Encyclopedia of Information Science and Technology

Second Edition

Mehdi Khosrow-Pour
Information Resources Management Association, USA

Information Science REFERENCE

INFORMATION SCIENCE REFERENCE
Hershey • New York

Director of Editorial Content: Kristin Klinger
Director of Production: Jennifer Neidig
Managing Editor: Jamie Snavely
Assistant Managing Editor: Carole Coulson
Cover Design: Lisa Tosheff
Printed at: Yurchak Printing Inc.

Published in the United States of America by
Information Science Reference (an imprint of IGI Global)
701 E. Chocolate Avenue, Suite 200
Hershey PA 17033
Tel: 717-533-8845
Fax: 717-533-8661
E-mail: cust@igi-global.com
Web site: http://www.igi-global.com/reference

and in the United Kingdom by
Information Science Reference (an imprint of IGI Global)
3 Henrietta Street
Covent Garden
London WC2E 8LU
Tel: 44 20 7240 0856
Fax: 44 20 7379 0609
Web site: http://www.eurospanbookstore.com

Library of Congress Cataloging-in-Publication Data

Encyclopedia of information science and technology / Mehdi Khosrow-Pour, editor. -- 2nd ed.
p. cm.
Includes bibliographical references and index.
Summary: "This set of books represents a detailed compendium of authoritative, research-based entries that define the contemporary state of knowledge on technology"--Provided by publisher.
ISBN 978-1-60566-026-4 (hardcover) -- ISBN 978-1-60566-027-1 (ebook)
1. Information science--Encyclopedias. 2. Information technology--Encyclopedias. I. Khosrowpour, Mehdi, 1951-
Z1006.E566 2008
004'.03--dc22
2008029068

British Cataloguing in Publication Data
A Cataloguing in Publication record for this book is available from the British Library.

All work contributed to this encyclopedia set is original material. The views expressed in this encyclopedia set are those of the authors, but not necessarily of the publisher.

Note to Librarians: If your institution has purchased a print edition of this publication, please go to http://www.igi-global.com/agreement for information on activating the library's complimentary online access.

Distributed Geospatial Processing Services

Carlos Granell
Universitat Jaume I, Spain

Laura Díaz
Universitat Jaume I, Spain

Michael Gould
Universitat Jaume I, Spain

INTRODUCTION

The development of geographic information systems (GISs) has been highly influenced by the overall progress of information technology (IT). These systems evolved from monolithic systems to become personal desktop GISs, with all or most data held locally, and then evolved to the Internet GIS paradigm in the form of Web services (Peng & Tsou, 2001). The highly distributed Web services model is such that geospatial data are loosely coupled with the underlying systems used to create and handle them, and geospatial processing functionalities are made available as remote, interoperable, discoverable geospatial services.

In recent years the software industry has moved from tightly coupled application architectures such as CORBA (Common Object Request Broker Architecture—Vinoski, 1997) toward service-oriented architectures (SOAs) based on a network of interoperable, well-described services accessible via Web protocols. This has led to *de facto* standards for delivery of services such as Web Service Description Language (WSDL) to describe the functionality of a service, Simple Object Access Protocol (SOAP) to encapsulate Web service messages, and Universal Description, Discovery, and Integration (UDDI) to register and provide access to service offerings. Adoption of this Web services technology as an option to monolithic GISs is an emerging trend to provide distributed geospatial access, visualization, and processing. The GIS approach to SOA-based applications is perhaps best represented by the spatial data infrastructure (SDI) paradigm, in which standardized interfaces are the key to allowing geographic services to communicate with each other in an interoperable manner. This article focuses on standard interfaces and also on current implementations of geospatial data processing over the Web, commonly used in SDI environments. We also mention several challenges yet to be met, such as those concerned with semantics, discovery, and chaining of geospatial processing services and also with the extension of geospatial processing capabilities to the SOA world.

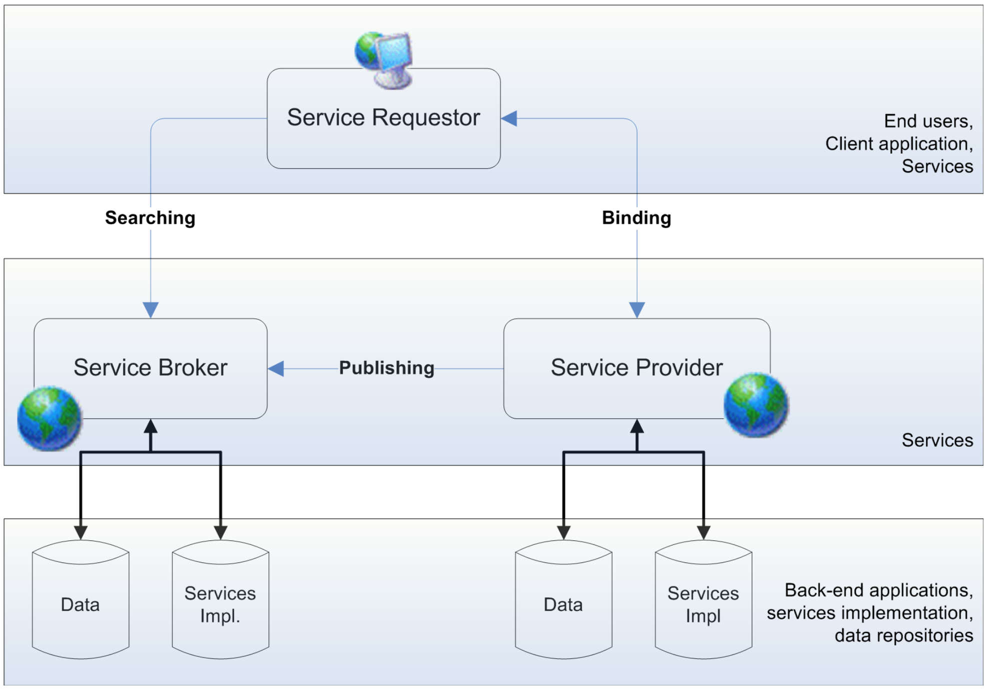
BACKGROUND

Service-Oriented Architecture

A Web service is an executable program available on the Internet. Services are the basic units for creating distributed applications in the context of SOAs. As Papazoglou (2008) stated, SOA is an architectural style to design service-centric applications relying on published and discoverable interfaces. Web services are, by definition, loosely coupled (independent units) and are well described (interface description contains functional properties), thereby promoting one of the goals of SOA: enabling interoperability or the ability of software components to interact with minimal knowledge of the underlying structure of other components (Sheth, 1999). Interoperability is achieved by using standard interfaces (SOA does not focus on the concrete implementations of components) and also by decomposing an application's functionality into modular and flexible services. Such building-block services can be published, discovered, aggregated, reused, and invoked using standard protocols and specifications, independently of the specific technology used to create each component. Essentially SOA introduces a new philosophy for building a pyramid of distributed applications where Web services can be published, discovered, and bound together to create more complex value-added services (Alameh, 2003; Lemmens et al., 2006).

Figure 1 illustrates some of the roles and operations in SOA-based applications. There are three different main SOA roles: service provider, service requestor, and service broker. Each SOA role interacts with others utilizing three basic operations: publication, search, and binding. The service provider publishes service descriptions to the service broker. The service requester searches the required services by querying the service broker and then consumes (binds to) them. Note that often the role of service requester is assigned both to end users (and client applications) and to other services. The latter makes use of two key mechanisms in SOA: service reuse and service chaining to create new,

Figure 1. Roles and operations in SOA



complex, value-added services from simpler, discoverable services. In this sense, services can play the role of service requestor and service provider.

The OWS Service Framework

Within the GIS community, the Open Geospatial Consortium (OGC)—an international industry consortium created in 1994 to develop consensus-based open standards and specifications to support the exchange, sharing, and processing of geospatial data—has adopted a general set of interfaces for a wide range of geospatial Web services (ISO 19119, 2005). Table 1 lists a sample of key OGC Web Services (OWS) categorized as defined in ISO 19119.

These OWS services fall into five categories as follows:

- *Application services* are client-side applications that provide an entry point for end users to find and access geospatial data and services. Among the notable

Table 1. Examples of OGC Web Services

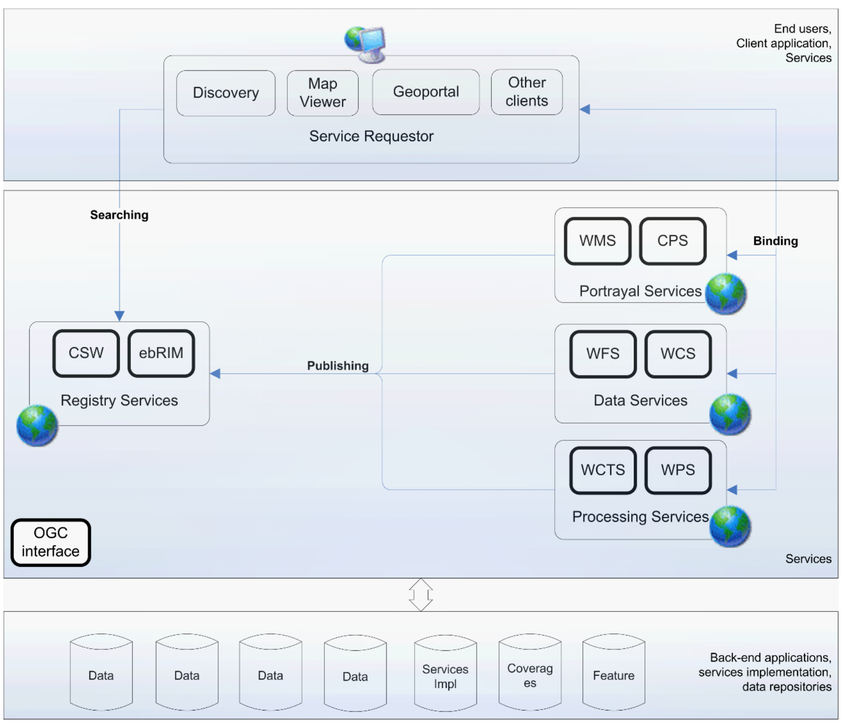
Service Category	Service Name
Application Services	Discovery Application Services Map Viewer Application Services Sensor Web Application Services Geoportal (one-stop portal)
Registry Services	Catalog Service (CSW)
Data Services	Web Feature Service (WFS) Web Coverage Service (WCS)
Portrayal Services	Web Map Service (WMS) Coverage Portrayal Service (CPS)
Processing Services	Web Coordinate Transformation Service (WCTS) Geocoder Services Gazetteer Services Route Determination Services Web Processing Services (WPS)

examples of application services are geoportals (Bernard, Kanellopoulos, Annoni, & Smits, 2005), which in turn may integrate other client-side application services such as discovery services and map viewers. Examples of geoportals may be found at *http://www.geodata.gov and http://geoportal.jrc.it/*.

- *Registry services* (often called catalog services) are a special kind of service that offers end users a common mechanism to register, search, and access discoverable geospatial data and services.
- *Data services* are the basic geospatial services that serve geospatial data to application services. Examples of data services include the Web Feature Service (WFS), which filters and retrieves vector format representations of geospatial features and feature collections encoded in Geographic Markup Language (GML) (Cox, Daisay, Lake, Portele, & Whiteside, 2002), and the Web Coverage Service (WCS), which provides access to client-specific continuous coverage or image datasets.
- *Portrayal services* may be also considered a specialized data service that produce rendered data such as portrayed maps, perspective views of terrain, annotated images, and so on. Examples are the Web Map Service (WMS) that dynamically produces spatially referenced maps of client-specified criteria from one or more geographic datasets, returning the map views in well-known image or graphics formats.
- *Processing services* essentially transform geospatial data to produce new data or actionable information. Examples are the Web Coordinate Transformation Service (WCTS), which transforms the geographic coordinates of feature (map) or coverage (imagery) data from one coordinate reference system (CRS) to another; and the Gazetteer Service, which provides location geometries for specified geographic names. In order to help standardize access and binding to processing services, the OGC created the Web Processing Service (WPS) specification (Schut, 2007), which describes the interfaces needed in order to offer generic geospatial processing services over the Internet, as described in the following section.

Spatial data infrastructures (SDIs) were designed to share existing geospatial data (most held by the public sector) and make them widely accessible and available at the lowest possible cost, where and when they are needed (Granell, Gould, Manso, & Bernabé, 2008). An SDI can be thought of as a network of interoperable Web services to facilitate basic geospatial data (e.g., a digital topographic map) and customized information (e.g., a daily forest fire risk map) and services. Figure 2 summarizes the conceptual

Figure 2. The OWS Service Framework (adapted from Percival, 2003; Yang & Tao, 2006)



SDI architecture that may be interpreted as a traditional three-tier client-middleware-server model, where GIS applications (clients) seek geospatial data content (servers) that are discovered and then possibly transformed or processed by intermediary services (middleware) before results are presented back to the client tier. The presentation layer in Figure 2 includes the application services, whereas the middleware layer contains data services, registry services, portrayal services, and processing services.

Beyond the three-tier model, however, under the SOA perspective, the previous SDI architecture also may be interpreted using the Web services 'publish-find-bind' triangle model (Papazoglou, 2008), as shown in Figure 1. In this context, the OGC proposed the OWS Service Framework (OFS) as the common set of interfaces required for enterprise-wide interoperability within and beyond the GIS community (see Figure 2). Following this framework, geospatial data content (and service) offers are published to registry services, which are later queried to discover (find) the data or services, and finally the client application binds to (consumes or executes) them. In this sense, the adoption of a common geographical data model expressed in GML and standardized OGC specifications constitutes one ingredient to achieve geospatial data integration and interoperability in the wider sense (Díaz, Granell, & Gould, 2008a). The next section will focus on OGC specifications for geospatial (Web) processing services.

DISTRIBUTED GEOSPATIAL PROCESSING SERVICES

Although OGC has already proposed specifications under the processing services category (see Table 1), these are devoted primarily to performing specific and well-defined processing functions. A substantial leap ahead in the domain of processing services was the recently released OGC Web Processing Service (WPS) specification (Schut, 2007), which was designed to encapsulate generic geoprocessing operations over the Internet. The WPS specification allows any piece of geospatial processing code to be published and accessed as if it were a common OWS service (WMS, WFS, etc.). This section focuses on this new specification and describes some emerging open source frameworks that support the implementation of distributed geospatial processing services as defined by the OGC WPS.

OGC Web Processing Service Interface

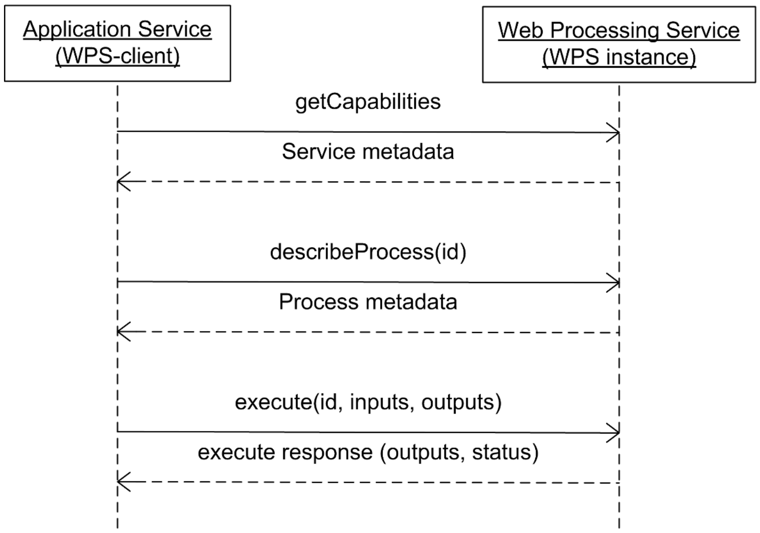
OGC WPS specification provides the service interface definitions to specify a wide range of geospatial processing tasks as geospatial Web services in order to distribute over the Internet many of the functionalities (computation, analysis, etc.) common in today's desktop GIS applications. Geospatial processing services can be considered as being similar to collections of operations in a software component library in the sense of preexisting components that deliver some concrete functionality. The main difference is that WPS can be accessed remotely and can be reused in many different scenarios. This can be achieved by creating accessible libraries of geospatial processing algorithms under the appearance of geospatial Web service chains (Alameh, 2003; Lemmens et al., 2006).

The OGC WPS provides access to calculations or models that operate on spatially referenced data. The data required by the service can be available locally or delivered across a network using data exchange standards such as GML or Geolinked Data Access Service (GDAS). The calculation can be as simple as subtracting one set of spatially referenced numbers from another (e.g., determining the difference in influenza cases between two different seasons) or as complicated as a global climate change model. While most OGC specifications and standards are devoted to geospatial data abstraction, access, and integration, the OGC WPS specification is focused on geospatial data processing of heterogeneous data sources. The main steps in this process are to identify the spatially referenced data required by the calculation, initiate the calculation, and manage the output from the calculation so that it can be accessed by the client. The OGC WPS specification is targeted at both vector and raster data processing.

The basic operational unit of the OGC WPS is the notion of process—a geospatial operation with inputs and outputs of a defined type. This means that a given WPS instance (a concrete WPS service running) may offer one or various operations (or processes) as normal Web services do. Figure 3 shows how a WPS client communicates with a WPS instance, issuing three types of requests. A request can be sent to the WPS instance via HTTP GET with parameters provided as Key-Value Pairs (KVP) or via HTTP POST, with parameters supplied in a XML document. These three types of requests are:

- *getCapabilities:* First, a WPS instance receives a KVP getCapabilities request (which is common for all OWS services) and simply responds with an XML document, containing metadata such as server provider, contact information, general description, and a list of contained geoprocessing operations (processes) offered by the queried WPS instance.
- *describeProcess:* A WPS client selects a process identifier from the getCapabilities response and performs a describeProcess request, either as a KVP or as an XML document. The WPS instance responds with an XML document containing needed information for the solicited process, such as input and output parameter

Figure 3. Synchronous interaction between a WPS-compliant client and a WPS service instance



names and types, so that the WPS client may later build the execute request.

- *Execute:* The WPS client eventually requests the execution of a geospatial operation, with all required input data by invoking the execute method as an XML document request. The WPS instance then runs the operation and returns the results, informing also of its status.

Implementations

This section summarizes some relevant and interesting open source frameworks that currently support one or both available versions (0.4 or 1.0) of the OGC WPS specification.

Cepický and Becchi (2007) introduce the *Python Web Processing Service* (PyWPS), an open source python framework that implements the OGC WPS specification version 0.4.0 (*http://pywps.wald.intevation.org/*). PyWPS includes native support for GRASS (Geographic Supported Analysis Support System, *http://grass.itc.it*) GIS, as well as with the R Project for Statistical Computing (*http://www.r-project.org/*). GRASS GIS is a well-known, powerful GIS tool for geospatial data management and analysis, image processing, graphics/maps production, geospatial modeling, and visualization, while the R Project is a free software environment developed for statistical computing. It is important to highlight that PyWPS allows developers to make native connections to both GRASS GIS and R Project commands, wrapping (or encapsulating) them as contained processes in a given WPS service. This capability fosters the proliferation of distributed geospatial processing services in new domains (environmental, hydrological, etc.) in which distributed geospatial processing services previously were not so easily implemented.

The *Tigris WPSint* implementation (*http://wpsint.tigris.org/*) is an open source Java plug-in for Spring—a Java framework for developing Web applications—to support the OGC WPS version 0.4. This implementation was initially developed by Peter Schut and colleagues during an OGC Interoperability Experiment in order to define initial interfaces and XML schemas for geospatial geoprocessing services, leading then to the first release of the OGC WPS specification in 2005. Contrary to PyWPS, the Tigris WPSint implementation has recently added support for SOAP and WSDL, key Web service components. This feature helps to converge SOA-based services and OGC-based services because both kinds of services may be combined to build heterogeneous service chains since both are described using the same service interface (WSDL).

The *52N Web Processing Service* (52N WPS) is an open source Java framework developed by the 52 North Open Source Initiative (*http://www.52north.org*) that enables the deployment of WPS services. It features a pluggable and extensible architecture for processes and data encodings based on the notions of repositories, which provide dynamic access to the embedded functionality of the WPS already registered in the framework (Foerster, 2006). The current release provides the first attempt to support both the GRASS GIS framework and the WSDL specification. In this sense, the 52N WPS follows the path chosen by PyWPS and Tigris WPSint to support GRASS GIS commands and WSDL/SOAP interfaces respectively, reinforcing the idea that both characteristics are crucial for a widespread use of distributed geospatial processing services in SOA contexts. The benefit of 52N WPS implementation is that it integrates both capabilities, although they are not (yet) as mature as in the previous two implementations.

The *Deegree* project (*http://www.deegree.org*) is an open source Java framework that implements the OGC WPS as well as traditional OGC services such as WMS, WFS, WCS, and so on. The benefit of Deegree is that it provides the most extensive implementation of OGC standards; however, unfortunately, their WPS implementation seems less mature when compared with the previous WPS implementations. Examples of WPS services using the Deegree project have been reported by Kiehle (2006).

FUTURE TRENDS

The OGC WPS services have been tested in different contexts (Friis-Christensen, Lutz, Ostländer, & Bernard, 2007; Foerster & Shäffer, 2007; Díaz et al., 2008b), illustrating that it is possible to combine several geospatial processing services for accessing, processing, and visualizing data within an SDI. However, many open issues remain regarding the structure and use of the OGC WPS specification itself (Michael & Ames 2007). Other technical and architectural design limitations that constraint the usability, flexibility, and scalability of applications based on distributed geospatial processing services also remain (Friis-Christensen et al., 2007).

One of the most essential problems in implementing distributed geospatial processing services is the overall service chain performance when distributed data sources are involved. This is the case when large processing tasks are performed over the network, because of network bandwidth, data transportation, and data validation. Historically a critical factor of distributed processing has been the network capability or network bandwidth. As GIS resources (inputs and outputs) are by nature large data files, the network bandwidth will always be a limiting factor for successfully distributed geospatial processing. Apart from the bandwidth factor, data transportation and validation (parsing of geospatial data used for the processes) may dramatically increase the response time to users as well. Friis-Christiensen et al. (2007) propose the use of asynchronous messaging to address time-consuming requests. In asynchronous messaging the WPS instance does not return immediately the process results, but rather it responds some time later in a different communication session. This means that the WPS client would not be waiting while the WPS instance is processing a request, but instead it would monitor the process and retrieve the results once the WPS instance has either finished or reported a failure; this essentially means processing results off-line.

Finally other open and challenging issues are enumerated that need further research:

- semantically enriching the descriptions of geospatial processing services by means of geo-ontologies and semantic descriptions that will help to clarify meanings when searching and combining geospatial processing services;
- creating alternative architecture designs and methodologies for chaining geospatial processing services, including in mobile computing contexts;
- creating a mechanism for improving discovery of geospatial processing services;
- using transactional processes;
- improving security; and
- introducing performance and novel techniques for overcoming data transportation issues.

CONCLUSION

The future scenario for geospatial Web services may not ever reach a wholly automated service chaining for a set of self-describing geospatial Web services; however in the near term, semi-automated solutions will emerge to assist users in solving geographical problems with remote services. The geospatial Web services listed in Table 1 mainly deal with the delivery of data instead of advanced processing which is performed online. More heterogeneous, complex geospatial processing services will need to be specified in order to distribute functionalities common in desktop GISs and frameworks such as GRASS GIS and Project R, and make them available over the Internet. The first steps towards distributed, advanced geospatial processing services online are outlined by the recently published OGC Web Processing Service (Schut, 2007), which provides interface specifications to enable geospatial Web services to support a wide range of geospatial processing operations, by creating accessible libraries of geospatial processing algorithms under the appearance of geospatial Web services.

Future research efforts in distributed geospatial processing services should involve new mechanisms for enhancing description and discovery of geospatial processing services, as well as new methodologies for improving composition of geospatial processing services in mobile contexts.

REFERENCES

Alameh, N. (2003). Chaining geographic information Web services. *IEEE Internet Computing, 7*(5), 22-29.

Bernard, L., Kanellopoulos, I., Annoni, A., & Smits, P. (2005). The European geoportal—one step towards the establishment of a European spatial data infrastructure. *Computers, Environment and Urban Systems, 29*(1), 15-31.

Cepický, J., & Becchi, L. (2007). Geospatial processing via Internet on remote servers—PyWPS. *OSGeo Journal, 1*(May).

Retrieved from http://www.osgeo.org/journal/volume1

Cox, S., Daisay, P., Lake, R., Portele, C., & Whiteside, A. (Eds.). (2002). *OpenGIS Geography Markup Language (GML) version 3.0.* Retrieved from http://www.opengeospatial.org/standards/gml

Díaz, L., Granell, C., & Gould, M. (2008a). Spatial data integration over the Web. In V.E. Ferraggine, J.H. Doorn, & L.C. Rivero (Eds.), *Encyclopedia of database technologies and applications* (2nd ed.). Hershey, PA: Information Science Reference.

Díaz, L., Granell, C., & Gould, M. (2008b). Case study: Geospatial processing services for Web-based hydrological applications. In J.T. Sample, K. Shaw, S. Tu, & M. Abdelguerfi (Eds.), *Geospatial services and applications for the Internet.* Berlin: Springer-Verlag.

Foerster, T. (2006). An open software framework for Web service-based geo-processing. *Proceedings of the Free and Open Source Software for Geospatial* (FOSS4G 2006), Lausanne, Switzerland.

Foerster, T., & Shäffer, B. (2007). A client for distributed geo-processing on the Web. *Proceedings of International Symposium on Web and Wireless Geographical Information Systems* (W2GIS 2007) (pp. 252-263), Cardiff, Wales. Berlin: Springer-Verlag (LNCS 4857).

Friis-Christensen, A., Lutz, M., Ostländer, N., & Bernard, L. (2007). Designing service architectures for distributed geoprocessing: Challenges and future directions. *Transaction in GIS, 11*(6), 799-818.

Granell, C., Gould, M., Manso, M.A., & Bernabé, M.A. (2008). Spatial data infrastructure. In H. Karimi (Eds.), *Handbook of research on geoinformatics.* Hershey, PA: Information Science Reference.

ISO 19119. (2005). *Geographic information services.* ISO Technical Committee 211 in Geographic Information/Geomatics.

Kiehle, C. (2006). Business logic for geoprocessing of distributed geodata. *Computers & Geosciences, 32*(10), 1746-1757.

Lemmens, R., Wytzisk, A., de By, R., Granell, C., Gould, M., & van Oosterom, P. (2006). Integrating semantic and syntactic descriptions to chain geographic services. *IEEE Internet Computing, 10*(5), 42-52.

Michael, C., & Ames, D.P. (2007). Evaluation of the OGC web processing service for use in a client-side GIS. *OSGeo Journal, 1.* Retrieved from http://www.osgeo.org/journal/volume1

Papazoglou, M.P. (2008). *Web services: Principles and technology.* Essex: Pearson Education.

Peng, Z.-R., & Tsou, M.-H. (2001). *Internet GIS: Distributed geographic information services for the Internet and wireless networks.* Hoboken, NJ: John Wiley & Sons.

Percivall, G. (Ed.). (2003). *OGC reference model, open geospatial consortium* (doc. no. 03-040). Retrieved from http://portal.opengeospatial.org/files/?artifact_id=3836

Vinoski, S. (1997). CORBA: Integrating diverse applications within distributed heterogeneous environments. *IEEE Communications Magazine, 45*(2), 46-55.

Schut, P. (Ed.). (2007). *OpenGIS Web processing service, version 1.0.0.* Retrieved from http://www.opengeospatial.org/ standards/wps/

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). Norwell, MA: Kluwer Academic.

Yang, C.P., & Tao, C.V. (2006). Distributed geospatial information. In S. Rana & J. Sharma (Eds.), *Frontiers of geographic information technology* (pp. 103-120). Berlin: Springer-Verlag.

KEY TERMS

Geography Markup Language (GML): An XML grammar defined by OGC to express geographical features. To help users and developers to structure and facilitate the creation of GML-based application, GML provides *GML profiles* that are XML schemas that extend the very GML specification in a modular fashion. A GML profile is a GML subset for a concrete context or application, but without the need for the full GML grammar, simplifying thus the adoption of GML and facilitating its rapid usage. Some common examples of GML profiles that have been published are *Point Profile,* for applications with point geometric data, and *GML Simple Features Profile,* supporting vector feature requests and responses, as in the case of a WFS.

Geospatial Processing Service: Similar to operations in a software library in the sense that these services are preexisting software components that deliver any geospatial processing functionality over the Internet.

ISO/TC211: ISO Technical Committee 211 in Geographic Information/Geomatics is in charge of establishing a set of standards for digital geographic information concerning objects or phenomena that are directly or indirectly associated with a location relative to the earth.

Open Geospatial Consortium (OGC): An international industry consortium participating in a consensus process to develop publicly available interface specifications. OGC members include government agencies, commercial companies, and university research groups.

Service: Functionality provided by a service provider through interfaces (paraphrased from ISO 19119).

Service Broker: Publishes service descriptions and is queried by the service requestor in order to discover suitable services that meet requestor needs.

Service Metadata: Metadata describing the operations and geographic information available at a particular instance of a service (paraphrased from ISO 19119).

Service Provider: Provides software applications as Web services, creating functional descriptions and making them available in public registries.

Service Requestor: Requires certain requirements and needs that are fulfilled by one or more Web services available over the Internet.