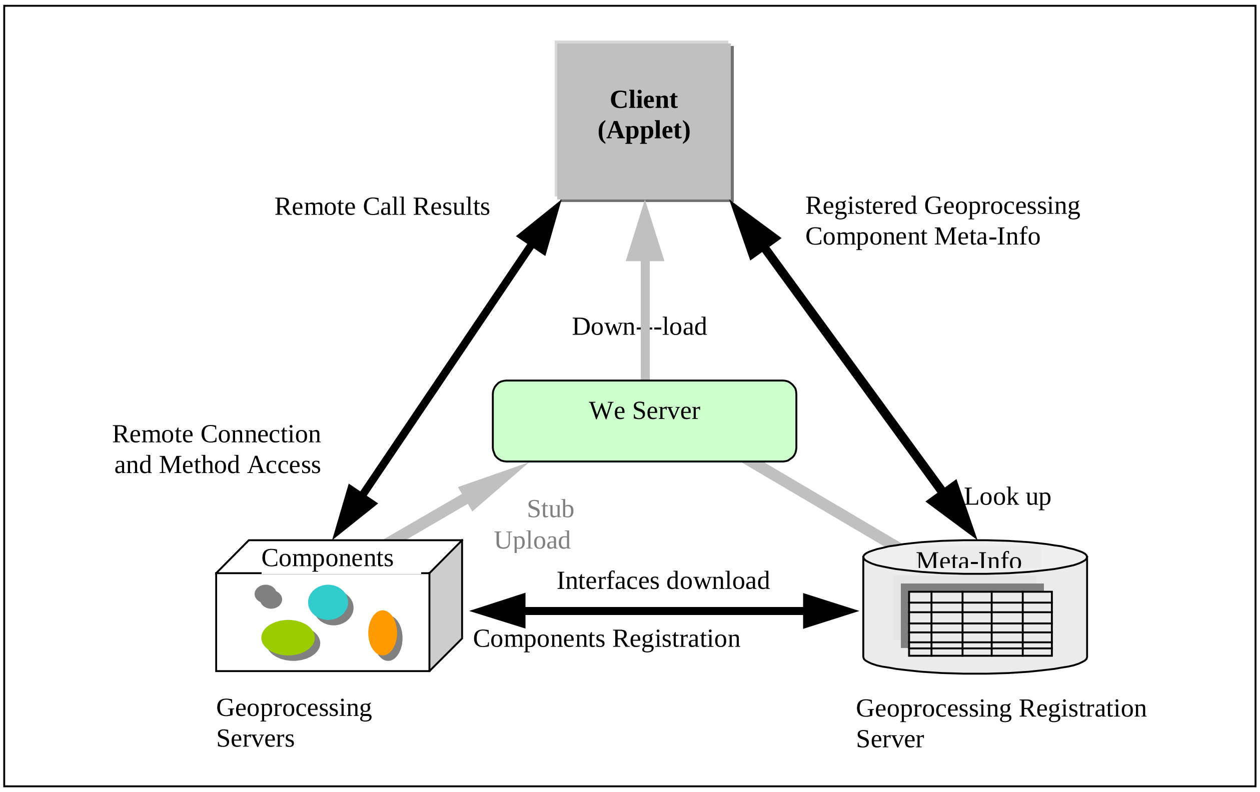


Figure 4 Relationships of the client and the servers in the geoprocessing registration model

In this model, the web server and the geoprocessing registration server are located at the server site and are bound together. External geoprocessing components can be connected to the server by registering the meta-information of the components into the service categories, the component meta-information database, before the components are accessed. Any client can look up the component meta-info database and connect to the components they chosen according to the meta-information stored in the meta-info database.

Similar to geodata catalog services in which geospatial metadata plays a key role (FGDC, 1998), meta-information of geoprocessing components is also of particular importance to the development of geoprocessing services. It provides fundamental information that allow software components to be integrated and assembled in a distributed computing service system.

Meta-information of geoprocessing components should contain two types of information:

a) its software component interface so that it can be identified by other components and can be integrated with others
b) the user identification information to allow users understand what it does, how it works, what is quality, as well as its related components, etc.

The first item is inevitable from the software engineering perspective, and the second is necessary to help users integrate and assemble the components.

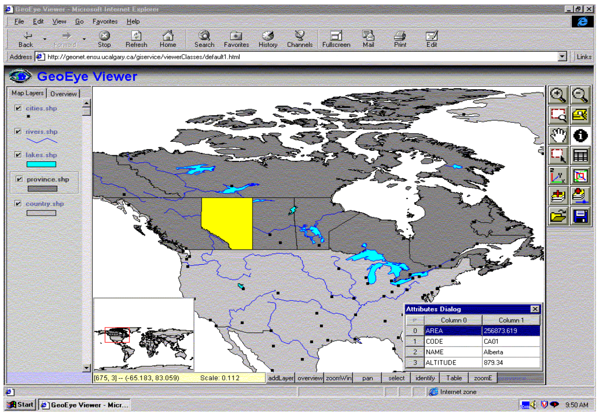
Application Examples

Some of the service examples available from the GeoServNet are given below:

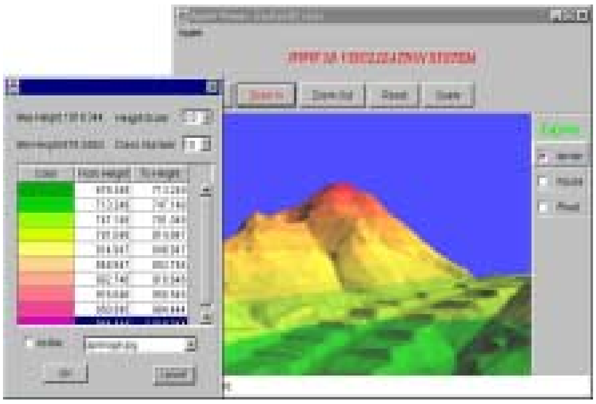
a) Geospatial Display Services (GeoEye): This service is fundamental to other services. Viewing and simple querying of data has been considered the most useful function in GIS. A fully developed Java geospatial viewer (GeoEye) supports most display functions, such as zoom, pan, select, identify, layer control, color style control, etc. Figure 5a shows a user interface of the system.
b) Geodata Access Services: This system allows the user to download data from servers or upload data from the local site. It also allows the users to overlay data layers coming from different servers or local machines. This is a very useful data access mode since in reality, data could come from anywhere in a network (Figure 5b, data access services)
c) Data Transformation Services: Some of the popular transformation functions have been implemented. To date, GeoServNet supports transformation between geographic coordinates and projected UTM coordinates. The transformation parameters such as unit (linear and angular), datum (NAD27, NAD83), map projection, etc., can be selected by the user.
d) Map Annotation and Symbolization Services: In cartographic mapping, map annotation and symbolization are of particular value for many on-line service applications. A novel algorithm has been developed to rapidly determine the centroid of any polygons so that attributes of the polygon can be annotated. This algorithm was also used for map conflation where layers that come from different sources do not match (Yuan and Tao, 1999).
e) Data conflation: Conflation is the process of combining the information from two or more geodata sets to make a master data set that is superior to either source data in either spatial or attribute aspect. This service is of particular importance to the distributed geocomputing environment where heterogeneous data sources are accessed and integrated for processing and analysis.
f) Terrain Analysis Services: It does not matter if the raw point data is located on a server or on the client's local machine. Various interpolation algorithms along with shading and profile analysis algorithms are provided. The user is able to compare different interpolation algorithms and chose the best for their application needs (shown in Figure 5c).
g) 3D Visualization: 3-D visualization functions that are traditionally only supported by high-end computers have also been developed in GeoServNet. Both terrain data as well as CAD-type data such as 3-D city models can be visualized on the net, as shown in Figure 5d. It is important that these 3-D models are built based upon a 3-D GIS concept in order to allow objects and layers in 3-D scenes to be queried and manipulated (Tao and Wang, 2000).

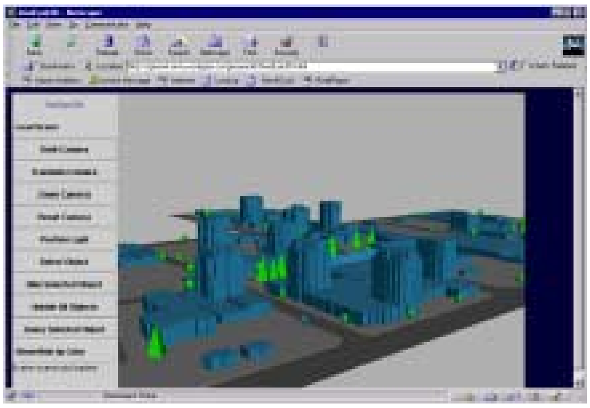
5 GIServices for Ubiquitous Geocomputing

As shown in Figure 6, the GeoServNet can also be deployed in a wireless network environment. If data sets are residing in a field computing device such as Palmtops or other handheld PCs (HPCs), the user could download the geoprocessing components on-line and process the data in the field. In this case, we simply install the GeoServNet on this field computing device. The advantage of this setup is that low power portable computing devices become much more powerful as they support on-line geocomputing. No installation of any large software system is required. The integration of GeoServNet with a wearable computer is under investigation (Tao and Yuan, 1999).



(a) Data display services **(b) Data access services**





(c) Terrain analysis service **(d) GIS-based 3-D Visualization services**

Figure 5 Various types of GIServices

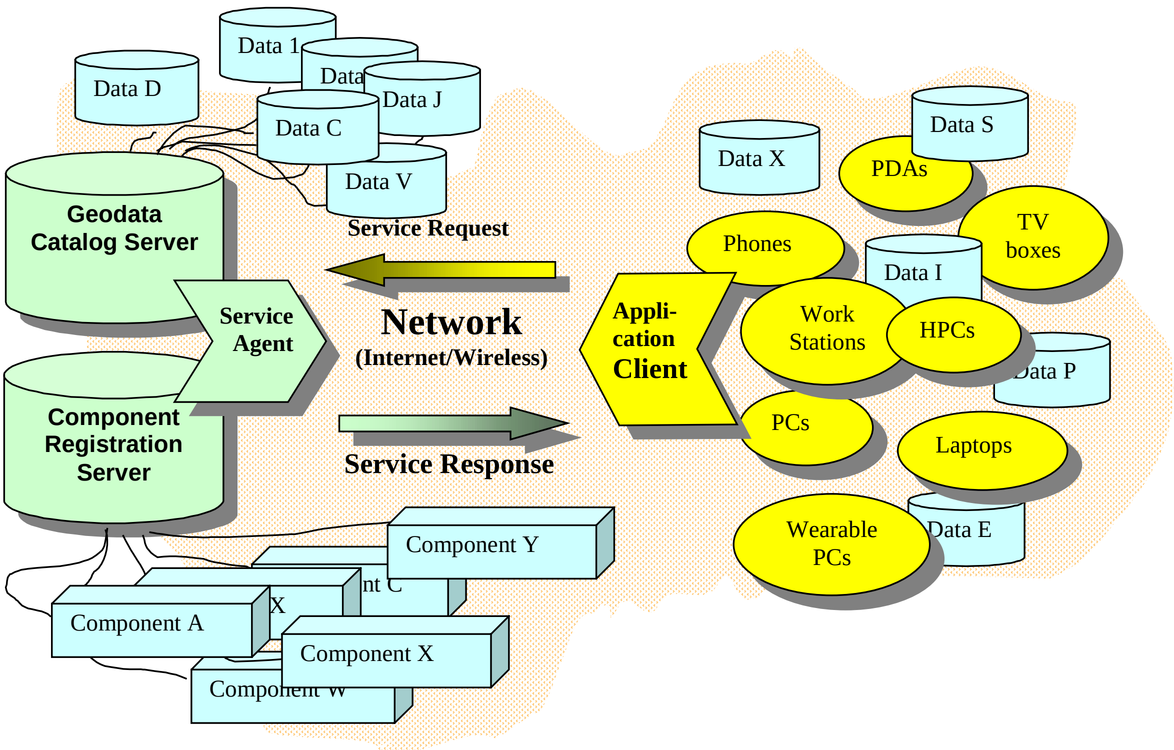


Figure 6 The Framework of the Online GIServices

The recent progress is to integrate a mobile wireless communication component with our architecture (Hunter, 2000). A typical architecture to support this work flow can be Java and XML based (see Figure 7). A Web Server would receive requests from a WAP or Palm Proxy and dispatch these requests for processing via a Java based servlet engine that acts as a gateway to a back-end distributed system. The servlet engine would be responsible for hosting application servlets and providing them with a standards-compliant servlet interface. An advantage of using a servlet engine instead of running the servlets inside the Web Server process itself, is that the servlet engine will provide a standard servlet API and decouples processor servlets from specific implementation details of the Web Server. This increases architectural flexibility as it allows the Web Server to change without impacting the overall system. The servlet engine can also provide various management features that help to shift the load of servlet processing away from the Web Server.

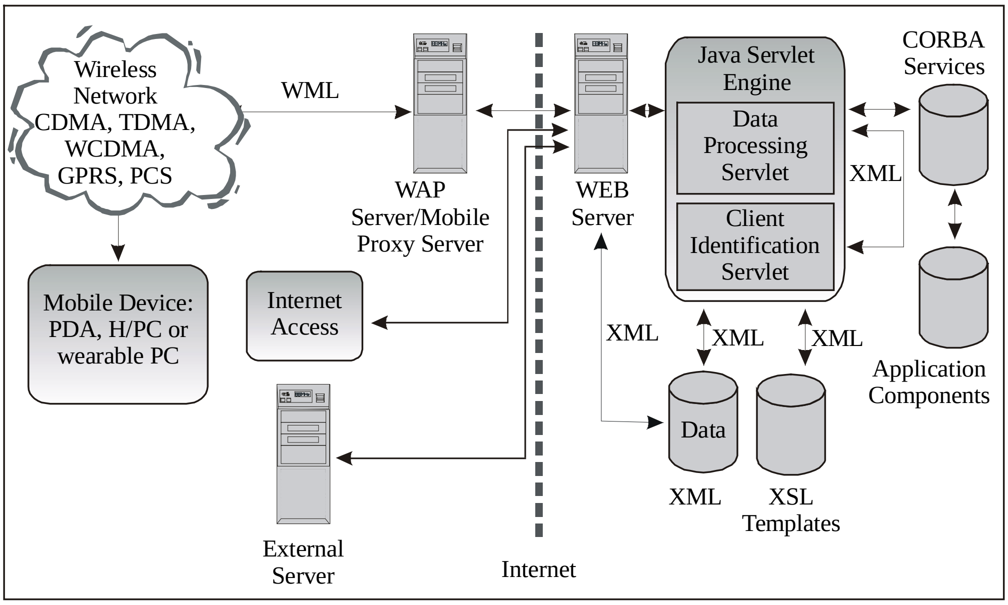


Figure 7 Mobile Distributed Computing Architecture (Hunter, 2000)

6 Conclusion and Discussion

The concept of GIS services changes the way GIS users can use GIS, the way GIS vendors will sell GIS and the way GIS information providers can supply spatial information. A distributed geoprocessing model must be developed in a way that the system is open and interoperable in comparison to traditional client/server models. The Open GIS based data model design and APIs development as well as component registration are some of key issues with respect to the implementation of a GIService system.

Technical challenges also include distributed transaction techniques, distributed spatial algorithms and rapid system response. In order to maintain consistency of data, transaction control will become significantly more important, especially in cases where multiple clients are working on the same geodata set in a collaborative manner. Another issue is the optimized design of geoprocessing components so as to minimize the response time of remote object invocation. Therefore, research on the distributed spatial handling algorithms needs to be emphasized.

Acknowledgements

The research is sponsored by the Natural Science and Engineering Research Council of Canada (NSERC), and Geoide Network Centre of Excellence (NCE) Research Grant. The author would like to thank S. Yuan, Q. Wang, W. Yang, C. Fei, A. Hunter, W. Akehurst, N. Shahriari, etc., and other research fellows and graduate students who have contributed to this research.

References

Abel, D., 1998. Towards integrated geographical information processing, *International Journal of Geographical Information System*, Vol. 12, No. 4, 1998, 353-371.

FGDC, 1998. *Content Standard for Digital Geospatial Metadata*, FGDC-STD-001-1998, http://fgdc.er.usgs.gov/ standards/documents/standards/metadata/v2_0698.pdf

Gardels K., 2000. *A Comprehensive Data Model for Distributed, Heterogeneous Geographic Information*, http:// www.regis.berkeley.edu/gardels/geomodel_def.htm

Gunther, O, Muller, M., Schmidt, P., Bhargava, H., and R. Krishnan, 1997. MMM: A WWW-based approach for sharing statistical software modules. *IEEE Internet Computing*, 1(3), http://computer.org/internet/

Hunter, A, 2000. The Road to Ubiquitous Geospatial Information Systems: Roam Anywhere - Remain Connected, *Presented at The 12th Annual Colloquium of the Spatial Information Research Centre*, University of Otago, Dunedin, New Zealand.

Laurini R., 1994. Sharing Geographic Information in Distributed Databases. *Proceedings of Conference on Urban and Regional Information Systems Association*, Washington DC, pp. 441-454.

Müller, R., 1999. From GISystems to GIServices: Spatial Computing on the Internet Marketplace, *Intermediaries for Information Services*, Humbolt-Universität zu Berlin, October 1999, 195-211.

Open GIS Consortium technical committee, 1998. *The OpenGIS Guide*, Draft, Fourth edition, http://www.opengis.org/techno/guide.htm.

Tao, C. V. and Yuan, S., 1999. Speech Recognition and Its Implications to GIS Data Collection, *Proceedings of 1999 ASPRS Annual Conference*, Portland, Oregon, May 17-21, 1999, pp. 1060-1069.

Tao, C. V. and Yuan, S., 2000a. GeoServNet: Renting Geotools over Internet, *Geo-Informatics*, Vol. 14, No. 12, pp 12-15, 2000.

Tao, C. V. and Yuan, S., 2000b. Development of Internet-based GIService, *Proceedings of GIS 2000 conference (CD-ROM)*, Toronto, March 14-16, 2000.

Tao, C. V. and Yuan, S., 2000c. Development of A GIS Service Model In Support of On-Line Geoprocessing, *GITA XXIII conference Proceedings (CD-ROM)*, Denver, March 27-29, 2000

Tao, C. V. and Wang, Q., 2000. Web-Based 3-D GIS and Its Applications for Pipeline Planning and Construction, Proceedings of International Pipeline Congress (CD-ROM), Calgary, October 2-5, 2000.

Yuan S. and Tao C. V., 1999. Development of Conflation Components, proceedings of Geoinformatics'99, *International conference on Geoinformatics and Socioinformatics*, Ann Arbor, Michigan, USA, June 19-21, 1999, 579-591.