A Framework for Developing Web-Service-Based Intelligent Geospatial Knowledge Systems

Liping Di

Laboratory for Advanced Information Technology and Standards (LAITS), Earth Systems and GeoInformation Sciences
George Mason University, 6301 Ivy Lane, Suite 620, Greenbelt, MD 20770
E-mail:Ldi@gmu.edu

Abstract

This paper discusses an interoperable system framework for developing web-service-based intelligent geospatial knowledge systems. This type of systems facilitates personalized, on-demand geospatial information or knowledge discovery and dissemination. The system will be able to answer "what if" questions by automatically and intelligently chaining individual service modules to form a complex geospatial processing model, matching the input data with the model, and executing the processing model to deliver answers to the users. The system also has the capabilities to evolve and improve by itself. The framework is based on the geo-object and geo-tree concepts, which simulate the geospatial processing models constructed by the experts in geospatial knowledge discovery. The framework also relies on the geospatial interoperability standards, developed primarily by the Open Geospatial Consortium (OGC), recent progress in semantic Web technology and interoperable geospatial Web services. The geo-object and geo-tree concepts provide the mechanism to unify the representation of geospatial data, information, and knowledge, capture the process of geospatial knowledge discovery, manage geospatial knowledge, and enable the dynamic reuse of geospatial knowledge in multiple applications. The framework supports (1) standards-based automated geospatial data and services discovery and access; (2) domain knowledge driven intelligent geo-object decomposition for geo-tree/workflow construction; (3) automated geospatial web service chaining, binding, and execution based on the geo-tree/workflow; and (4) management of workflows and geospatial models.

I. INTRODUCTION

With the advances in sensor and platform technologies, the capability of collecting geospatial data has significantly increased in recent years. In the U.S., both military and civilian agencies have collected a huge amount of geospatial data via remote sensing (King, 1999; King et al., 2003; McDonald and Di, 2003). While those data are potentially valuable to national security and economic growth, they have to be converted to geospatial information and knowledge before they become useful. The challenge is how to discover useful knowledge embedded in the mountains of geospatial data in an effective and timely manner. The traditional approach in which only geospatial experts will analyze data falls short of today's increased demand for geospatial knowledge by a large margin. As a result, much data may never been analyzed even once after collection. Therefore, technologies for automated geospatial knowledge discovery and dissemination are urgently needed for both military and civilian applications.

Three significant features distinguish geospatial knowledge discovery (GKD) from other scientific endeavors: (1) the research is multidisciplinary; (2) the research is data, information, and computation intensive; and (3) the regions covered by the research may vary from micro to global scale. Normally, the processes of geospatial knowledge discovery involve three consecutive steps: (1) *Geoquery,* locating and obtaining data from data repositories; (2) *Geo-assembly,* assembling the data and information from data centers based on the needs of geocomputation; and (3) *Geocomputation,* analyzing and modeling the complex geospatial phenomena by using data and information obtained from geoqueries.

Because of the multidisciplinary nature of GKD, data from data centers are diverse. Often, the temporal and spatial coverage, resolution, origin, format, and map projections of the data are incompatible. As a result, even when the analysis is very simple, considerable time is required in the geoquery and geo-assembly processes to obtain and assemble the data into a format ready for analysis. If the datasets requested are not readily available at the data centers, the data and information system (DIS) at the data centers cannot provide the datasets on demand even if the process to make them is very simple. Therefore, analysts may have to spend a considerable amount of time in ordering and processing the raw data to produce the data they need in the analysis. It has been estimated that more than 70% of scientists' time is spent on steps 1 and 2 of geospatial knowledge discovery(Di and McDonald, 1999). Although DIS is the core supporting component for GKD, current systems, which generally facilitate only data search and ordering, are inadequate.

This paper discusses the framework and key technologies for automatic geospatial knowledge discovery in the web service environment. A geospatial knowledge system built on the proposed framework should be able to (1) fully automate the first and second steps of the geospatial knowledge discovery process in the distributed web service environment so that analysts can focus more on the creative processes of hypothesis generation and knowledge synthesis rather than

spending huge amounts of time on data preparation; (2) fully automate a range of knowledge discovery processes in limited geospatial domains based on automated construction and execution of choreographed web services; and (3) facilitate the construction of complex geocomputation services and models.

II. GEOSPATIAL PROCESSING MODELS

In order to build a geospatial knowledge system that can provide personalized, on-demand information or knowledge to the end users, we have to understand how geospatial experts handle end-users' requests. First, geospatial experts have a set of geospatial processing functions or tools that the experts are very familiar with both semantic and syntax meanings of the inputs and outputs of individual functions. Second, the experts have access to geospatial data archives where the required data reside. They also have knowledge about geospatial data discovery and access as well as about data conversion so that data from multiple sources can be integrated. Third, the experts must have domain knowledge of how the requested geospatial information or knowledge can be generated step-by-step from the assembled data by applying geospatial processing functions. The step-by-step processing from raw data to end-user products is a geospatial processing model. Such a model is normally kept at the conceptual level in the minds of geospatial experts. Only when an end-user requests a product will the expert generate the product by selecting and executing the processing functions that match the function of the model. The geospatial processing model discussed above actually represents knowledge of geospatial information sciences.

III. THE GEOOBJECT AND GEOTREE CONCEPTS

In order to enable a geospatial knowledge system to automatically derive users' requested geospatial information or knowledge from raw data, the system has to be able to simulate what a geospatial expert does in problem solving. Therefore, the system should have access to a set of geospatial processing functions. The system has to be able to access multiple data sources of data intelligently and seamlessly. The system also has to be able to automatically construct the geospatial processing model that can produce user-requested products.

To represent the geospatial processing models in computers and facilitate the generation of user-specified information or knowledge, we developed the concepts of geo-objects and geo-trees. First, we consider a granule of geoinformation (for example, a dataset, a query result, or geocomputation output which describes some aspects of the Earth) to be a *geo-object*, which consists of data, a set of attributes (metadata), and a set of methods that can operate on. A geo-object stored at a data center is an *archived geo-object*. All geoinformation and knowledge products are derived from archived geo-objects.

Thus, from the object point of view, all procedures for geoinformation or knowledge discovery involve creating new geo-objects from existing geo-objects(Di and McDonald, 1999).

If we consider a user's request to be a user-defined geo-object, which may be called a *user geo-object*, the object is either an archived geo-object in a data archive or can be derived by executing a geo-processing algorithm (e.g., unsupervised classification) on a set of input geo-objects. An input geo-object, if not available in an archive, can be further derived by executing a geo-processing function that can generate the new geo-object with a set of input geo-objects and so on. A user geo-object can be decomposed to construct a tree of processing workflow, which is called a *geo-tree* with the root as the user geo-object. Construction of a *geo-tree* is a geospatial modeling process; and the geo-tree itself is a geospatial model that contains knowledge in a specific application domain.

With the geo-tree, we know how to produce the user geo-object although the object does not exist yet in any archive. We call such a geo-object a *virtual geo-object*. In fact, the root of any sub-tree in the geo-tree is a virtual geo-object. Since a geo-tree captures only the workflow, not a specific product, it represents a geo-object type that it can produce, not an instance (an individual dataset). The virtual geo-object can be materialized on demand when all required methods and inputs are available. When a user requests such an object, the user has to specify the geographic location, time, and data format, etc. Those specifications will instantiate the virtual geo-object. By propagating the specifications down to each node of the geo-tree, the whole geo-tree is instantiated. This process is called *instantiation of a geo-tree*. Because the archival geo-objects may not be available for certain user-specified geographic regions and conditions, we can tell if a virtual geo-object can be materialized by doing the instantiation only. After the instantiation, the geo-tree is executable in the system and the virtual geo-object can be produced. The production processing is called *the materialization of the virtual geo-object*, which will produce an instance of the virtual geo-object.

IV. IMPLEMENTATION OF GEOTREES CONCEPTS IN WEB SERVICE ENVIRONMENT

The geoprocessing algorithm for each processing node of a geo-tree is called a *geospatial service module*. The algorithm may handle only a tiny part of the overall geoprocessing or a large aggregated processing. However, the service should be well defined, have clear input and output requirements, and can be executed independently. A geospatial service module implemented in the web environment is called a *geospatial web service module*. Such service modules can be reused to construct different geospatial models for geospatial knowledge discovery. If the necessary service modules are available, any complex geospatial model can be constructed. In a distributed

data and information environment, such as the World Wide Web, there are many independent data and service providers and subsystems. Services needed for a virtual geo-object may be scattered among multiple service providers. Therefore, standards for publishing, finding, binding and executing services are needed. Because the *geo-tree* concept requires the output of one service to be the input to another, standards for service chaining are required.

From the services point of view, a geo-tree is a complex *service chain*. A service chain is defined as a sequence of services in which the occurrence of the first action is necessary for the occurrence of the second action for each adjacent pair of services(ISO, 2005). When services are chained, they are combined in a dependent series to achieve larger tasks. We call a chained geospatial web service *the choreographed geospatial web service*.

The construction of a geo-tree is a service-chaining process. Three types of chaining are defined in ISO 19119(ISO, 2005):
- User-defined (transparent) —a user defines and manages a chain.
- Workflow-managed (translucent)—a user invokes a service that manages and controls the chain and the service is aware of the individual services in the chain.
- Aggregate (opaque)—a user invokes a service that carries out the chain, and the user has no awareness of the individual services in the chain.

A web-service based geospatial knowledge system built on the geo-tree framework will enable **opaque** automated chaining of available data and services in a web service environment to answer users' queries. The system can automatically answer geospatial questions through the following steps:
- A user queries the geospatial knowledge system with a geospatial question in natural language or with controlled vocabularies.
- The system converts the query into a user geo-object with a set of formal descriptions.
- The system decomposes the user geo-object into a geo-tree with help of domain knowledge, which is expressed using ontology and catalogs for service and data types.
- The system converts the geo-tree into an executable workflow by working with the system's data and service catalogs. If the workflow cannot be created, the user query cannot be answered automatically by the knowledge system. In such a case, transparent and translucent chaining methods, which require human intervention, can be used.
- The system executes the workflow to generate and disseminate the user geo-object to answer the user's query.

V. KEY TECHNOLOGIES

In order to automate the geospatial knowledge discovery

described above, key technologies have to be developed to enable the opaque chaining. These technologies include:
- How to automatically decompose the user's query (user geo-object) to construct the geo-tree based on distributed data and service type catalogs and domain ontology?
- How to represent geo-trees in computer-understandable and executable workflows?
- How to manage, share, and reuse geo-trees that represent the geospatial knowledge for a specific domain area?
- How to execute a geo-tree automatically in the distributed web service environment to derive the product that exactly satisfies the user's query?
- How to convert a natural language query into a machine understandable geo-object?

Answering the first question requires an understanding of the relationships and constraints among the data objects and services that support analysis, manipulation, transformation, etc., of geospatial data. In the geo-object concept, if a user object is not at the archive or does not exactly meet the user's requirement, the knowledge system will automatically create the object from the archived geo-objects through a set of dynamically chained services. In the geo-knowledge discovery process, the data and information assembly services involve data reduction and transformation services such as data subsetting, subsampling, reformatting, geometric correction, and radiometric correction. Such services will not change the semantic meaning of data. These services are common to most geospatial analysis, data mining, and feature extraction processes. The rules for chaining those services to derive user products are simple and universally acceptable. Fully automating the first and second steps of the geo-knowledge discovery process can be achieved by chaining those services on demand.

The step for geocomputation will derive information and/or knowledge from a set of archived objects through a set of domain-specific geospatial services. If a user requests a geo-object that is not readily available at the archives, significant domain knowledge is needed to derive the object. For example, a military analyst conducting a battlefield traversability study may ask the intelligent geospatial knowledge system to identify areas in the battlefield that can be traversed by a specific type of military vehicle if a given amount of rain will fall in the next three days. For a geospatial knowledge system to answer such kind of "what if" questions automatically, it has to access all available source data and all required service modules. It must also have accumulated a significant amount of domain knowledge and modeling capabilities (intelligence) in order to automatically construct and execute a geo-tree. With the help of domain-specific ontology and powerful service catalogs, automatic creation of a geo-tree through the decomposition of a geo-object can be achieved via a strict type match between inputs to a downstream service and the outputs from the upstream service. The service catalog has to be able to register geo-object instances, geo-object types, geospatial service instances, and service types, and the associations among them.

When a geo-object or service instance is registered at the catalog, its associated service type has to be declared.

Technology questions 2–4 above are currently being addressed by the Open Geospatial Consortium web services initiatives (Lieberman, 2003; OGC, 2005). In addition, the Semantic Web community has been applying ontology concepts developed in the Artificial Intelligence (AI) community to various aspects of Web Services, Web information search and manipulation. The goal of the emerging Semantic Web services is to provide the mechanisms to organize geospatial information and services so that user queries may be correctly structured for the available application services (the model components and data) in order to "automatically" build workflows for specific problems, determine the correct relationships between available data and services, and generate workflows to provide the "answers" to "what if" questions(DAML, 2004). From this point of view, intelligent geospatial knowledge systems share the same goal as the semantic web services except for that the geospatial systems deal with geospatial problems in particular. The approach to the Semantic Web is to work with the AI community to provide a set of layered extensions to XML in order to develop the ontology language and tools. W3C has recommended Web Ontology Language (OWL) as the standard (W3C, 2004).

Intelligent geospatial knowledge systems have to provide user interfaces that allow users to express their queries easily and clearly and to convert such expressions to user geo-objects accurately. The technologies available for such user interface include controlled vocabulary query interface and natural language query interface. Controlled vocabulary query interface allows users to unambiguously describe what they want, but it is less flexible and extendable, and is not very user friendly. Natural language query interface does not have the problems associated with controlled vocabulary query but it requires the system the capability of understanding natural language. Understanding of natural language by machines is one of the most popular research topics in the AI community. Several technologies and open source tools for geospatial applications have been developed and available now. Two examples of such technologies include **Quark** and **GeoLogica**. Quark (Waldinger et al., 2004) is a natural-language deductive question-answering system assistant to intelligence analysts, which translates English-language queries into a logical form. GeoLogica (Waldinger et al., 2003) is similar to Quark but is applied to the Earth Science domain.

VI. INTEROPERABILITY AND STANDARDS

The framework for intelligent geospatial knowledge systems requires interoperability of both geospatial data and services so that the system can pull out and chain data and services from providers to complete the users' request for geospatial information and knowledge. In order to facilitate interoperability, two standards-based interoperability environments are needed: the common data environment and the common service environment.

The *common data environment* is a set of standard interfaces for finding and accessing data in data archives of varied sizes and sources. This environment allows geospatial services and value-added applications to access diverse data provided by different providers in a standard way without worrying about the internal handling of data. The interface standards for the common data environment are the OGC Web Data Services Specifications, including Web Coverage Services (WCS) (Evans, 2003), Web Feature Services (WFS)(Vretanos, 2002), Web Map Services (WMS) (de la Beaujardière, 2004), and Catalog Services for Web (CS/W) (Nebert, 2004). The specifications allow seamless access to geospatial data in a distributed environment, regardless of the format, projection, resolution, and the archive location (Di, 2004; Di and McDonald, 2004). The OGC technology allows requestors to specify the requirements for the data users want. An OGC-compliant server preprocesses the data on demand into requestor-specified form and then returns the data back to the requestor.

The *common service environment* is a set of standard interfaces for service declaration, description, discovery, binding, chaining, and execution. This environment allows the geospatial knowledge systems to discover and chain standards-compliant services provided by any service providers dynamically to generate user-specific geospatial information/knowledge. Requirements for this set of standards in a geospatial knowledge system are very similar to the requirements in mainstream web services technology. Therefore, the standards used in mainstream web service arena can be adopted for geospatial knowledge systems. Figure 1 shows a set of web service standards from W3C that could be
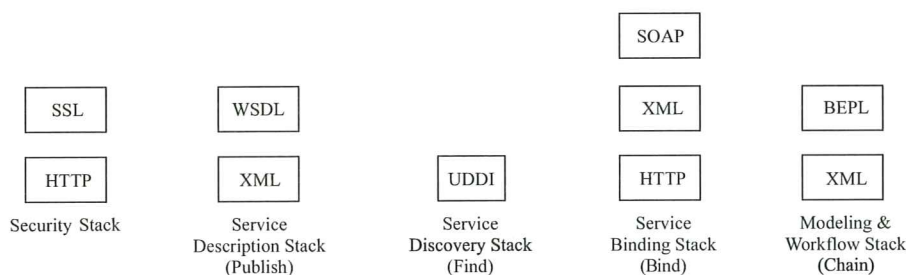

Figure 1. Web service standards for the common service environment

used for building intelligent geospatial knowledge systems. Of course, some extensions to or profiling of such standards are necessary to deal with the specific situations in the geospatial domain.

VII. CONCLUDING REMARKS

The ultimate goal for developing intelligent geospatial knowledge systems is to let everyone, not just a small number of elites, to obtain and use geospatial knowledge from anywhere they want when they need it. This requires the system to be able to automatically derive geospatial information and knowledge requested by the users from vast data archives regardless of where the data are located and how the data are archived. The framework presented in this paper allows building such an intelligent geospatial knowledge system in the web service environment for automatic geospatial knowledge discovery and dissemination. The system will make it much easier and faster for users to investigate a wide range of geospatial problems in both the military and civilian domains. The system will make it possible to explore fully and use effectively the huge amount of geospatial data collected by various organizations. Furthermore, the system will make large-scale-geospatial-knowledge discovery possible from desktop computers that cannot be done currently. The system will also provide a mechanism for capturing experts' geospatial knowledge and making them available to general users.

Currently, such framework is being implemented in GeoBrain, a prototypical geospatial web service system(Di, 2004b), as an experiment to test the generality and utility of the approach. The results of the experiment will include how wide a range of questions a user could ask; what scope of geospatial data and operations is practical to represent in ontology; what the difficulty is of integrating this system into a distributed web service system; and, for questions that result in a query that cannot satisfy the inherent constraints and relationships, whether the system could give clues about the nature of the missing pieces so that users would know what additional data or analysis algorithms would be necessary. The implementation will provide experience and lessons learnt for building an operational web-service-based intelligent geospatial knowledge system in the near future.

ACKNOWLEDGEMENTS

REFERENCES

[1] DAML, 2004, *Semantic Web Services*. http://www.daml.org/services/de la Beaujardière J.(Ed.), 2004. Web Map Service Implementation Specification, Version 1.3. OGC 04-024, Open Geospatial Consortium Inc. http://portal.opengis.org/files/?artifact_id=5316

[2] Di L., 2004, The Open GIS Web Service Specifications for Interoperable Access and Services of NASA EOS Data. In Qu J, et al(Eds.). *Earth Science Satellite Remote Sensing*. Springer-Verlag. (to appear).

[3] Di L., 2004b, GeoBrain-A Web Services based Geospatial Knowledge Building System. Proceedings of NASA Earth Science Technology Conference 2004. June 22-24, 2004. Palo Alto, CA, USA. (8 pages. CD-ROM).

[4] Di L., K. McDonald, 2004, The NASA HDF-EOS Web GIS Software Suite (NWGISS). In Qu J. et al.(Eds.), *Earth Science Satellite Remote Sensing*. Springer-Verlag. (to appear).

[5] Di L., K. McDonald., 1999, Next Generation Data and Information Systems for Earth Sciences Research, in *Proceedings of the First International Symposium on Digital Earth*, Volumn I. Science Press, Beijing, China. pp.8.

[6] Evans J. D. (Ed.), 2003, Web Coverage Service (WCS), Version 1.0.0. OpenGIS© Implementation Specification. Open Geospatial Consortium Inc. http://www.opengis.org/docs/03-065r6.pdf.

[7] ISO, 2005, *ISO 19119:2005 Geographic Information – Services*. http://www.iso.org/iso/en/CatalogueDetailPage.CatalogueDetail?CSNUMBER=39890&ICS1=35&ICS2=240&ICS3=70

[8] King M. D.(Eds.), 1999, *EOS Science Plan*. National Aeronautics and Space Administration, Washington D C. NP-1998-12-069-GSFC.

[9] King M. D., Closs J., Spangler S., Greenstone R.(Eds.), 2003, *EOS Data Products Handbook*. National Aeronautics and Space Administration, Washington D C. NP-2003-4-544-GSFC.

[10] Lieberman J.(Ed.), 2003, OpenGIS Web Services Architecture, OGC 03-025. Open Geospatial Consortium Inc. http://www.opengis.org/techno/discussions/03-025.pdf.

[11] McDonald K., Di L., 2003, Serving NASA EOS Data to the GIS Community Through the OGC-standard Based NWGISS System. In *Proceedings of 2003 Asia GIS Conference*. Asia GIS Society. Oct 16-18, Wuhan, China. pp.13.

[12] Nebert D.(Ed.), 2004. OpenGIS® Catalog Services Specification, Version: 2.0. OGC 04-021r2. Open Geospatial Consortium Inc. http://portal.opengis.org/files/?artifact_id=5929&version=1.

[13] OGC, 2005, *OGC Web Services, Phase 3 (OWS-3)*, http://www.opengeospatial.org/initiatives/?iid=162

[14] Vretanos P. A.(Ed.), 2002, Web Feature Service Implementation Specification, Version 1.0.0 OGC 02-058. Open GIS Consortium Inc. http://www.opengis.org/techno/specs/02-058.pdf.

[15] W3C, 2004, Web Ontology Language (OWL). http://www.w3c.org.

[16] Waldinger R., Jarvis P., Dungan J., 2003, *Using Deduction to Choreograph Multiple Data Sources*. In Semantic Web Technologies for Searching and Retrieving, Sanibel Island, Florida, Oct 2003. http://www.ai.sri.com/pubs/files/993.pdf

[17] Waldinger R., Appelt D. E., Fry J., Israel D. J., Jarvis P., Martin D., Riehemann S., Stickel M. E., Tyson M., Hobbs J., Dungan J. L., 2004, *Deductive Question Answering from Multiple Resources*. In New Directions in Question Answering, AAAI, 2004 (to appear). http://www.ai.sri.com/pubs/files/986.pdf