

Chapter I

Geospatial Web Services

Peisheng Zhao, George Mason University, USA

Genong Yu, George Mason University, USA

Liping Di, George Mason University

Abstract

As Web service technologies mature in recent years, a growing number of geospatial Web services designed to interoperate spatial information over the network have emerged. Geospatial Web services are changing the way in which spatial information systems and applications are designed, developed, and deployed. This chapter introduces all aspects of geospatial Web services from service-oriented architecture to service implementation. It covers the life cycle of geospatial Web services in terms of geospatial interoperable standards, including publish, discovery, invocation, and orchestration. To make geospatial Web services more intelligent, semantic issues about geospatial data and services are discussed here. Furthermore, the applications of standard-compliant geospatial Web services are also reviewed.

Introduction

Web service technology promises standard-based information interoperability. There are many different definitions for Web services (Andersson, Greenspun, & Grumet, 2003; Booth et al., 2004; Hirsch & Just, 2003; Mateos, Zunino, & Campo, 2005; Skonnard, 2002; Vaughan-Nichols, 2002). In essence, a Web service is a modular, self-describing, and self-contained software application which is discoverable and accessible through standard interfaces over the network (Tsalgatidou & Pilioura, 2002).

The core technology associated with Web service is the standardization of data/message exchange between applications or systems during every stage of their life cycle, including transporting, invoking, and discovering (Akinci, 2004; Di, 2004a; Hecht, 2002). XML (eXtensible Markup Language) is used as the primary language to encode data/message in Web services since it hides the details of underlying transport protocols and provides a platform-independent structured information format. Structured information can be exchanged using standard protocols, such as SOAP (simple object access protocol), or XML-RPC (Gudgin, Hadley, Mendelsohn, Moreau, & Nielsen, 2003; Winer, 1999). The public interface (functionality and input/output parameters) of a Web service is described following a machine-processable format, such as Web Service Description Language (WSDL) (Booth & Liu, 2005; Chinnici, Haas, Lewis, Moreau, Orchard, & Weerawarana, 2005; Chinnici, Moreau, Ryman, & Weerawarana, 2005; Christensen, Curbera, Meredith, & Weerawarana, 2001; Vedamuthu, 2005). A standard registry or catalog is often used to publish and discover these Web services, such as UDDI (Universal Description, Discovery and Integration) (Booth et al., 2004; Clement, Hately, Riegen, & Rogers, 2004). These characters distinguish a Web service from traditional proprietary distributed systems, such as distributed common object model (DCOM) by Microsoft, java remote method invocation API (RMI) by Sun, and common object request broker architecture (CORBA) by Object Management Group (OMG).

The major benefit of Web services is the interoperability enabled by those standards; in other words, Web services are capable of collaborating process control and sharing data and information across applications over different platforms (Di, 2005; Di, Zhao, Yang, Yu, & Yue, 2005; Kralidis, 2005). A Web service hides all the details of implementation under a well-defined interface, and thus other applications or services can invoke such a Web service through the standard interface. Such type of interoperation is not just limited within one organization, but also can be conducted across organizations. From the technical point of view, the advantages of using Web services can be summarized as: (1) enabling the sharing of computational resources (hardware, software, and data/information holdings) across the organization boundary; (2) easy to maintain and wrap legacy system since the modularity of Web service allows the partial updating and change to existing systems; (3) independent from

platforms and operating systems since Web services interact with clients or other Web services through standard messages; and (4) independent from programming languages and implementations as long as the claimed interfaces are obeyed (Akinci, 2004; Di, 2004a; Kralidis, 2005; Zhao et al., 2004). Other advantages are: real-time data access, on-demand/customized information delivery, value-added composition, easy integration, and extended collaboration (Kralidis, 2005). These benefits are achieved at the slightest cost of performance compared to proprietary distributed computing approaches. Overall, Web services are more suitable for handling huge amount, distributed, and diverse geospatial data/information as compared to the conventional approach of geospatial data dissemination.

Traditionally, geospatial information is delivered to end users through media such as magnetic tapes (e.g. 8mm storage tape, computer compatible tape), optical disks (e.g., CD-ROM, DVD), and electronic transfer (e.g., file transfer protocol, hypertext transfer protocol, simple mail transfer protocol). Often a catalog is provided for searching the data in granules or some pre-determined units, and a matched list of geographic information is given for downloading or mailing. The problems with such approaches to deliver geospatial information are numerous (Di & McDonald, 2005; Di, Yang, Deng, Deng, & McDonald, 2002; Kralidis, 2005). Firstly, the geospatial information is delivered in file-based format and is not customizable. Geospatial information may be packaged in a provider-defined size, which potentially lead to overload of network (with larger dataset than user desires) or to under-sized data (with many small datasets to meet user's requirement that leave a lot of work on converting and merging these datasets). Geospatial information may not be updated timely with the most current status as the data change at the provider's site. Delay of obtaining data may not be desirable when the study is very time-sensitive. Secondly, the process to obtain geographic information is labor-intensive. With the traditional approach, a user may spend a lot of time in many steps: (1) identifying data source and its catalog, (2) interactively searching the dataset with multiple attempts to narrow down the desired dataset, (3) downloading the data or waiting for the data media to be delivered, (4) converting and importing the data into the system the user uses, and (5) processing the data and presenting the results. All of these processes cannot be automated or prescribed with batch-processing plans. Thirdly, geospatial information is not interoperable. The information may be stored in vendor's proprietary formats, and a specific processing has to be taken care of before it can be used in user's system. The messages communicated between different vendors' systems always cannot be directly processed by other machines or systems. Finally, it is very difficult to share the value-added geospatial information. A user may be in need of a processing function that the provided software package does not possess and then pushed to purchase a suite of software to get the specific information. All these issues with the traditional geospatial information dissemination and processing are what Web services promise to solve, by leveraging Web service capabilities: (1) modularity for the sharing of specific function, (2) standard

interfaces for the interoperability of applications, (3) on-demand capability for the customized, real-time value-added geospatial information delivery, and (4) self-contained functionality for the loosely-coupled operation on dataset. The marriage of Web service with geospatial information leads to geospatial Web services.

Generally, a geospatial Web service can be viewed as a modular Web application that provides services on geospatial data, information, or knowledge (Di, Zhao, Yang, Yu, & Yue, 2005). It refers to the use of Web services technologies to manage, analyze, and distribute spatial information. Furthermore, a geospatial Web service can be sorted and searched through its geospatial characteristics, such as location, area, neighborhood, and other spatial features. As a Web service, a geospatial Web service involves three actors: user entity (consumer), provider entity (provider), and register entity (broker) (Booth et al., 2004). Basic operations during the life cycle of a geospatial Web service include publication, discovery, binding, invoking, and execution. The interfaces of some geospatial Web services have been standardized. The most important players for such standardization efforts are ISO/TC211 and Open Geospatial Consortium (OGC). Because of the influential specifications from OGC, geospatial Web services have an add-on unique geospatial flavor on the message-communicating other than those popular W3C Web services standards such as WSDL, SOAP and UDDI. Geospatial Web services focus on standard interface directly through HTTP protocol, which is now recognized as more related to representation state transfer (REST) (Booth et al., 2004; Fielding, 2000). Further details about the OGC standards will be discussed in Section 2. Geospatial Web services are changing the way in which spatial information systems and applications are designed, developed, and deployed. The added characters give a flexible extension of functionality required to enable the interoperable operation of geospatial data and their processing.

In this chapter, following the lead of standards and the ultimate goal of interoperability, we will cover all aspects of geospatial Web services from conceptual level to technology level. In section 2, the interoperability standards, especially those from ISO and OGC, will be introduced. In section 3, examples will be given to illustrate the life cycle of Geospatial Web services and guide readers how to design, create, deploy, and execute Geospatial Web services. Section 4 discusses the discovery of geospatial Web service with three approaches — registry, index, and peer-to-peer. Registry, or service catalog, will be the main focus. Section 5 presents the composition of Web services or assembling individual services into a more useful process flow. Section 6 introduces some Geospatial Web service solutions that are widespread in use. Section 7 touches the most advanced aspects of Geospatial Web services, semantics. Finally, a short summary and future direction are given.

Standards for Geospatial Web Services

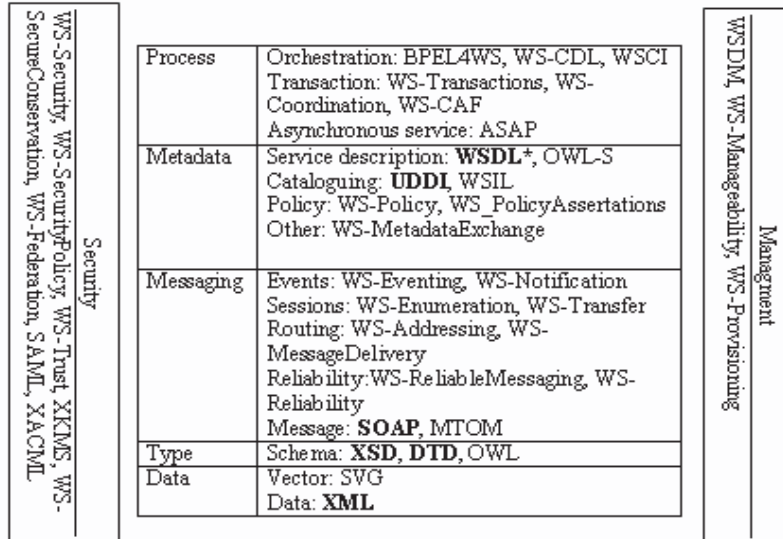
Geospatial interoperability is achieved through the use of standards by different standard-setting organizations, including government organizations (e.g., FGDC), international standard bodies (e.g., ISO), and industry associations (e.g., OGC) (Groot & McLaughlin, 2000; Kralidis, 2005). Major bodies for standards of geospatial Web services are ISO/TC211, OGC, FGDC/NSDI (EOP, 2003; Evans & Bambacus, 2005; ISO/TC211, 2005; OGC, 2005a). The general Web service standard stacks can be partitioned into three parts: data (message encoding), interface (transport protocol), and metadata (ontology), which cover all the aspects of interoperations (Alameh, 2001; Nebert, 2005). These are similar in geospatial Web services. At the data level, the standards specify the message encoding and data formatting that are used for communicating between Web services and applications. At the interface level, the standards define common interfaces for both applications/Web services and human users. At the metadata level, a set of consensus data types and descriptions are associated with each Web service or data.

One should notice that OGC geospatial Web services are different from Web services in the e-business world. OGC geospatial Web services have been developed in parallel with the evolution of W3C and OASIS Web services. Standards from W3C and OASIS, such as WSDL, SOAP, and UDDI, are the standards for Web services in the e-business world. The OGC geospatial Web services do not comply with these Web service standards. Recently, OGC started to explore the possibilities in bridging the gaps and implementing OGC Web services using W3C/OASIS Web services (Duschene & Sonnet, 2005a, 2005b; Lieberman, Reich, & Vretanos, 2003; Sonnet, 2004; Sonnet & Savage, 2003). Another move in standardization of geospatial Web services is the exploration of the semantic Web. OGC initiated the geospatial semantic Web interoperability experiment in April, 2005 (Bacharach, 2005).

Web Service Standard Stacks

Web services are an emerging technology for which standards are currently in great need. Many organizations are participating in the standardization of data, interfaces, and metadata, such as World Wide Web Consortium (W3C), Organization for the Advancement of Structured Information Standards (OASIS), and Internet Engineering Task Force (IETF) (Alameh, 2001; IETF, 2005; Kralidis, 2005; Nebert, 2005; OASIS, 2005; W3C, 2005b; Wilkes, 2005). Figure 1 shows the protocols for Web services that are either widely-adopted or in development. The bold-faced protocols formed the *de facto* core standards for Web services. XML is the underlying language used to encode messages, metadata, schema, and interfaces. XSD (XML schema) replaces document type definition (DTD) to define information structure (W3C, 2005a). SOAP is a commonly-used message encoding protocol (Gudgin,

Figure 1. Web service protocol stack (Source: Tsalgatidou & Pilioura, 2002; Wilkes, 2005)



*Bold face protocols or standards are most popularly adopted in industry.

Hadley, Mendelsohn, Moreau, & Nielsen, 2003). WSDL is used for describing each Web service, including input/output, operations, and underlying transport protocols (Christensen, Curbera, Meredith, & Weerawarana, 2001). UDDI is a protocol for building a directory of Web services (Clement, Hatley, Riegen, & Rogers, 2004). Other protocols are mostly in the stage of early adoption, experimentation, or specification proposal (Wilkes, 2005).

Geospatial Web Service Standard Stacks

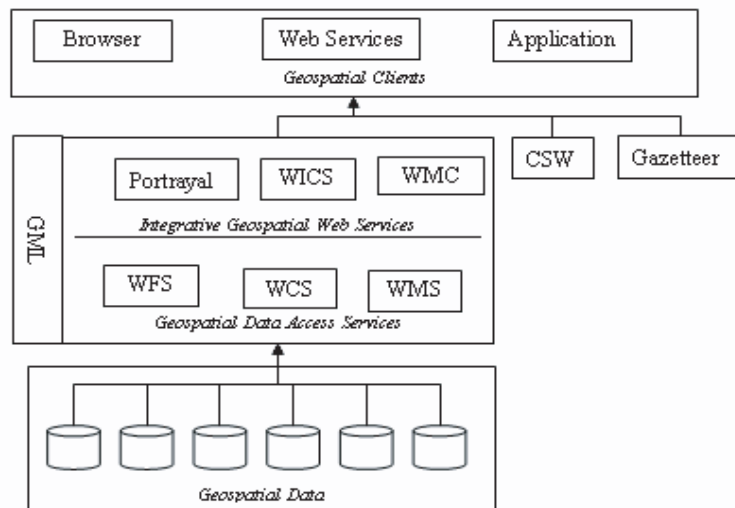
OGC and ISO/TC211 are the major players in standardizing geospatial Web services. OGC specifications focus on developing implementation standards while ISO/TC 211 is concentrating on developing theoretical/abstract standards. ISO/TC 211 standards specify methods, tools, and services for acquiring, processing, analyzing, accessing, presenting, and transferring spatial information between different users, systems, and locations. OGC specifications support the full integration of “geo-enabled” Web services into mainstream computing to make complex spatial information and services accessible and useful with all kinds of applications. Through the cooperation between ISO TC 211 and OGC, most of approved OGC implementation standards are either already ISO standards or in the process to become the ISO standards (Di, 2003).

Figure 2. Geospatial Web service standard stack (Source: ISO/TC211, 2005; Nebert, 2005; OGC, 2005a)

Process	Query: ISO 19125-1 Integrative: WFS
Metadata	Service description: WSDL, ISO 19119, ISO19109 Data description: ISO19115:2003, ISO TS19139 Cataloguing: CAT, ISO19110:2005
Messaging	Application interfaces: WCS, WFS, WMS, WICS, WCTS, CQL User interface: WMC Message: HTTP PUT/GET
Type	Schema: XSD, DTD, OWL
Data	Vector: GML Data file: SDTS, VPF, DIGEST, HDFEOS Data: XML

Since 1999, OGC has finished phase 1, 2, and 3 of Open Web Services (OWS) initiatives in 2002, 2004, and 2005 that addresses geospatial interoperability requirements and standards to enhance the discovery, retrieval, and use of geospatial information and geoprocessing services. Under these OGC initiatives, a series of geospatial Web services specifications have been published, such as Web coverage service (WCS) (Evans, 2003), Web feature service (WFS) (Vretanos, 2005), Web map service (WMS) (de La Beaujardiere, 2004), Web image classification

Figure 3. Overall architecture of OGC geospatial Web services (Nebert, 2005)



service (WICS) (Yang & Whiteside, 2005), Web coordinate transformation service (WCTS) (Whiteside, Müller, Fellah, & Warmerdam, 2005). Those specifications are the foundation for geospatial interoperability (Di, 2005b). Figure 2 shows most standards and specifications for geospatial Web services.

Similar to the standards for general Web services, these standards and specifications can also be grouped into three groups: data, interface, and metadata. They play different roles in the overall architecture of OGC geospatial Web service. The data standards specify the storage file format of geospatial data. The interface specifications enable the interactions between geospatial Web services. The metadata are used to describe geospatial data and geospatial Web services. Figure 3 shows the overall architecture of OGC geospatial Web services.

Data Standards

The standardization for geospatial data formats sets a good basis for geospatial Web services to communicate with each other in an understandable format. These efforts have resulted in many specifications and standards for data storage and exchange, such as digital line graph (DLG), digital raster graph (DRG), spatial data transfer standard (SDTS), and hierarchical data format for Earth observation system (HDFEOS) (Alameh, 2001; GSFC, 2005). In compliance with ISO 19118 for the transport and storage of geospatial information, the OGC Geography Markup Language (GML) provides an open, portable, and vendor-neutral framework to define geographic features and datasets (Cox, Daisey, Lake, Portele, & Whiteside, 2004). GML uses XML text to encode geometry and properties of geographic feature, spatial reference system, and feature collections. The GML also absorbs great graphic quality of scalable vector graphics (SVG). These data formats are commonly used for the delivery of geospatial data with geospatial Web services.

Interface Standards

OGC has been dedicated to the standardization of interfaces of geospatial Web services to enable the interoperability. All OGC Web services are based on a common model and share some unified characteristics: (1) some common operation request and response contents (e.g., *getCapabilities* operation), (2) some unified parameters for operation requests and responses, and (3) exclusive adoption of XML and key-value-pair (KVP) in encoding (Whiteside, 2005).

Under such a common specification, a series of specifications were developed and adopted for delivering and analyzing geospatial data over the Web. CAT defines a catalog service for discovery, browsing, and querying of distributed and heterogeneous catalog servers (Nebert & Whiteside, 2004). Coordinate transformation

service (CT) specifies interfaces for positioning, coordinate systems, and their transformation (Daly, 2001). WCS defines common interfaces for grid analysis and processing (Burry, 2001; Di, 2005b). WFS uses GML to encode geographic features and defines common interfaces to handle geographic vector information (Vretanos, 2005). WMS enables the display of registered and superimposed map-like views of geographic information in some image formats (e.g., JPEG) (de La Beaujardiere, 2004). Web map context (WMC) supports the creation and manipulation of a context map that consists of several WMS maps (Sonnet, 2005). Location service (OpenLS) describes an open platform for location-based application servers and outlines related activities (Mabrouk, 2005).

Metadata Standards

To enable human interpretation and machine processing of geospatial data and geospatial Web services, another important standardization endeavor is the definition of a common metadata (description about data or services). For geographic information, ISO19115 provides abstract guidelines for geographic information metadata and ISO19119 for geospatial services metadata (ISO/TC211, 2002b, 2003a). ISO 19139 defines the XML schema for ISO19115 (ISO/TC211, 2003b).

ISO 19115 defines metadata for geospatial data product in fourteen parts: (1) metadata entity set (*MD_Metadata*) is the mandatory part that includes identification, constraints, data quality, maintenance, spatial representation, reference system, content, portrayal catalog, distribution, metadata extension, and application schema; (2) identification (*MD_Identification*) uniquely identifies the data by defining format, graphic overview, specific uses, constraints, keywords, maintenance and aggregate information; (3) constraint (*MD_Constraints*) defines the restrictions placed on the data; (4) data quality (*DQ_DataQuality*) contains quality of the dataset and information about the sources and production processes; (5) maintenance (*MD_MaintenanceInformation*) describes the scope and frequency of updating; (6) spatial representation (*MD_SpatialRepresentation*) points out the mechanism to represent spatial information; (7) reference system (*MD_ReferenceSystem*) describes spatial and temporal reference system; (8) content (*MD_ContentInformation*) identifies the feature catalog; (9) portrayal catalog (*MD_PortrayalCatalogReference*) gives the type for displaying data; (10) distribution (*MD_Distribution*) describes the distributor of the data; (11) metadata extension (*MD_MetadataExtensionInformation*) is for user-specified extensions; (12) application schema (*MD_ApplicationSchemaInformation*) is for the schema used to build a dataset; (13) extent (*EX_Extent*) describes the spatial and temporal extent; and (14) citation and responsible party (*CI_Citation and CI_ResponsibleParty*) provides citation information.

ISO 19119 presents taxonomy of geospatial services and gives a list of example geospatial services in each service category. Major geographic services are: (1) geo-

graphic human interaction, for example, catalog viewer, geographic feature editor, (2) geographic model/information management, for example, feature access, map access, catalog service, (3) geographic workflow/task management, for example, chain definition service, workflow enactment service, (4) geographic processing (spatial, thematic, temporal, and metadata), for example, coordinate conversion service, thematic classification service, temporal reference system transformation service, statistical calculation service, (5) geographic communication, for example, encoding service, transfer service, and (6) geographic system management.

Geospatial Web Service Implementation

There are many tools available for the implementation and deployment of Web services. Theoretically, any programming languages can be used in developing Web services.

Web Service Implementation

Web service, technically and typically, uses WSDL to describe interfaces, encodes communicate messages using SOAP, and publishes itself in a UDDI registry. A typical process to create a Web service from scratch: (1) starts with the WSDL design using Unified Modeling Language (UML) or other visual design tools, (2) decides the message styles (either RPC encoded/literal or document encoded/literal), (3) develops the program, and (4) ends with the publication and testing of the Web services in a SOAP container (Peltz, 2003).

Another approach is to adapt the existing programs and expose some of their functions as Web services (Peltz, 2003). The task to deploy such Web services mainly involves the creation of adapters that make the legacy system compatible with the Web services. For example, a developer may create an HTTP adapter to convert the input and output between SOAP and HTTP messages, or may develop a Java adapter to call the C/C++ component through Java native interface (JNI) if the published Web service container is a Java server page container (e.g., Axis for Java). Other design patterns may be considered to efficiently re-use the Web services and balance the loading of network traffic, such as MVC (model-view-controller), and asynchronous messaging (Peltz, 2003).

Geospatial Web Service Implementation

In the implementation of geospatial Web services, similar approaches can be applied. In a narrower sense, geospatial Web services are limited to those following geospatial standards, such as OGC WFS, WCS, and WICS. In a broader sense, geospatial Web services embrace Web services that handle geospatial data and provide services to processing geospatial data. Examples are the Web services adapted from geographic resource analysis support system (GRASS) at LAITS (GRASS, 2005; LAITS, 2005).

Implement Standard Geospatial Web Services

A typical implementation practice for OGC geospatial Web services may start with the analysis of the geospatial data to be served and determine the proper interfaces to be implemented, especially those optional operations and parameters. Secondly, all the internal functions should be coded and tested modularly. Thirdly, these required interfaces (operations) and their parameters should be exposed by abiding the relevant specifications. Finally, test and debugging should be carried out. A series of quality assurance and performance evaluation should be conducted (Di, 2004b; Kolodziej, 2004; Zhao et al., 2004).

It is essential to implement the specific interfaces for developing OGC Web services. Some of the interfaces are mandatory. Others are optional. For example, the operations of *GetCapabilities*, and *GetMap* in WMS are mandatory, and its *GetFeatureInfo* operation is optional (de La Beaujardiere, 2004; Kolodziej, 2004). Furthermore, the parameters for each operation are partly required and partly optional. For example, the *GetMap* operation of a WMS require parameters of “VERSION=1.3.0”, “REQUEST=GetMap”, “LAYER=?”, “STYLES=?”, “CRS=?”, “BBOX=?,?,?,?””, “WIDTH=?”, “HEIGHT=?”, and “FORMAT=?”. Other KVP (key value pairs) are optional, such as “TRANSPARENT=TRUE|FALSE”, “BGCOLOR=?”, and “EXCEPTIONS=?”.

Currently, OGC Web service supports only HTTP GET and/or HTTP PUT binding at the transport level by default. This is one aspect that differs from the W3C/OASIS Web services. OGC is trying to bridge this gap by making some experiments on SOAP binding. OGC Web services can be published through common gateway interface (CGI) (a standard to interact with external applications through an information servers, such as HTTP or Web servers), servlets (a Java-enabled server), active server pages (ASP) (a Microsoft-based approach), Java server pages (JSP) (a Java-enabled approach), and ASP.NET (a Web page service upon a Microsoft .NET framework).

When there are existing routines or Web services, the adaptation of these geoprocessing services or functional modules to comply with OGC specifications may be

required. There are three possible cases: (1) to adapt a Web service, an OGC-compliant translator (using XSL/XSLT) may be sufficient since the encoding is already XML-based, (2) to adapt a server program, extra operations may be required to meet the specifications, and (3) to adapt a client, corresponding operations may be implemented to access the OGC-compliant Web services (Kolodziej, 2004; Zhao et al., 2004; Zhao, Nusser, & Miller, 2002).

Implement Customized Geospatial Web Services

In most cases, we may still be required to deploy some specific geospatial Web services that complete certain geospatial operations. If so, the developers may need some thinking on determining which operations are to be published, how to manage the intermediary and temporary file spaces, how to achieve the best performance in a distributed environment, and how to secure the system at large. This is especially true for these legacy geographic information systems. Their rich functions may be demanded in a service-oriented environment. Here we use one case that we adapted the GRASS functions and publish most of the operations as geospatial Web services at LAITS. These services are required for the effective demonstration on the intelligent chaining of Web services using a large repository of geospatial Web services.

GRASS has a huge number of operations. It is a comprehensive system of more than 350 programs for raster/vector geographic data manipulation, image processing, and map production. The functions range over spatial analysis, map generation (on paper and monitor), data visualization (2-dimensions, 2.5-dimensions, and 3-dimensions), data modeling, database management, and data manipulation (GRASS, 2005). In order to make GRASS functions as Web services, the first issue is to transfer these functions from a desktop environment to a networked environment. As a stand-alone system, the GRASS starts with a series of global variable initialization and keeps several states over the execution of programs. These need to be adapted to handle multiple routines running simultaneously. This can be solved by wrapping the initialization with each function to be explored, or every function would start with a set of own initialization. The second issue is what functions are proper to be deployed as Web services, and what is the basic level of operations to be exposed, command level or combined script. The command level is an individual program in GRASS to complete a small operation, such as Fast Fourier Transform, statistical summation, and optimum-index-factor table calculation. The combined script may complete a series of operations by editing and executing a script file of multiple steps of commands. The solutions are: (1) The Web service should be mainly based on the command level as this is the lowest level of function units. At this level, the creation of Web services from each command is straightforward, so as to ease the maintenance when new functions are added into GRASS. Combined operation can be easily achieved by combining several Web services through workflow manager, such as BPEL engines; and (2) Some commands are not exposed as Web services,

especially those purely dealing with disk storage and monitor display. Finally, the use of customized geospatial Web services by an OGC-compliant Web service or an OGC-compliant client is a big issue. The input and output geospatial data are not exposed through an OGC-compliant data services. They are merely a universal resource identification (URI). This can be resolved by providing a series of wrap services to enable the feeding of geospatial data from an OGC-compliant data-providing service and the delivering of geospatial data to an OGC-compliant data server. By doing so, the intermediate processes are invisible to OGC-compliant geospatial services, but they can be easily handled by following Web service standards.

Geospatial Web Service Discovery

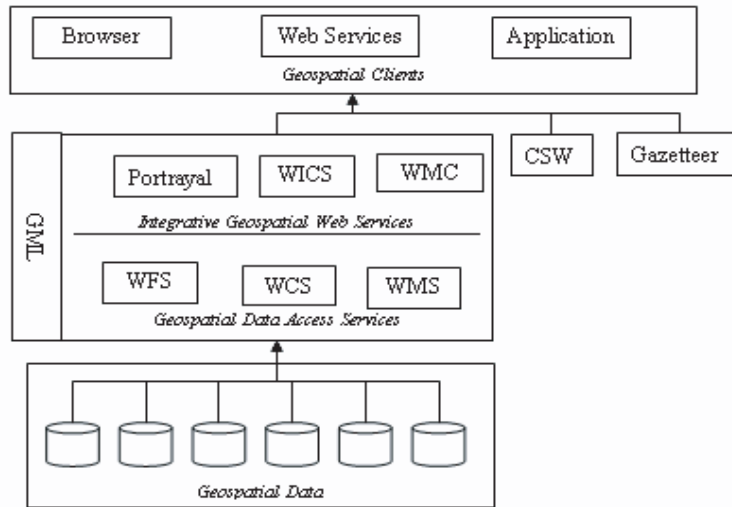
One of the first steps in consuming geospatial Web services is to find the desired geospatial Web services. Web service discovery is to locate a machine-processable description of a Web service that meets desired functional criteria. There are mainly three types of approach: Registry, Index, and Peer-to-Peer (Booth et al., 2004). Similarly, geospatial Web services can be searched through these three approaches.

Search by Registry

A registry is a catalog or a directory that is authoritative on the description of each registered Web service or metadata. The “catalog” plays a “directory” role: Providers advertise the availability of their resources through metadata, and users can then query the metadata to discover interesting resources and determine how to access those resources at run time (Nebert & Whiteside, 2004; Zhao et al., 2004). The success of a registry search highly depends upon what metadata is recorded for each geospatial Web service and how they are managed. If limited metadata information is registered, a search may give an explosive number of returns because of the broad matching. Heavy requirements on the metadata information would be a burden for the provider to submit all the details. There are many efforts in building the core metadata that are necessary for describing a geospatial Web services. Leading standards are ISO19115 for data and ISO19119 for service (ISO/TC211, 2002a, 2002b, 2003a).

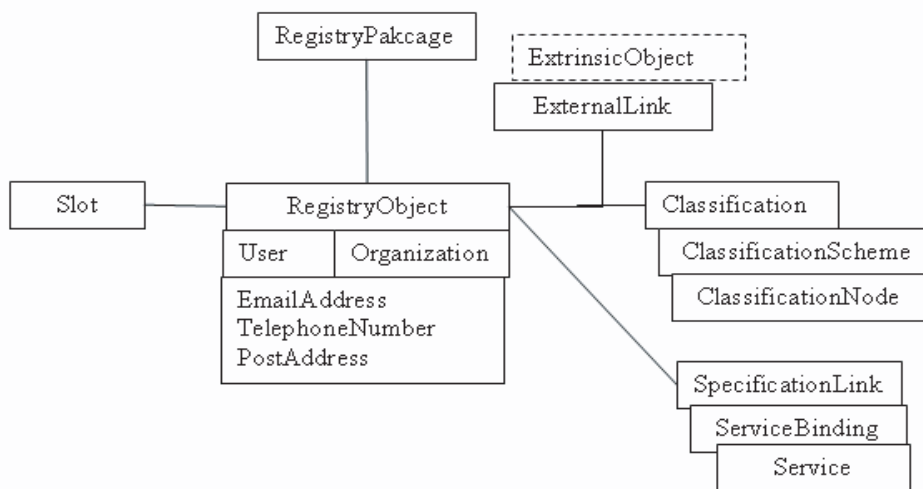
Internally, a registry can be organized into a database, either a relational database system or an XML-based database. Once all the metadata information is entered into the registry, users or applications can query the registry with certain constraints clauses or criteria and retrieve matched service descriptions. Figure 4 illustrates major actors who interact with the registry and main operations which the actors perform on the registry.

Figure 4. Geospatial Web Registry Server



In a distributed infrastructure, the registry is presented as a Web service which targets at interoperation with client applications or other Web services (Alameh, 2001). In such a case, certain interoperable interfaces should be followed for the design of the registry service. UDDI is a standard registry to manage Web services with

Figure 5. A simplified illustration of ebXML information model (Source: OASIS/ ebXML, 2003)



SOAP (Simple Object Access Protocol) as messaging protocol and WSDL (Web service Description Language) as Web service description language. And the OGC Catalog Service for Web (CSW) is a standard specification specifically designed for cataloging geospatial Web services (Martell, 2004; Voges & Zenkler, 2005; Wei et al., 2005; Zhao et al., 2004).

The information model of CSW is based on ebXML information model (ebRIM) (see Figure 5) (OASIS/ebXML, 2003). Table 1 lists classes used in this information model. The ebRIM is designed to be extensible through several mechanisms, either creating an ExtrinsicObject or adding a Slot. Therefore, CSW can easily be extended to describe different objects, such as organization, data, and services. For geospatial data and geospatial Web services, it is logical to extend the internal description capability of an CSW registry server with metadata described by ISO 19115 and ISO 19119 (Wei et al., 2005).

Table 1. Classes in an ebRIM

Class	Description
RegistryObject	The base object for the registry
RepositoryItem	Any object in a repository for storage and safekeeping
Slot	Arbitrary attributes to the RegistryObject
Association	Define the many-to-many relationship between objects in the information model
ExternalIdentifier	Additional identifier information to a RegistryObject
ExternalLink	An URI to external RegistryObject
ClassificationScheme	A scheme to classify RegistryObject instances
ClassificationNode	A child node to a ClassificationScheme
Classification	A taxonomy value in a classification scheme
RegistryPackage	A group of Registry
AuditableEvent	An audit trail for RegistryObject instances
User	A registered user
PostalAddress	A postal address
E-mailAddress	An e-mail address
Organization	For example, submitting organization
Service	Service information
ServiceBinding	Technical information on a specific way to access a specific interface of a Service
SpecificationLink	A link referring to the technical specification that describes how to use the service

Table 2. CSW interfaces and their operations (Source: Nebert & Whiteside,

Interface	Operation	Description
OGC_Service	getCapabilities()	To retrieve metadata describing what the Catalog Service can provide
Discovery	query()	To execute a query that searches the cataloged metadata and returns results satisfying the search criteria
	present()	To retrieve selected metadata for specific records
	describeRecordType()	To retrieve type definitions used for that dataset
	getDomain()	To retrieve the domain (allowed values) of a metadata property or request parameter
Manager	transaction()	To request a specific set of “insert”, “update”, and “delete” action
	harvestResource()	To retrieve a resource from a specified location and to optionally create one or more entries for that resource
Session	Not discussed. Possible operations are: cancel(), close(), initialize(), and status().	
BrokedAccess	No discussed. Possible operation is order().	

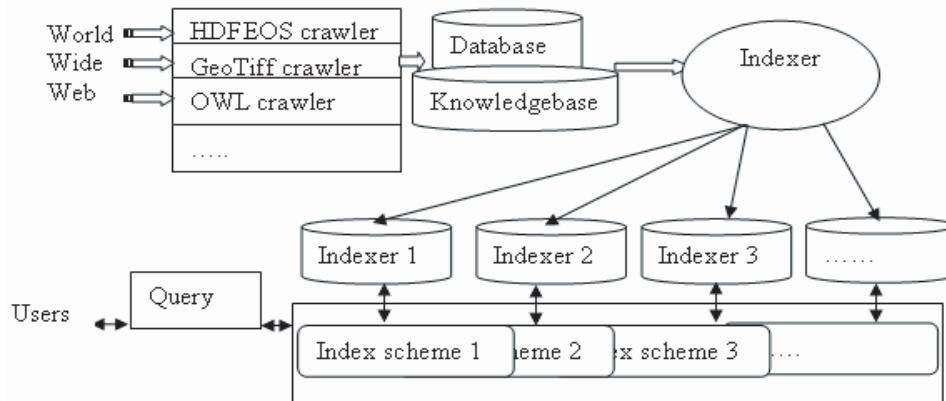
The CSW interfaces are well-defined that unify the operations for the registration of new geospatial data/Web services and the query and retrieval of item description. Table 2 lists the interfaces and operations defined by CSW. This specification is still in evolving.

Search by Index

A search for geospatial Web services can be done more actively by using an index approach. The index approach harvests the Web services by crawling through all Web sites and servers and collects their metadata information into an indexer. This approach is a bottom-up search in which several dedicated crawlers keep on scrubbing through all Web sites to find geographical data and geospatial Web services. Figure 6 shows a proposed architecture for such an index search engine. The OGC Registry can be used to manage these collected metadata, and the query interfaces can follow the OGC Registry standard as well. The difference is that such an approach relies upon a passive publication of geospatial Web service and geospatial data; the crawlers of the search engine scrub through the Web to find this data instead of providers’ submission.

The most challenge in this approach is how to determine if a link or Web service is geospatial or non-geospatial. The accuracy of identifying geospatial data and geospatial Web service depends upon the metadata associated with each data or Web service. For standard geospatial data (e.g., HDF-EOS file), an extraction of internal

Figure 6. Crawler-based geospatial Web service discovery



geospatial information will gather its geographic extent, time, and coordinate systems. For standard-compliant geospatial Web services (such as OGC WFS, WCS), further contents can be extracted by calling its standard interface, for example, `getCapabilities` for OGC-compliant geospatial Web services. For semantically-described geospatial Web service or data (such as associated taxonomy information in Ontology Language for Web [OWL]), a reasoner can be embedded to extract geospatial knowledge, such as class hierarchy of the data.

Search by Peer-to-Peer

Peer-to-peer (P2P) computing treats each computer as equally as another computer in the network. It is a different view from client/server architecture. Grid computing is considered as a P2P. The P2P search architecture does not need a centralized registry or a collected indexer. Each node in the P2P responds to the queries it receives (Booth et al., 2004). Once a user submits a query, the query is passed on to each individual node in the P2P network and returns the matched results through relaying. Such a search system may suffer performance inefficiencies, for example, lag time in propagation of queries and relaying of matches, incomplete routing through the network.

Geospatial Web Services Orchestration

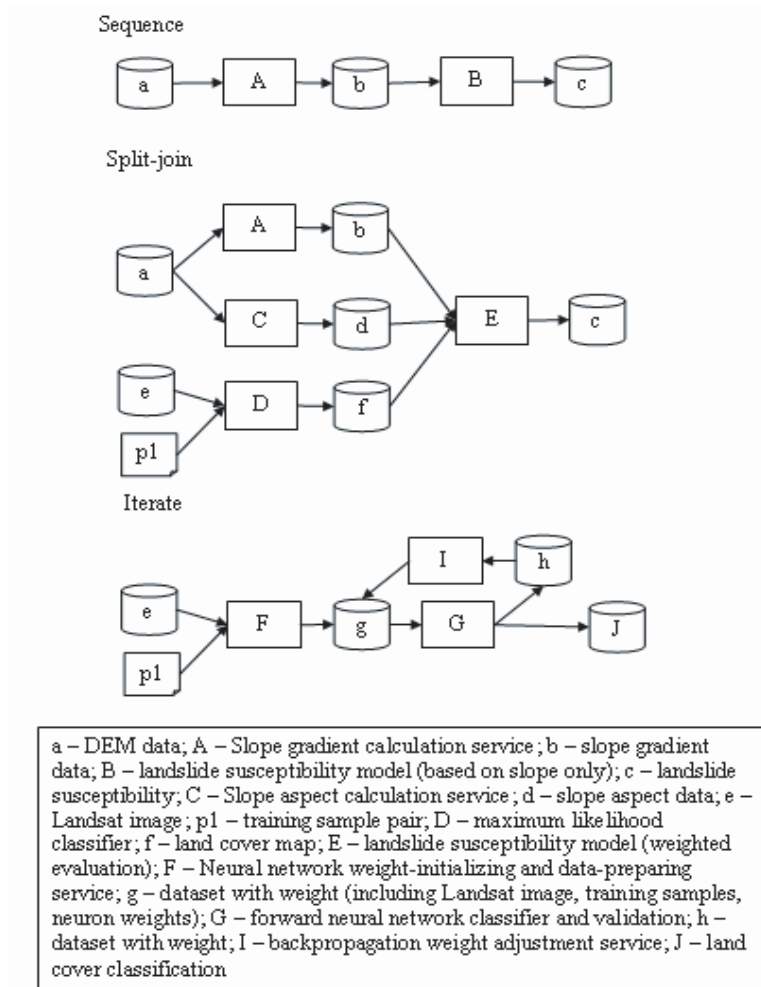
Assembling individual geospatial Web services into a service chain for representing a more complicated geospatial model and process flow proves to be essential for

complex geospatial applications and knowledge discovery (Di, 2004a; Di, 2004c; Di, 2005; Di, 2005a). Such a process is called geospatial Web service orchestration. Service orchestration introduces a new method of application development and has great potential to reduce the development time and effort for new applications.

Processes of Orchestration

Geospatial Web service orchestration, in the context of this chapter, is nearly exchangeable with choreography, to embrace any service chaining activities, including

Figure 7. Typical composite processes for service chaining



both top-down (orchestration) and bottom-up (choreography) service composition. Typical processes in service chain include *Sequence*, *Split*, *Split + Join*, *Choice*, *Any-Order*, *Condition*, *If-Then-Else*, *Iterate*, *Repeat-While*, and *Repeat-Until*. Figure 7 illustrates some typical composite processes by aggregating Web services or orchestrating (conducting) heterogeneous Web services.

Orchestration Approaches

Geospatial Web service orchestration can be completed in three approaches: transparent, translucent, and opaque (Alameh, 2001; Di, 2004b; ISO/TC211, 2002b).

Transparent Geospatial Web Services Orchestration

With transparent geospatial Web service orchestration, the user plays a central role in finding all the required geospatial Web services and geospatial data. Once the user identifies all the required services or data, a composite process can be achieved by two ways: The user invokes the services one by one in a user-controlled sequence (see Figure 8(a)), or the user creates a composite process in Business Process Execution Language (BPEL) or OWL-based Web service ontology (OWL-S) and execute the composite process using a proper engine (see Figure 8(b) for a composite sequence) (Andrews et al., 2003; Arkin et al., 2005; Martin et al., 2004). For the former, the user takes care of the sequence of requests as well as the transfer of information between geospatial Web services. For the latter, the user pre-composes the sequence and the information or message is directly relayed between geospatial Web services.

Translucent Geospatial Web Service Orchestration

A workflow management service or a composition service is used in such a case to coordinate all the invocation and scheduling of component Web services. The user or expert only needs to prepare an abstract composite service and store it in a composite service repository. The user later on can view and adjust the component Web service using a composite service editor/viewer.

Opaque Geospatial Services Orchestration

From the user viewpoint, an opaque geospatial Web service chain acts exactly the same as a simple Web service — the user sets all the required parameters, submits the request, and gets back the results without knowing what are the component Web services. From the system viewpoint, the opaque Web service chain can be a

Figure 8. Transparent geospatial Web service chaining (Alameh, 2001)

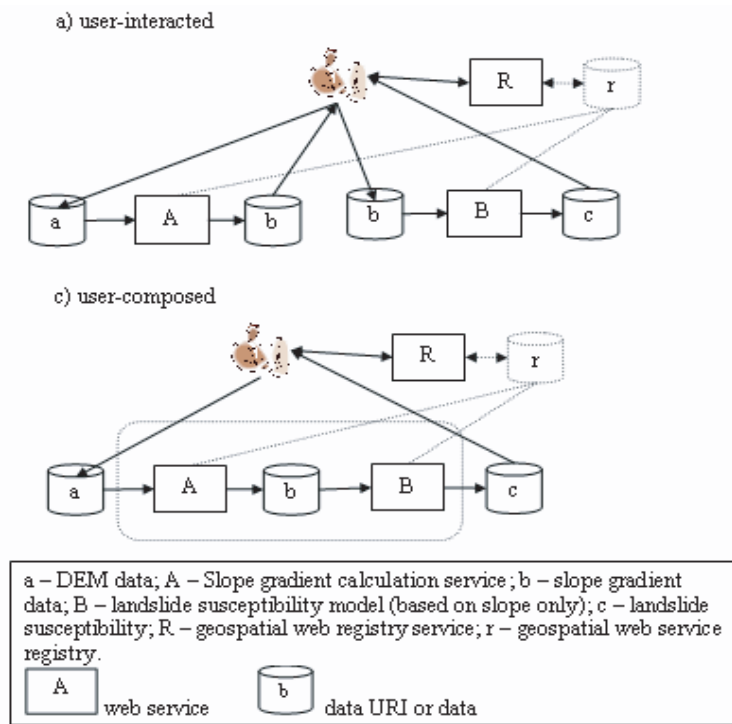


Figure 9. Translucent geospatial Web service chaining (Alameh, 2001)

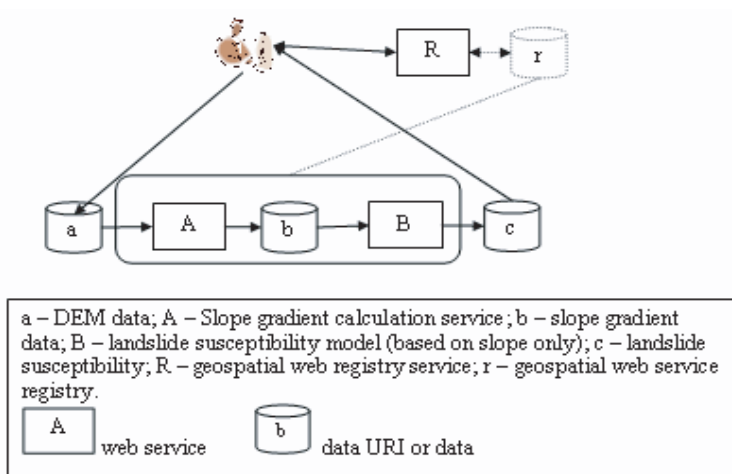
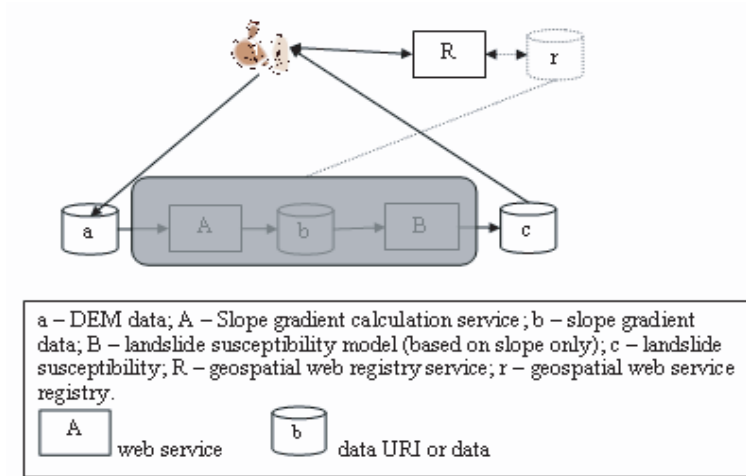


Figure 10. Opaque geospatial Web service chaining (Source: Alameh, 2001)



pre-defined composite Web service or a dynamically-chained Web service. For the former, the component Web services are fixed for the service chain even though the user does not know what they are. These can be pre-defined by experts. For the latter, the component Web services can only be fixed when the specific request is fixed. Some intelligent reasoning is used to complete the chaining instantly. This is especially useful in serving earth-science data when the data needs some intermittent process to meet customized requirements of users (Di et al., 2005).

Geospatial Web Service Solutions

The emerging standards enable the sharing of spatial data (data service) and processing functions (Web services) over the Web. Envisioning the prospect of avoiding the headache of formatting, geo-referencing, and integrating spatial data with existing data, all sectors (governments, industries, educational institutes, and individuals) are getting involved in publishing their geospatial Web services following these standards (Sayar, Pierce, & Fox, 2005). More than 260 products have claimed to be compliant with or implementing OGC specifications or interfaces (OGC, 2005b). Figure 11 shows part of the summary report for the compliant or implementing OGC specifications and services. Web map service (WMS), Web feature service (WFS), and Geography Markup Language (GML) are among the most popular specifications. By further analyzing the products claimed to be compliant with OGC

Figure 11. Part of the statistical summary of OGC specification or interface implementations (Source: OGC, 2005b)

Total (Compliant)	Specification / Version	Abrv / Version
107 (16)	Web Map Service (1.1.1)	WMS 1.1.1
105 (0)	Web Map Service (1.0)	WMS 1.0
103 (0)	Web Map Service (1.1)	WMS 1.1
67 (7)	Web Feature Service (1.0)	WFS 1.0
31 (0)	Geography Markup Language (2.1.1)	GML 2.1.1
28 (0)	Geography Markup Language (2.0)	GML 2.0
27 (0)	Filter Encoding (1.0)	Filter 1.0
21 (0)	Web Map Context Documents (1.0)	WMC 1.0
20 (7)	Simple Features - SQL - Normalized Geometry (1.1)	SFS(NG) 1.1
19 (0)	Styled Layer Descriptor (1.0)	SLD 1.0
19 (0)	Web Feature Service (Transactional) (1.0)	WFS(T) 1.0
17 (0)	Geography Markup Language (3.0)	GML 3.0
14 (1)	Catalog Interface (1.0)	CAT 1.0
14 (10)	Simple Features - SQL - Types and Functions (1.1)	SFS(TF) 1.1
12 (0)	Simple Features - SQL (1.1)	SFS 1.1
12 (0)	Gazetteer (0.8)	Gaz 0.8
12 (0)	Geography Markup Language (2.1.2)	GML 2.1.2
12 (0)	Web Terrain Server (0.3.2)	WTS 0.3.2
11 (5)	Simple Features - SQL - Binary Geometry (1.1)	SFS(BG) 1.1
10 (0)	Catalog Interface (1.1.1)	CAT 1.1.1
9 (0)	WMS Part 2: XML for Requests using HTTP Post (0.0.3)	WMS POST 0.0.3
8 (0)	Web Registry Server (0.0.2)	WRS 0.0.2

specifications, this section gives users a head-start guide to know about commercial and open-source programs.

Server-Side Solutions

More and more geospatial software products have been published using standards-compliant specifications, for example, GeoMedia WebMap (Intergraph), and ArcWeb services (ESRI). Table 3 lists some of the server-side products that support OGC specifications, including WMS, WFS, catalog service specification (CAT), GML, simple feature — SQL (SFS), and WCS. To achieve more interoperability, several data services from governments start to implement these specifications. Geospatial-one-stop (geodata.gov) is one of them which support a wide suite of OGC specifications.

Client-Side Solutions

On the client side, more and more client software have claimed that they support the access of geospatial data and services through open geospatial standards, for

Table 3. List of standard-compliant geospatial web-services (server side) (OGC, 2005b)

Specification	Product
WMS 1.0, 1.1, 1.1.1	MapServer 3.5, 4.2; CarbonTools 2; Drill Down Server Web services; TakukGIS Developer Kernel; POLYGIS MAPSERVER 9.1; SuperGIS 1.2; SuperWebGIS 1.2; e-NeXt server 6.2; GIS Broker 3.3; TerraVision WMS Server 1.1; PostGIS / PostgreSQL 0.8.0; rasdaman 5.1; XIS (Extensible information Systems) 1.1; WebPix WMS Web service 1.0; Geomatica - WebServer Suite 10.0; shp2gml GeoBroker 1.1; Oracle Application Server MapViewer, 10g Release 2 (10.1.2); COP-WDS 1.1; SpatialFX 4.0; Navici; GIS/LINE WMS 1.0; DMAP GIDB Portal OGC Web services WMS 1.1.1, 1.1.0; MIT Ortho Server ; MiraMon Map Server 5.0; MapXtreme 2004 6.1, Java Edition 4.7; WEGA-MARS 3.1, 4.1; LuciadMap 4.2, 5; deegree owsProxy 0.5, deegree Web Map Service 1.1.2; Gothic Integrator: Java Edition (GI:JE) WMS, Gothic WebMapper v4-1a-12a-2; JGisBr - JGisCatalogo V2; WMSControl. Net 1.0b; RedSpider Catalog 2.0, Web 3.0, 3.1, 3.3; International Interfaces AWMS Server and Cookbook 1; GeoMedia; XtraWMS 0.38; JCarnacGIS 2.0; FasterMap 1.0; Geologica Map Server 1.1; GeoServer 1.2; GeoSecurity 2.0; GIS Portal 2.0; Global Aerial and Satellite Image Service 3.0g-business integrator 2.0; Exposure Image Server 1, Spatial Server 1; SciMapProxy 1; SciMapServer 2.30; GenaWare 8; sisIMS 2004.0301; GE Smallworld Internet Application Server; PlanAccess WMS 1.5; FreeStyle Suite 1.15; NAVImap 1; ArcIMS 3.0, 4.0, 4.0.1, 9.0; GIS Portal Tool Kit 2.0; iSMART 4.3; Demis Map Server 2.0, 3.0, 4.0; XMap Web 3.0; CubeServ – Web Map Server 2.10.7, Cascading Web Map Server 4.0.2; sdi.suite terraCatalog 2.0, terrainServer 1.0; ArcIMS to OGC Connector 1.2; Map Manager 3.2; Service manager 3.0; Web Enterprise Suite 3.1; Web Feature Portrayal Service 1.2; SpatialAce Web Map Server 2.0; CARIS Chartserver 2.0, Spatial Fusion 2.5.1, 3.0, 3.2, Cascading Web Map Server 3.2; HPDWeb 1.1; Cadcorp GeognoSIS.NET V6.1, SIS Feature Server 6, Map Server 5.2, 6, 6.1; GeoMapServer 1.0; Autodesk WMS Extention 6.3; SICAD/open – IMS 6.0, SICAD/Raster – IMS 6.0, SICAD/SD – IMS 6.0; GeoServer 4.1
WFS 1.0	MapServer 4.2; CarbonTools 2; TakukGIS Developer Kernel 8.x; SuperWebGIS 1.2; Geomatica-WebServerSuite 10.0; GEOINTEGRATION 2.0; GIS/LINE WFS 1.0; LuciadMap 5; deegree Web Feature Service 1.2.1; Gothic WebMapper v4-1a-12a-2; RedSpider Catalog 2.0, Web 3.0, 3.3; GeoMedia; JCarnacGIS 2.0; GeoServer 1.2; g.business integrator 2.0, organizer 2.0; FreeStyle Suite 1.15; Cartalinea Geographic Data Server (GDS) 1.4; ArcIMS 4.0, 4.0.1, 9.0, 9.1; GIS Toolk Kit 2.0; CubeServ – Web Feature Server 2.10.7; sdi.suite terraCatalog 2.0; Meta Manager – WFS 5.0; Service Manager 3.0; CARIS Spatial Fusion 3.2; Cadcorp GeognoSIS.NET V6.1; LIDS Application Server 6.5.3
GML 1.0, 2.0, 2.1.1, 2.1.2, 3.0, 3.1	MapServer 4.2; CarbonTools 2; TakukGIS Developer Kernel 8.x; SuperWebGIS 1.2; WFS Server 1.0; PCI Web Prototypes 0.1; shp2gml GeoBroker 1.1; Envinsa 3.5; MapInfo Routing J Server 3.2; MapMaker J Server 3.0; MapXtreme 2004 6.1, Java Edition 4.7; LuciadMap 5; deegree Web Feature Service 1.2.1; Gothic WebMapper v4-1a-12a-2; RedSpider Web 3.3; GeoMedia; JCarnacGIS 2.0; FasterMap 1-0; g.buisness integrator 2.0; SelfFeatureServer 1.3; Cartalinea Geographic Data Server (GDS) 1.4; ArcIMS; iSMART 4.3; CubeServ – Web Map Server; Map manager 3.2; XchainJ 1.1; LIDS Application Server 6.5.3
SFS	TakukGIS Developer Kernel 8.x; PostGIS / PostgreSQL 0.8.0; Oracle Application Server Map Viewer, 10g, Oracle8i Spatial 8.1.x; MapInfo SpatialWare 4.5-4.8; JGisBr-JGisCatalogo v2; GeoMedia Data Server for Oracle Object Model Server 05.01; IBM DB2 Spatial Extender 7.1, 8.1; GeoTask Server on IBM DB2, on Oracle 9i; sisNET 2004; ArcSDE; TOPOBASE Geodataserver; AEDIDB (option AEDIDB-SimpleFeature) 3.5
CAT 1.0, 2.0	deegree Web Catalog Service; JGisBr- JGisCatalogo V2; RedSpider Catalog 2.0; SMMS GeoConnect 5.x; ogcatsrv 1.2; ArcIMS Metadata Server; GIS Portal Tool Kit 2.0 con terra Catalog Server 1; Meta Miner 3.1
WCS 0.7, 1.0	Geomatica –WebServer Suite 10.0; MiraMon Map Server 5.0; LuciadMap 5; deegree Web Coverage Service 1.1.5; RedSpider Catalog 2.0, Web 3.1, 3.3; ArcGIS Server 9.0; GIS Portal Tool Kit 2.0

*Table 4. List of standard-compliant geospatial web-services (client-side only)
(Source: OGC, 2005b)*

Specification	Product
WMS 1.0, 1.1, 1.1.1	Gaia 2; J2ME WMS Client 1.0.1; Skylab GPS Simulator 1.1; WebMap Composer 2.03; Arc3WMS 1.0; XIS View Point 1.1; NACMAP4.0; Internet Mapping Framework 1.0.0; Java based wmsClient 1; deegree iGeoPortal 0.3; deeJUMP 1.0; RedSpider Studio 3.0; GeoMedia Viewer; XtraZWMSClient 0.27; Community Mapbuilder mapbuilder-lib; g.business navigator 2.0; Mapplets 5.0; sclMapJSClient; WebWraptor 1.0; sisVIEW 1.0; ArcExplorer Web; Chameleon 1.x; CubeXPLO 2.10.7; sdi.suite mapClient 1.0; Mapbender 1.4, 2.0; CARIS WMS Client 1.0; Cadcorp apSIS 6; SICAD-IS D-HTML Client 6.0, HTML Client 6.0, Java Client 6.0
WFS 1.0	Gaia 2; WebMap Composer 2.03; deegree iGeoPortal 0.3; deeJUMP 1.0; RedSpider Studio 3.0; GeoMedia Viewer; g.business organizer 2.0; SclMapJSClient; ArcGIS Data Interoperability Extension 9.0; Chameleon 1.x
GML 1.0, 2.0, 2.1.1, 2.1.2, 3.0, 3.1	Gaia 2; GO Loader 0.3; deegree iGeoPortal 0.3; RedSpider Studio 3.0; GeoMedia Viewer; sisVIEW 1.0; ArcGIS Data Interoperability Extension 9.0; CubeXPLO 2.10.7;
SFS	Munsys Cadasrral/Drainage/Electricity/Roads/Sewer/Water; MapInfo Professional 8.0; JGisBr-JGisEditor V2; HiRDB Spatial Search Plug-in 01-01; Spatial Database Engine for DB2 Datajoiner (Infomix or Oracle) 3.0.2; Cadcorp SIS 5.2
CAT 1.0, 2.0	JGisBr-JGisEditor V2; ogcatclt 1.2; Catalog Server for Informix 1; terraCatalog 1.2

example, ArcExplorer Web (ESRI), Interoperability Add-on for ArcGIS (ESRI), MapInfo Professional (MapInfo), and GeoMedia (Intergraph). Most server-side vendors have their own clients to access their standards-compliant servers with similar names. Table 4 gives some extra client applications that implement the OGC specifications.

Open-Source Solutions

One noticeable aspect is that open source software products are emerging as a key force in developing standard-compliant geospatial Web services and clients. For the server side, examples of geospatial Web services are LAITS OGC WCS, WICS, and WMS from George Mason University (<http://laits.gmu.edu>), MapServer (University of Minnesota), and GeoTools Web Map Server (<http://www.geotools.org>). For the client side, examples of standards-supported clients include the LAITS client (George Mason University) and the MapServer client (University of Minnesota), and so forth. These offer researchers the opportunity to explore the mechanisms in designing, developing, implementing, and testing of geospatial Web services.

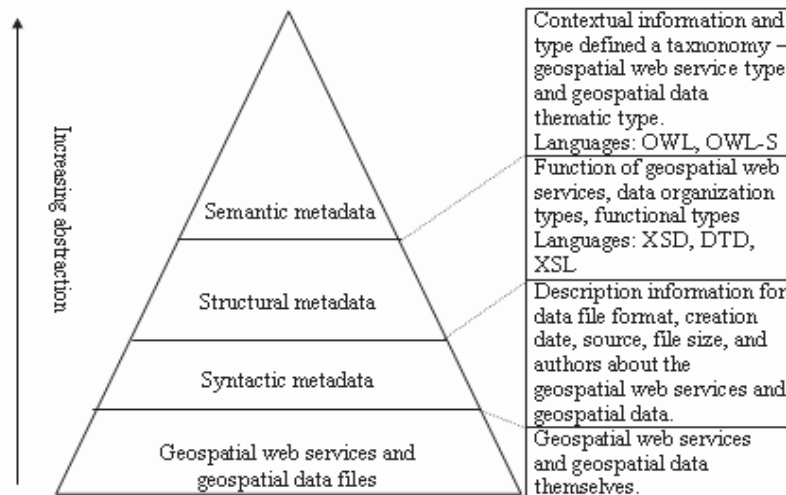
Semantic Issues on Geospatial Web Service

With ever-growing geospatial Web services and geospatial data, it is very important to find the proper data and service matching specific requirements. Metadata is always used in describing and discovering geospatial Web services. However, mismatching may arise due to semantic ambiguity, or the same term used in different domains may lead to a different meaning (Sivashanmugam et al., 2003). This section reviews the semantic issues and their possible solutions for geospatial Web services.

Semantic Issues

Metadata for geospatial Web services can be classified into three levels: syntactic, structural, and semantic (Sheth, 1999, 2003; Sivashanmugam et al., 2003). Syntactic metadata describes basic sources and simple authorship information about geospatial Web services, such as data file format, creation date, sources, file size, and authors. Structural metadata describes the structure of geospatial data and the functions of the geospatial Web services, such as XML schema, data organization types (vector or raster), and functional types (data conversion, manipulation). Semantic metadata defines the context or domain of the geospatial data and geospatial services, such as thematic type for data (e.g., landslide in geoscience, population distribution in economic geography) and functional domain types (e.g., vegetation index computation in remote sensing, land surface temperature estimation in climate).

Figure 12. Data and their semantic hierarchy (Source: Sheth, 1999, 2003; Sivashanmugam et al., 2003)



Semantic issues for interoperability have been recognized in many distributed multiple data systems (Kashyap & Sheth, 1996; Sheth, 1991; Sivashanmugam et al., 2003). Similar semantic issues rise for geospatial Web service chaining. For example, a geospatial service chain is required to compute a landslide susceptibility map, given a large repository of geospatial Web services. A landslide susceptibility model (as a Web service) may take seven input datasets: surface slope, surface aspect, base rock slope, base rock aspect, depth to base rock, soil moisture, and vegetation. All these input data types may not be directly available at data repositories, but they can be derived from some other Web services. Therefore, the first task is to find the Web services with output matching the required data type. The following problems may occur if semantics is not considered.

Semantic Issue 1: Ambiguous Match Due to Lack of Semantics

Each output of the services may be represented by a URI (universal resource identification) string, or other basic data types (e.g., integer, double, or single). If the match criteria are only defined as the match on data types (e.g., the only search criterion being that the output of a service is a URI string), the returned results may be overwhelming, containing numerous improper matches. This is because a URI string may carry different meaning.

Semantic Issue 2: Ambiguous Match Due to Incomplete Semantics

If the search criteria is based upon the service structure (e.g., a service with one input of DEM and one output of slope), the returned results may not satisfy the exact requirement; an improper computation algorithm may have been used to derive the slope. There are many algorithms for slope calculations: neighborhood method, quadratic surface method, maximum slope method, and maximum downhill method. If a desired method is quadratic surface method, these matches with other methods are mistakes for this searching task.

Semantic Issue 3: Missed Match Due to Lack of Relationship Semantics

A search for a vegetation cover map data service may fail to retrieve any match although a land cover map data service may exist in the repository. It is common knowledge that a land cover map can be actually used in place of a vegetation map since the vegetation categories (forest land, grass land, cultivated land) available in the land cover map may be sufficient for this purpose. Without a clear definition of the relationship between vegetation cover and land cover maps in the semantics of the metadata, the search would not be able to make this intelligent association.

Semantic Solutions

One solution to the above problems is to add semantic information to each geospatial data and geospatial Web services using a proper common ontology. An ontology is a specific set of vocabulary and relationships that have explicit assumptions about the intended meanings and relationships of the vocabulary in the set (Sivashanmugam et al., 2003). It is a technology that enables the computer-understandable description of Web services. Emerging standards for ontology are OWL, OWL-S, and Semantic Web services Language (SWRL) (Dean & Schreiber, 2004; Horrocks, Patel-Schneider, Boley, Tabet, Grosz, & Dean, 2003; Martin et al., 2004). These standards evolve the description of Web services from a syntactical description to a semantic description, mainly based on description logic. These can be used to describe the taxonomy of geospatial data and geospatial Web services and define their relationships with one another. Then, a reasoner can be used to validate the logic and induce the relationship underlying every term. A match can be made using this knowledge by matching both data semantics for input and output and functional semantics for the Web services (Di, 2004c; Di, 2005a; Di et al., 2005; Sivashanmugam et al., 2003; Zhao et al., 2004).

Conclusions

Web services give promising prospects in interoperability across applications and organizational boundary, independence from languages, and ease of maintenance. Geospatial data and service dissemination can take advantage of Web services to achieve higher usability and efficiencies. This chapter covered all aspects of the life cycle for geospatial Web services, including standards, implementation, discovery, and orchestration. It also reviewed example solutions and applications leveraging the standards for geospatial Web services. Semantic issues and possible solutions are discussed.

Web service is an emerging technology. Standards, specifications, and practices for Web service are evolving. The same is also true to geospatial Web services. Many issues in the geospatial Web services still need to be explored. The following lists a few:

1. **A Web service is originally designed to be stateless:** However, transaction is required in many cases because geospatial modeling is a complicated process requiring multiple interactions. How to evolve the geospatial Web service to deal with transactions should be further studied.

2. **Web service uses plain XML as the base to communicate between applications:** Compared to other distributed computing approaches, Web service is relatively poor in performance and lack of security. These issues should be researched for geospatial Web services as well.
3. **Semantics for geospatial Web services is in development:** Many research issues need to be solved before an operational geospatial semantic Web can be established.

Acknowledgments

The authors of this chapter, as well as some of the example systems and services discussed in the chapter, were supported fully or partially with grants from the NASA Earth Science Data and Information System Project (ESDISP) (NCC5-645, PI: Dr. Liping Di), NASA Earth Science Technology Office (ESTO) (NAG-13409, PI: Dr. Liping Di), NASA REASoN program (NNG04GE61A, PI: Dr. Liping Di), and National Geospatial-intelligence Agency (NGA) University Research Initiative (NURI) (HM1582-04-1-2021, PI: Dr. Liping Di). We also sincerely thank to our colleagues at Laboratory for Advanced Information Technology and Standards, George Mason University. Numerous discussions with our colleagues at LAITS have brought forth much inspiration and enlightenment and shaped the chapter.

References

- Akinci, H. (2004, 12-23 July). *Geospatial Web services for e-municipality*. Paper presented at the XXth ISPRS Congress, Istanbul, Turkey.
- Alameh, N. S. (2001). *Scalable and extensible infrastructures for distributing interoperable geographic information services on the Internet*. Cambridge, MA: Massachusetts Institute of Technology..
- Andersson, E., Greenspun, P., & Grumet, A. (2003). *Software engineering for internet applications*. Retrieved from <http://philip.greenspun.com/seia/>
- Andrews, T., Curbera, F., Dholakia, H., Golland, Y., Klein, J., Leymann, F., et al. (2003). *Business Process Execution Language for Web Services (Version 1.1)*. Retrieved from <http://dev2dev.bea.com/webservices/BPEL4WS.html>
- Arkin, A., Askary, S., Bloch, B., Curbera, F., Golland, Y., Kartha, N., et al. (2005, July 13). *Web Services Business Process Execution Language Version 2.0*

- (Working Draft). Retrieved from <http://www.oasis-open.org/apps/org/workgroup/wsbpel/>
- Bacharach, S. (2005). *OGC to begin geospatial semantic Web interoperability experiment*. Retrieved August 20, 2005, from <http://www.opengeospatial.org/press?page=pressrelease&year=0&prid=220>
- Booth, D., Haas, H., McCabe, F., Newcomer, E., Champion, M., Ferris, C., et al. (2004, February 11). *Web services architecture* (W3C Working Group Note). Retrieved August 8, 2005, from <http://www.w3.org/TR/2004/NOTE-wsdl20-primer-20050803/>
- Booth, D., & Liu, C. K. (2005, August 3). *Web Services Description Language (WSDL) Version 2.0 Part 0: Primer* (W3C Working Draft). Retrieved from <http://www.w3.org/TR/2005/WD-wsdl20-primer-20050803/>
- Burry, L. (2001). *OpenGIS implementation specification: Coordinate transformation services, revision 1.00*. Retrieved August 20, 2005, from http://portal.opengeospatial.org/files/?artifact_id=6628
- Chinnici, R., Haas, H., Lewis, A., Moreau, J. -J., Orchard, D., & Weerawarana, S. (2005, August 3). *Web Services Description Language (WSDL) Version 2.0 Part 2: Adjuncts* (W3C Working Draft). Retrieved from <http://www.w3.org/TR/2005/WD-wsdl20-adjuncts-20050803/>
- Chinnici, R., Moreau, J.-J., Ryman, A., & Weerawarana, S. (2005, August 3). *Web Services Description Language (WSDL) Version 2.0 Part 1: Core Language* (W3C Working Draft). Retrieved from <http://www.w3.org/TR/2005/WD-wsdl20-20050803/>
- Christensen, E., Curbera, F., Meredith, G., & Weerawarana, S. (2001, March 15). *Web Services Description Language (WSDL) 1.1* (W3C Note). Retrieved from <http://www.w3.org/TR/2001/NOTE-wsdl-20010315>
- Clement, L., Hatley, A., Riegen, C. Y., & Rogers, T. (2004, October 19). *UDDI Version 3.0.2* (UDDI Spec Technical Committee Draft). Retrieved from <http://uddi.org/pubs/uddi-v3.0.2-20041019.htm>
- Cox, S., Daisey, P., Lake, R., Portele, C., & Whiteside, A. (2004). *OpenGIS Geography Markup Language (GML) implementation specification*. Retrieved August 20, 2005, from <http://schemas.opengis.net/gml/3.1.0/base/>
- Daly, M. (2001). *OpenGIS implementation specification: Coordinate transformation services, revision 1.00*. Retrieved August 20, 2005, from http://portal.opengeospatial.org/files/?artifact_id=999
- de La Beaujardiere, J. (2004). *OpenGIS Web map service, Version 1.3*. Retrieved August 20, 2005, from http://portal.opengeospatial.org/files?artifact_id=5316
- Dean, M., & Schreiber, G. (2004). *OWL Web Ontology Language reference*. Retrieved from <http://www.w3.org/TR/2004/REC-owl-ref-20040210/>

- Di, L. (2003, July 21-25). The development of remote-sensing related standards at FGDC, OGC, and ISO TC 211. In *Proceedings of 2003 IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2003)*, Toulouse, France (p. 4).
- Di, L. (2004a). Distributed geospatial information services-architectures, standards, and research issues. In *The International Archives of Photogrammetry, Remote Sensing, and Spatial Information Sciences* (Vol. XXXV, Part 2, Commission II, p. 7).
- Di, L. (2004b). GeoBrain — A Web services based geospatial knowledge building system. In *Proceedings of the Fourth annual NASA Earth Science Technology Conference*, Palo Alto, CA (p. 8).
- Di, L. (2004c, September 20-24). A framework for intelligent geospatial knowledge discovery at Web service environment. In *Proceedings of 2004 IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2004)*, Anchorage, AK (p. 8).
- Di, L. (2005, July 25-29). Customizable virtual geospatial products at Web/grid service environment. In *Proceedings of 2005 IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2005)*, Seoul, Korea (p. 4).
- Di, L. (in press). A framework for construction of Web-service based intelligent geospatial knowledge systems. *Journal of Geographic Information Science*.
- Di, L. (in press). The open GIS Web service specifications for interoperable access and services of NASA EOS Data. In Qu, J. etc (Eds.), *Earth science satellite remote sensing*. Springer-Verlag.
- Di, L., & McDonald, K. (2005). The NASA HDF-EOS Web GIS software suite (NWGISS). In Qu, J. etc (Eds.), *Earth science satellite remote sensing*. Springer-Verlag.
- Di, L., Yang, W., Deng, M., Deng, D., & McDonald, K. (2002). Interoperable access of remote sensing data through NWGISS. In *Proceedings of 2002 IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2002)*, Toronto, Canada (pp. 255-257).
- Di, L., Zhao, P., Yang, W., Yu, G., & Yue, P. (2005, July 25-29). Intelligent geospatial Web services. In *Proceedings of 2005 IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2005)*, Seoul, Korea (p. 3).
- Duschene, P., & Sonnet, J. (2005a). *OGC WCS change request: Support for WSDL & SOAP, Version 0.1.0*. Retrieved August 20, 2005, from http://portal.opengeospatial.org/files/?artifact_id=9540
- Duschene, P., & Sonnet, J. (2005b). *OGC WMS change request: Support for WSDL & SOAP*. Retrieved August 20, 2005, from http://portal.opengeospatial.org/files/?artifact_id=9541

- EOP (2003). Coordinate geographic data acquisition and access; the national spatial data infrastructure, *Federal Register*, 68, 10619-10633.
- Evans, J. D. (2003). *OpenGIS Web coverage service (WCS) Version 1.0.0*. Retrieved August 20, 2005, from https://portal.opengeospatial.org/files/?artifact_id=3837
- Evans, J. D., & Bambacus, M. J. (2005, 25-29 July). *NASA's earth-sun system gateway: An open standards-based portal to geospatial data and services*. Paper presented at the IEEE 25th Anniversary International Geoscience and Remote Sensing Symposium, Seoul, Korea.
- Fielding, R. T. (2000). *Architectural styles and the design of network-based software architectures*. Irvine, CA: University of California, Irvine.
- GRASS (2005). *Geographic resource analysis support system*. Retrieved August 10, 2005, from <http://grass.itc.it/>
- Groot, R., & McLaughlin, J. D. (2000). *Geospatial data infrastructure: Concepts, cases, and good practice*. Oxford, England; New York: Oxford University Press.
- GSFC (2005). *HDF-EOS*. Retrieved August 20, 2005, from <http://hdfeos.feos.gsfc.nasa.gov/>
- Gudgin, M., Hadley, M., Mendelsohn, N., Moreau, J. -J., & Nielsen, H. F. (2003, June 24). *SOAP Version 1.2 Part 1: Messaging framework (W3C Recommendation)*. Retrieved from <http://www.w3.org/TR/soap12/>
- Hecht, L., Jr. (2002). Web services are the future of geoprocessing. *GEO World*.
- Hirsch, F., & Just, M. (2003, May 5). *XML key management (XKMS 2.0) Requirements (W3C Note)*. Retrieved from <http://www.w3.org/TR/2003/NOTE-xkms2-20030505>
- Horrocks, I., Patel-Schneider, P. F., Boley, H., Tabet, S., Grosz, B., & Dean, M. (2003). *SWRL: A Semantic Web Rule Language combining OWL and RuleML*. Retrieved from <http://www.daml.org/2003/11/swrl/>
- IETF. (2005). *The Internet Engineering Task Force*. Retrieved August 20, 2005, from <http://www.ietf.org/>
- ISO/TC211. (2002a). *Draft new work item proposal: Geographic information — metadata: Part 2 - Metadata for imagery and gridded data*. Retrieved from <http://www.isotc211.org/protdoc/211n1337/211n1337.pdf>
- ISO/TC211. (2002b). *Geographic information services*. Retrieved from <http://www.ncits.org/ref-docs/DIS19119.PDF>
- ISO/TC211. (2003a). *Geographic information — metadata*. Retrieved from http://www.ncits.org/ref-docs/FIDIS_19115.pdf

- ISO/TC211. (2003b). *ISO 19139 dataset implementation model and schema, Version 0.7*. Retrieved August 20, 2005, from http://metadata.dgiwg.org/ISO19115/ISO19115_v0_7detail.htm
- ISO/TC211. (2005). *ISO/TC211 Geographic information/geomatics*. Retrieved August 20, 2005, from <http://www.isotc211.org/>
- Kashyap, V., & Sheth, A. P. (1996). Semantic and schematic similarities between database objects: A context-based approach. *VLDB Journal*, 5(4), 276-304.
- Kolodziej, K. (2004). *OpenGIS Web map server cookbook, Version 1.0.2*. Retrieved August 20, 2005, from http://portal.opengeospatial.org/files/?artifact_id=7769
- Kralidis, A. (2005). *Geospatial Web services: An evolution of geospatial data infrastructure*. Ottawa, Ontario, Canada: Carleton University.
- LAITS (2005). *Mobilization of NASA EOS data and information through Web services and knowledge management technologies for higher-education teaching and research*. Retrieved August 20, 2005, from <http://geobrain.laits.gmu.edu>
- Lieberman, J., Reich, L., & Vretanos, P. (2003). *OWS 1.2 UDDI experiment, Version 0.5*. Retrieved August 20, 2005, from http://portal.opengeospatial.org/files/?artifact_id=1317
- Mabrouk, M. (2005). *OpenGIS® location service (OpenLS) implementation specification: Core services, Version 1.1*. Retrieved August 20, 2005, from http://portal.opengeospatial.org/files/?artifact_id=8836
- Martell, R. (2004). *OGC™ catalogue services – ebRIM (ISO/TS 15000-3) profile of CSW*. Retrieved from https://portal.opengeospatial.org/files/index.php?artifact_id=6495
- Martin, D., Burstein, M., Hobbs, J., Lassila, O., McDermott, D., McIlraith, S., et al. (2004, November 22). *OWL-S: Semantic markup for Web services (W3C Member Submission)*. Retrieved from <http://www.w3.org/Submission/2004/SUBM-OWL-S-20041122/>
- Mateos, C., Zunino, A., & Campo, M. (2005). Integrating intelligent mobile agents with Web services. *International Journal of Web Services Research*, 2(2), 85-103.
- Nebert, D. (2005). *Current standards, specifications, and practices of relevance to SDI development*. Retrieved August 20, 2005, from <http://www.anzlic.org.au/pubinfo/2413322725>
- Nebert, D., & Whiteside, A. (2004). *OpenGIS catalogue services specification*. Retrieved from http://portal.opengeospatial.org/org/files/?artifact_id=5929&version=1

- OASIS. (2005). *Organization for the Advancement of Structured Information Standards*. Retrieved August 20, 2005, from <http://www.oasis-open.org/home/index.php>
- OASIS/ebXML. (2003). *OASIS/ebXML registry information model V2.5*. Retrieved from <http://www.oasis-open.org/committees/regrep/documents/2.5/specs/ebrim-2.5.pdf>
- OGC. (2004). *Critical infrastructure protection initiative, Phase 2 (CIPI2)*. Retrieved August 20, 2005, from <http://www.opengeospatial.org/initiatives/?iid=66>
- OGC. (2005a). *OpenGIS specifications*. Retrieved August 20, 2005, from <http://www.opengeospatial.org/specs/download.php?view=IS&keyword=&excel=1>
- OGC. (2005b). *Resources — registered products*. Retrieved from <http://www.opengeospatial.org/resources/?page=products>
- Peltz, C. (2003). *Best practices for Web services development*. Retrieved August 20, 2005, from <http://devresource.hp.com/drc/resources/WSBestPractices/index.jsp>
- Peltz, C., & Rogers, C. (2003). *Leveraging open source for Web services development*. Hewlett-Packard Company.
- Sayar, A., Pierce, M., & Fox, G. (2005). *OGC compatible geographical information services*. Bloomington, IN: Indiana University.
- Sheth, A. P. (1991). Semantic issues in multidatabase systems. *SIGMOD Record*, 20(4), 5-9.
- Sheth, A. P. (1999). Changing focus on interoperability in information systems: From system, syntax, structure to semantics. In M. F. Goodchild, M. J. Egenhofer, R. Fegeas, & C. A. Kottman (Eds.), *Interoperating geographic information systems* (pp. 5-30). Boston: Kluwer Academic Pub.
- Sheth, A. P. (2003). Semantic meta data for enterprise information integration. *DM Review*, 13, 52-54.
- Sivashanmugam, K., Sheth, A., Miller, J., Verma, K., Aggarwal, R., & Rajasekaran, P. (2003). Metadata and semantics for Web services and processes. In W. Benn, P. Dadam, S. Kirn & R. Unland (Eds.), *Datenbanken und Informationssysteme: Festschrift zum 60. Geburtstag von Gunter Schlageter* (pp. 245-271). Hagen, Germany: Fachbereich Informatik.
- Skonnard, A. (2002). The XML files: The birth of Web services. *MSDN Magazine*.
- Sonnet, J. (2004). *OWS 2 common architecture: WSDL SOAP UDDI, Version 1.0.0*. Retrieved August 20, 2005, from https://portal.opengeospatial.org/files/?artifact_id=8348

- Sonnet, J. (2005). *OpenGIS Web map context implementation specification, Version 1.1*. Retrieved August 20, 2005, from https://portal.opengeospatial.org/files/?artifact_id=8618
- Sonnet, J., & Savage, C. (2003). *OWS 1.2 SOAP experiment report*. Retrieved August 20, 2005, from http://portal.opengeospatial.org/files/?artifact_id=1337
- Tsalgatidou, A., & Pilioura, T. (2002). An overview of standards and related technology in Web services. *Distributed and Parallel Databases*, 12, 135-162.
- Vaughan-Nichols, S. J. (2002). Web services: Beyond the hype. *Computer*, 35(2), 18-21.
- Vedamuthu, A. S. (2005, August 3). *Web Services Description Language (WSDL) Version 2.0 SOAP 1.1 binding* (W3C Working Draft).
- Voges, U., & Zenkler, K. (2005). *OpenGIS catalog services specification 2.0 — ISO19115/ISO19119 application profile for CSW 2.0, Version 0.9.3*. Retrieved from https://portal.opengeospatial.org/files/?artifact_id=8305
- Vretanos, P. A. (2005). *OpenGIS Web feature service implementation specification, Version 1.1.0*. Retrieved August 20, 2005, from https://portal.opengeospatial.org/files/?artifact_id=8339
- W3C. (2005a). *W3C XML schema*. Retrieved August 20, 2005, from <http://www.w3.org/XML/Schema>
- W3C. (2005b). *World Wide Web Consortium*. Retrieved August 20, 2005, from <http://www.w3.org/>
- Wei, Y., Di, L., Zhao, B., Liao, G., Chen, A., Bai, Y., et al. (2005, 25-29 July). *The design and implementation of a grid-enabled catalogue service*. Paper presented at the IEEE 25th Anniversary International Geoscience and Remote Sensing Symposium, Seoul, Korea.
- Whiteside, A. (2005). *OGC Web services common specification*. Retrieved August 20, 2005, from https://portal.opengeospatial.org/files/?artifact_id=8798
- Whiteside, A., Müller, M. U., Fella, S., & Warmerdam, F. (2005). *OGC Web coordinate transformation service (WCTS) draft implementation specification, Version 0.3.0*. Retrieved August 20, 2005, from https://portal.opengeospatial.org/files/?artifact_id=8847
- Wilkes, L. (2005). *The Web services protocol stack*. Retrieved August 20, 2005, from <http://roadmap.cpdiforum.com/reports/protocols/index.php>
- Winer, D. (1999). *XML-RPC specification*. Retrieved from <http://www.xmlrpc.com/spec>
- Yang, W., & Whiteside, A. (2005). *OGC Web image classification service (WICS) implementation specification, Version 0.3.3*. Retrieved August 20, 2005, from https://portal.opengeospatial.org/files/?artifact_id=8981

- Zhao, P., Chen, A., Liu, Y., Di, L., Yang, W., & Li, P. (2004, 12-13 November). *Grid metadata catalog service-based OGC Web registry service*. Paper presented at the 12th ACM International Workshop on Geographic Information Systems (ACM-GIS 2004), Washington DC.
- Zhao, P., Nusser, S., & Miller, L. (2002). *Design of field wrappers for mobile field data collection*. Paper presented at the Proceedings of the Tenth ACM International Symposium on Advances in Geographic Information Systems (ACM-GIS 2002), McLean, VA.