1998 * Tsou, M. and Buttenfield, B.P. Client/Server Components and Metadata Objects for Distributed Geographic Information Services. **Proceedings**, GIS/LIS '98, Fort Worth, TX, November, 1998: 590-599.

CLIENT-SERVER COMPONENTS AND METADATA OBJECTS FOR DISTRIBUTED GEOGRAPHIC INFORMATION SERVICES

Ming-Hsiang Tsou and Barbara P. Buttenfield

Department of Geography, University of Colorado, Boulder, CO 80309 USA
E-mail: tsou@ucsu.colorado.edu, babs@colorado.edu Phone: +1-303-492-3684

ABSTRACT

The need for global access and decentralized management of geographic information is pushing the GIS community to establish an open GIS architecture and provide distributed geographic information services. From an operational perspective, the role of client/server components underlie specification of task-oriented programming, and the modularization of GIS software. Exchange of geographic information services cannot happen without development of metadata strategies for exchange of processing modules. This research proposes a flexible and dynamic Client/Server relationship in the context of Lego-like distributed GIS components, which can be moved, combined, and used in distributed network environments. Derived from generic GIS tasks, four representative client-side GIS components and two server-side GIS components illustrate the balance of functionality between client and server components. An object-oriented metadata scheme is proposed to formalize description of GIS operators as well as geospatial data sets. The metadata scheme introduces two new types of metadata for GIS components: system metadata and GIS operator metadata, which describe GIS component behaviors and specify data requirements of specified GIS operators. Distributed GIS components become reusable, modularized, self-described, and self-managing with the collaboration of system metadata and GIS operator metadata. The use of operational metadata objects is the key to interoperability and plug-and-play functions for open and distributed GIS components.

Keywords: GIS components, metadata, distributed geographic information services.

INTRODUCTION

The development of Geographic Information Systems (GIS) is highly influenced by the progress of information technology. The motivations for adopting new technology are derived from the essential needs of GIS users and the GIS community. From a management perspective, there are two main reasons for distributed geographic information services. The first reason is *the globalization of geographic information access*. Currently, federal agencies are facing the problem of how to make information available to the public and meet research needs via effective and efficient methods. Traditionally, geographic information has been distributed via paper maps or off-line disks or tapes, which are costly and difficult to update. By utilizing the

Internet and the rapid growth of network communications, the GIS community will be able to provide on-line, distributed geographic information services on the Internet accessible to everyone in the world in a fast and economic way.

The second reason for distributed geographic information services is *the decentralization of geographic information management*. More and more GIS applications and projects focus on a large scope of spatial problems and deal with huge databases. Many federal institutions and agencies are facing the problem of managing such huge databases. Huge and bulky GIS databases cause serious management problems, including maintaining, updating, and exchanging geographic information. Therefore, federal agencies are looking for new ways to more widely and effectively disseminate data, primarily via the Internet (Jones, 1997). On-line, distributed geographic information services under an open and distributed GIS architecture will facilitate the decentralization of colossal GIS databases. One advantage is that the update and maintenance of specific geospatial data sets may be more appropriately associated with one site rather than another. Another advantage is the increased reliability, where failure at one site will not mean failure of the entire geographic information services (Worboy, 1995). Thus, an open and distributed GIS environment will improve the efficiency of GIS database management.

In the GIS community, many research projects conducted by both academia and industry have begun to focus on how to provide distributed geographic information services to the public and researchers (Huse, 1996; Zhang and Lin, 1996; Plewe, 1997; Buttenfield, 1997). However, the current main problem for distributed geographic information services is the lack of a free communication environment. The GIS community needs to deploy a comprehensive architecture for distributed geographic information services. Several organizations and GIS projects currently are focusing on the integration and interoperability for open and distributed geographic information services, including Alexandria Digital Library Projects (Buttenfield and Goodchild, 1996), ISO/TC211 (Ostensen, 1995), and the Open GIS project (Buehler and McKee, 1996). This paper extends the basic concepts of these projects, then defines the roles of Client/Server components and proposes a metadata objects scheme by addressing two major problems in the development of distributed geographic information services.

PROBLEMS IN THE DEVELOPMENT OF DISTRIBUTED GIS

The first problem in developing distributed GIS is the lack of an architecture which can provide a logical construction of systems. Most current on-line geographic information services adopt a quick, ad hoc, technology-centered approach to provide a “temporary” solution. Once the technology changes, everything in the previous system is abandoned and a whole new system has to be designed and implemented. Without an open architecture, distributed GIS would not be widely used by the GIS community due to the short-term life cycle and rapid change in information technology. A comprehensive architecture will facilitate the development of open and distributed GIS from an ad-hoc, short-term strategy to a long-term, logical, and sustainable development strategy.

The second problem is that current development of open and distributed GIS mainly focuses on the database issue and data interoperability. However, GIS systems are both data-oriented and task-oriented. Besides the consideration of data interoperability, the GIS community needs to deploy the architecture from an operational perspective, which includes the definition of Client/Server component relationships and the interactions between data objects and GIS operators. Without taking GIS operations and practical use into consideration, the data interoperability may become unrealistic and have some implicit defects when being applied in the real case.

SERVICES AND METADATA IN AN OPEN ENVIRONMENT

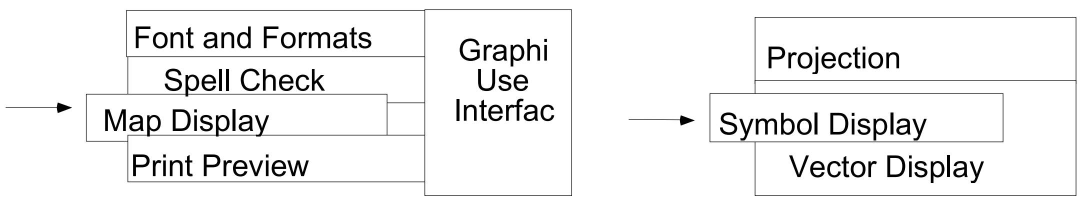
From an operational perspective, two major issues must be addressed to establish an open GIS architecture in distributed network environments. The first issue is the definition of Client/Server relationships delivering heterogeneous services. In distributed network environments, the major obstacle is the integration of services. A key issue for the integration is the development of "independent" client-components and server-components along with formal definitions of their interactions and relationships. The second issue is the formalization of metadata for GIS operators, which incorporates comprehensive description and functionality. In distributed network environments, data sets and operators are more dynamic, transportable, interoperable. A comprehensive formalization of both geospatial data and GIS operators will facilitate the effective and correct use of geographic information services. The content of formalization will include the identification and collaboration of distributed data objects and GIS operators. This paper proposes task-oriented distributed GIS components and an object-oriented metadata scheme for open and distributed GIS.

Task-Oriented Distributed GIS Components

Currently, both academic and industrial studies of on-line distributed systems are focusing on “distributed components” in open environments which can provide new capabilities for the next generation client/server architecture (Montgomery, 1997). Common Object Request Broker Architecture (CORBA) developed by the Object Management Group (OMG) and Distributed Component Object Model (DCOM) developed by Microsoft Corporation are two well-known examples of distributed component infrastructure (Orfali and Harkey, 1997). In both, distributed component object servers adopt the concepts of object-oriented modeling (OOM) and a distributed computing platform (DCP). In general terms, a distributed component is "a ready-to-run package of code that gets dynamically loaded into your system to extend its functionality" (Pountain, 1997: 93). JAVA applets, ActiveX controls, or even plug-in functions for the Web browser can be called "distributed components". In principle, the features of distributed components are modular, interoperable, portable, reusable. They are both self-describing and self-managing. (Orfali at al, 1996; Pountain, 1997)

In practice, distributed components are "Lego-like" pieces of binary code, which can be ported, combined and used in distributed network environments. The most important advantage for distributed components is independence from operating systems, hardware-platforms, network environments, vendors, and applications. The development of distributed components shifts the software paradigm from a monolithic, feature-heavy approach to a flexible, modularized, and plug-and-play approach, which dramatically improves the cycles of program development and efficiency of software engineering.

Figure 1 shows an example of a distributed map display component which can be used (for example) in a word processing application or a GIS package. The word processing application is combined with several distributed components, including a graphic user interface component, a spell checker, and so on. The map display component should be independent to the extent that it can be easily plugged into other packages when users need a map display function. Moreover, the component strategy is hierarchical. Here, the map component is made up of sub-components, including (in this case projection control and vector display controls. Alternative sub-components can be added into a map display component to extend its display functions, such as adding a symbol display control.



A Word Processor Package *A Map Display Component*

Figure 1. The use of distributed components

Advantages of Task-Oriented Components

Task orientation provides a flexible and dynamic Client/Server relationship. Distributed component technology will allow different clients to access heterogeneous servers, which is an essential feature of an open and distributed GIS architecture. Distributed client-components and server-components can freely interact and inter-operate on the Internet (Figure 2). This allows users to select software modules from multiple GIS packages, intermixing these modules or even merging (for example) an ARC kriging process to an Intergraph MGE scale-changing routine, and integrate those procedures into IDRISI to merge with a Thematic Mapper classification routine. The principle is that various software modules reside on multiple servers, allowing clients distributed on local networks or on the Internet to access only the modules needed for a particular application. Users can select modules that work most efficiently, contain the most flexibility, the smallest disk overhead, etc. In this way, distributed component technology provides an effective and efficient mechanism to individualize and/or streamline one's software, according to one's tasks.

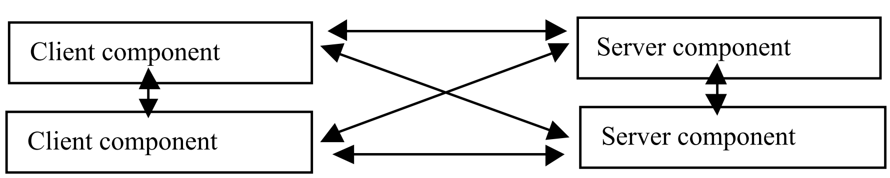


Figure 2. The interaction between client-components and server components

Based on generic GIS tasks, we propose four classes of client-side GIS services.

- Spatial and text-based query services
- Map display services
- Data download, pre-processing, and integration services
- On-line GIS processing and spatial analysis services

We propose two classes of server-side GIS services.

- Database clearinghouse services
- GIS engine services

These GIS services can be freely combined and used in an open and distributed GIS architecture, where GIS users build their specified, task-oriented GIS based on their needs. The availability of services must take into account issues of interoperating with a "thin client" or "thick client". In networking terminology, the thick client is defined as having operations and calculations executed on the client-side. On the other hand, a "thin client" may require that selected operations run on the server-side. Whether the client-side GIS component should be thick or thin will depend on the task and associated performance requirements. For example, it may be appropriate to use thick clients for map display services to let the user take over the many intuitive decisions of graphic design, layout, etc. Alternatively, network routing or location modeling may be better off to run on the server side because complicated calculations and algorithms may be more efficiently handled by the server, without an intervening network. The role of client and server components should be dynamic and changeable. The balance of functionality between client services and server components will be a critical issue for the success of open and distributed GIS.

Additional advantages are also evident. In traditional GIS software, 90% of users use less than 10% of an application's features. These users must nonetheless pay for the full monolithic software suite, as opposed to licensing only those modules they require. The monetary issue is underscored by the computational overhead, disk space requirements, difficulty of installation, etc. The remaining 10% of advanced users requiring more complex features are dependent upon version update cycles that dictate when new features become available. It is probable that software licensing for individual task components, on a short- or long-term pricing structure, will begin to be available as open GIS architecture becomes commonplace.

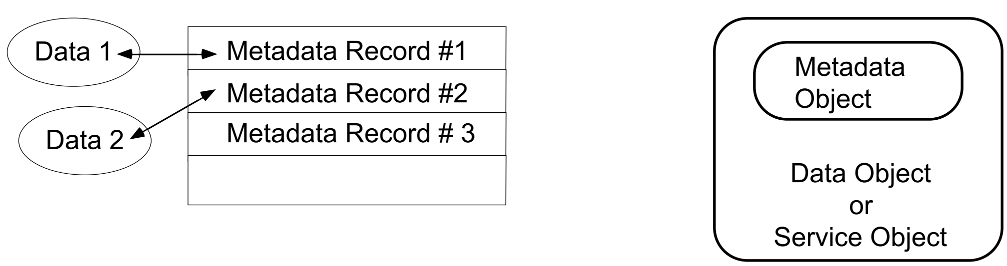
An Object-Oriented Metadata Scheme

For distributed geographic information services, metadata is the information that supports the exchange of processing operations between client and server. (OGC, 1998) In general terms, metadata describes the content, quality, condition, and other characteristics of data. The major uses of metadata include: 1) organizing and maintaining an organization's investment in data; 2) providing information to data catalogs and clearinghouses; and 3) providing information to aid data transfer (FGDC, 1995). Little research has reported on formalization of metadata to distribute GIS operators, and geographic information services.

This research proposes to adopt an object-oriented metadata scheme to solve the problem of the formalization of metadata for GIS operators in distributed network environments. Currently, many agencies are conducting metadata research (FGDC, 1995); individual research projects also address metadata issues, as for example the Alexandria Project (Smith, 1996). Existing work presents metadata schemes emphasize the establishment of a standardized format and adopt traditional relational database concepts, where each metadata item is represented as an individual record. However, ad-hoc approaches to the metadata issues do not scale and cause problems for interoperability (Baldonado et al., 1997). The standardization of metadata formats may undermine their application to services because it is impossible to design a single standard for heterogeneous geospatial data processing methods. Consider for example how a single standard would be inadequate to simultaneously describe both a TIN data model and a raster data model, without lots of extraneous fields. Likewise, a single service -based metadata model designed to describe both interpolation and buffering would be both cumbersome and inefficient. Thus a single standard for metadata likely will not be feasible.

Recently, the metadata standards developed by ISO/TC211 (Kuhn, 1997; OGC, 1998) already mentioned that geospatial metadata may need some extensions, such as adding to an existing data elements or adding a new metadata element. These metadata schemes detach metadata from its own data, storing them separately in the database. The detachment of metadata and data jeopardizes the availability of metadata when geospatial data sets are frequently moved, downloaded or modified in the dynamic network environment. It is quite possible to lose metadata during data processing and copying. In fact, metadata are always problem- and application-dependent. This research suggests that the scheme of metadata for distributed information should embed metadata object within the data object itself. Figure 3 demonstrates these two different metadata schemes.

There are several advantages for metadata encapsulation. First of all, metadata objects provide a flexible approach to construct metadata according to a specified data model. The adoption of object-oriented modeling methods will change the format of metadata from a traditional standard template to a flexible, dynamic structure. Second, when a user moves or copies geodata objects, metadata will automatically be exchanged. Users will never worry about where to find the metadata for their data objects. Third, encapsulation of metadata information will protect the metadata from outside environments. Only authorized programs can access the metadata information. Moreover, when a new geodata object is generated, the new metadata object can inherit most of its parent metadata



Traditional Relational Metadata Scheme *Encapsulated Metadata Scheme*

Figure 3. Two metadata schemes

information, and then add new metadata information for itself. For example, if a subset-area is "clipped" from a satellite image, the new metadata object will inherit the information about image resources, sensor types, and resolution from the original image and add new spatial boundary coordinates for the new metadata. In a comprehensive metadata object scheme, each data object should be able to automatically generate its own metadata object and encapsulate it into the data object in doing the process. Each geodata object must have its metadata in distributed geographic information environments and metadata objects can be retrieved from data objects and saved in a repository, and be accessed by other application programs.

An object-oriented metadata scheme can be applied to objects defining geospatial data and services. For on-line GIS services, different GIS components will be developed and designed for their specific user tasks and functions. In order to effectively describe them, the component metadata need to be implemented inside each component. Metadata embedded in a GIS service should include two major parts: ***system metadata*** and ***data-operation requirements*** (Figure 4). The system metadata describes available functions, methods, and behaviors for system controls and program specifications. The data-operation requirement specifies the data requirements for the specified operations.

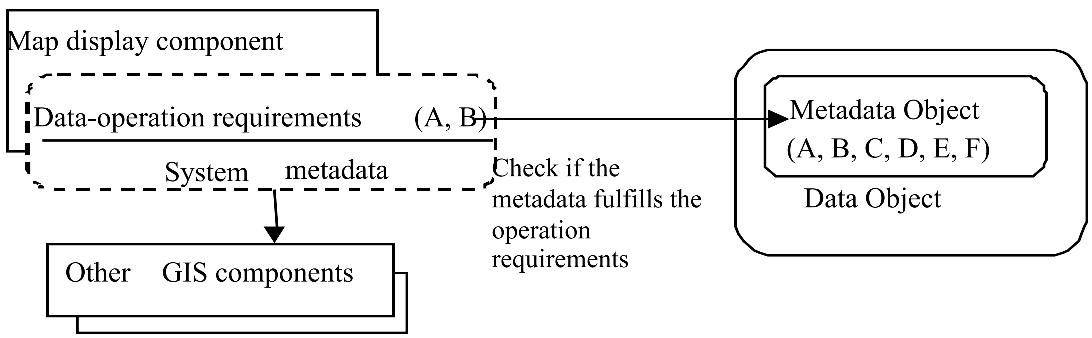


Figure 4. The functions of GIS component metadata.

For example, the map display service will embed system metadata for its "plug-in" function and the collaborations with other GIS components. The data-operation requirement will specify the requirements for the map display function, including "boundary coordinates", "projection method", and "scales". The information will be used to check if the available data object fulfills the requirements of the specified operation. With the collaboration of system metadata and data-operation metadata, distributed GIS components will become reusable, modularized, self-described, and self-managing. The use of GIS component metadata is the key to interoperability and plug-and-play function for open and distributed GIS components.

A FEW IMPEDIMENTS

Several classes of impediments to open-architecture GIS include computational obstacles and telecommunication bottlenecks. These are not trivial challenges to overcome, but they have been discussed at some length by other authors. Standard for data exchange forms another impediment: a lack of robust and flexible data models impedes the ability to fully implement intelligent agents for Internet data harvest, and for full exchange and data sharing via online clearinghouses. Again, these issues are complicated, and covered in detail in other outlets. It is interesting that most of the work on standards focuses specifically on data, and not on establishing standards for GIS processes. This impediment is clearly important for implementation of task-oriented distributed components; and yet it has not been widely addressed in the literature on standards to date.

Standards for Distributed Services

The design of an open system model attempts to solve the problems that arise from a distributed system, where the systems are from different vendors using different data formats and exchange protocols. (Worboy, 1995). The IEEE Technical Committee in Open Systems (TCOS) defines open systems as "a comprehensive and consistent set of international information technology standard and functional standard profiles that specify interface, services, and supporting formats to accomplish interoperability and portability of applications, data, and people." (Ganti and Brayman, 1995, p. 53) Keywords in this quote are "interface" and "services". Examples of the open system model demonstrate elemental distributed services, such as Transmission Control Protocol and Internet Protocol (TCP/IP), X-windows environment, and HTTP. In the GIS domain, an open system model must provide higher level services to support exchange of models and model parameters, automatic metadata construction, error detection, and so forth. Data agents must be able to deliver one data model to a distributed process and receive a different data model to return to the client. Specifications for these kinds of transformations are not yet defined.

Two major organizations of relevance are the Open GIS Consortium, Inc. (OGC) and ISO/TC211. The main task of OCG is the full integration of geospatial data and geoprocessing resources into mainstream computing and the widespread use of interoperable geoprocessing software and geodata products throughout the information infrastructure (OGC, 1998). ISO/TC211 is the Technical Committee tasked by the International Standards Organization (ISO). ISO/TC211 emphasizes a service-oriented view of geoprocessing technology and a balanced concern for information, application, and systems (Kuhn, 1997). The OpenGIS Services Architecture proposed by the OGC follows ISO Reference Model for Open Distributed Processing (RM-ODP). It itemizes a framework of services required for the development and execution of geospatially oriented applications (OGC, 1998). Currently, the specifications for OpenGIS Services Architecture are still under development.

Metadata-Related Issues

The use of metadata can support distribution of GIS data and services, such as data identification, establishment of fitness for use, and auto-transfer functions. A comprehensive metadata structure must incorporate service metadata, identifying for a particular class of service what are the input requirements, computational requirements, anticipated output structures, error detection routines, and abort dialogues. This type of metadata is essential for the future development of open and distributed GIS. However, the complexities of various modeling procedures coupled with a myriad of software specifications, may undermine this application of metadata design. The construction of metadata should be flexible and application-oriented -- such a statement is easy to propose, and very difficult to implement. An alternative approach is to establish "procedure exchange mechanisms" instead of enforcing the standardization of a few metadata structures (Gardner, 1997). These mechanisms may initially amount to middleware, or software patch kits to translate modeling parameters into a system-acceptable form prior to processing. Object data structures seem to promise the most flexibility for distributing geographic information services. Related work includes study of machine-readable features, self retrieval mechanisms, error propagation, and dynamic data lineage (Wu, 1993; Lanter and Surbey, 1994; FGDC, 1995).

SUMMARY

The long-term goal of "geographic information services" is to facilitate the "synergy" of the GIS community by sharing geographical information, spatial analysis methods, users' experiences and knowledge. This research argues for development of an architecture for open and distributed GIS, which will facilitate distributed geographic information services and improve the development of GIS software. By distributing GIS components within a task-oriented processing environment, the community will move beyond the need for single, bulky GIS packages to adopt a streamlined and more efficient computing strategy. Distributed GIS will encourage geographers and spatial scientists to share their analysis methods, spatial models, and geographic knowledge, which in turn could accelerate the accumulation of spatial analysis theories. Such sharing cannot be accomplished without a robust infrastructure formalizing protocols for exchange of both data and services. The use of metadata will become more and more important for bridging the heterogeneous world in distributed network environments. Continued advances in design of metadata models, and automating the collection of metadata, stand as firm prerequisites to the full establishment of interoperable GIS.
Metadata can support portability of services from both functional and technical perspectives. Portability in turn supports interoperability and encourage the whole GIS community to interact with entire new communities and for geographic information to become even more important to a range of human activities (Goodchild, 1996).

Acknowledgments

This paper forms a portion of the Alexandria Digital Library Project, jointly sponsored by NSF, NASA, and ARPA. Funding by the National Science Foundation (NSF IRI-94-11330) is greatly appreciated. Matching funds from the University of Colorado are also acknowledged.

REFERENCES

Baldonado, M., Chang, C. K., Gravano, L., and Paepcke, A. (1997). The Stanford Digital Library Metadata Architecture, *International Journal on Digital Libraries*, 1: pp.108-121.

Buehler, K., & McKee, L., editors. (1996). *The Open GIStm Guide: Introduction to Interoperable Geoprocessing*. Wayland, Massachusetts: Open GIS Consortium, Inc.

Buttenfield, B. P. (1997). The Future of the Spatial Data Infrastructure: Delivering Geospatial Data. *GeoInfo Systems*, June 1997, pp. 18-21.

Buttenfield, B. P., & Goodchild, M. F. (1996). The Alexandria Digital Library Project: Distributed Library Services for Spatially Referenced Data. In *Proceedings of GIS/LIS'96 , Denver, Colorado*. pp. 76-84.

Federal Geographic Data Committee (FGDC) (1995*). Content Standards for Digital Geospatial Metadata Workbook*, (Version 1.0) FGDC/USGS: Reston, Virginia

Ganti, N., & Brayman, W. (1995). *The Transition of Legacy Systems to a Distributed Architecture*, New York: John Wiley & Son, Inc.

Gardner, S. R. (1997). The Quest to Standardize Metadata, *BYTE* November 1997, 22(11), pp.47-48.

Goodchild, M. F. (1996). *Distributed Computing* (The White Paper for UCGIS Research Priorities No. 2). *URL:http://www.ncgia.ucsb.edu/other/ucgis/research_priorities/paper2.html.*

Huse, S. M. (1995). *GRASSLinks: A New Model for Spatial Information Access in Environmental Planning*. Unpublished doctoral dissertation, University of California at Berkeley, Berkeley, California, USA.

Jones, J. (1997). Federal GIS Projects Decentralize. *GIS World,* 10(8), pp. 46-51.

Kuhn, W. (1997). Toward Implemented Geoprocessing Standards: Converging Standardization Tracks for ISO/TC211 and OGC, *ISO/TC211 white paper: N418*.

Lanter, D. P. & Surbey, C. (1994). Metadata Analysis of GIS Data Processing: A cCase Study. In *the Proceedings of the 6th International Symposium on Spatial Data Handling, Edinburgh, UK*. IGU, pp. 314-324.

Montgomery, J. (1997). Distributing Components, *BYTE*, April 1997, 22(4), pp. 93-98.

Open GIS Consortium (1998). *The OpenGIS Abstract Specification Model*. Wayland, Massachusetts: Open GIS Consortium, Inc. *URL: http://www.opengis.org/techno/specs.htm*

Orfali, R., & Harkey, D. (1997). *Client/Server Programming with Java and CORBA*. New York: John Wiley & Sons .

Orfali, R., Harkey, D., & Edwards, J. (1996). *The Essential Distributed Objects Survival Guide*, New York: John Wiley & Sons, Inc.

Ostensen, O. (1995). Mapping the Future of Geomatics. *ISO Bulletin, December 1995. URL: www.statkart.no/isotc211/isobulen.html.*

Plewe, B. (1997). *GIS Online: Information Retrieval, Mapping, and the Internet*. Santa Fe, New Mexico: OnWord Press.

Pountain, D. (1997). The Component Enterprise, *BYTE* May 1997, 22(5), pp. 93-98.

Smith, T. R. (1996). The Meta-Information Environment of Digital Libraries. *D-Lib Magazine.* July/August, *URL: http://www.dlib.org/dlib/july96/new/07smith.html*

Worboys, M. F. (1995). *GIS: A Computing Perspective*. London: Taylor & Francis.

Wu, C. V. (1993). *Object-Based Queries of Spatial Metadata*. Unpublished doctoral dissertation, Geography Department, State University of New York at Buffalo, Buffalo, New York.

Zhang, L., & Lin, H. (1996). A Client/Server Approach to 3D Modeling Support System for Coastal Change Study, *GIS/LIS'96 Proceedings*, November 19-21, 1996, Denver Colorado, ACSM, pp. 1265-1274.