Computers & Geosciences 33 (2007) 649–665

COMPUTERS & GEOSCIENCES

www.elsevier.com/locate/cageo

Semantics-based automatic composition of geospatial Web service chains

Peng Yue[a,b], Liping Di[a,*], Wenli Yang[a], Genong Yu[a], Peisheng Zhao[a]

[a]Center for Spatial Information Science and Systems (CSISS), George Mason University,
6301 Ivy Lane, Suite 620, Greenbelt, MD 20770, USA
[b]State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing,
Wuhan University, 129 Luoyu Road, Wuhan 430079, China

Received 6 February 2006; received in revised form 11 September 2006; accepted 12 September 2006

Abstract

Recent developments in Web service technologies and the semantic Web have shown promise for automatic discovery, access, and use of Web services to quickly and efficiently solve particular application problems. One such application area is in the geospatial discipline, where Web services can significantly reduce the data volume and required computing resources at the end-user side. A key challenge in promoting widespread use of Web services in the geospatial applications is to automate the construction of a chain or process flow that involves multiple services and highly diversified and distributed data. This work presents an approach for automating geospatial Web service composition by employing geospatial semantics in the service-oriented architecture (SOA). It shows how ontology-based geospatial semantics are used in a prototype system for enabling the automatic discovery, access, and chaining of geospatial Web services. A case study of the chaining process for deriving a landslide susceptibility index illustrates the applicability of ontology-driven automatic Web service composition for geospatial applications.

Keywords: Geospatial Web service; Service-oriented architecture (SOA); Ontology; Semantic Web; Service composition; Service chaining

1. Introduction

More than two dozen spacecrafts are currently measuring the state of the earth system. These spacecrafts, together with countless air-, land-, and water-based monitoring systems, are generating vast volumes of geospatial data. For example, NASA's earth observing system (EOS) alone is generating about 3.5 TB of data each day. This unprecedented data-collecting capability brings considerable challenges to geospatial research and applications, one of which is how to derive high-level information and knowledge from the large volumes of data. Furthermore, the difficulty in deriving knowledge is linked to the complex nature of geospatial data and data archive systems, which are highly multidisciplinary, heterogeneous, and distributed.

One of the major approaches in tackling the challenges is to promote the use of Web services, a method that can significantly reduce the data volume, computing steps, and resources required

---

*Corresponding author. Tel.: +1 301 982 0795; fax: +1 301 345 5492.
E-mail addresses: geopyue@gmail.com (P. Yue), ldi@gmu.edu (L. Di).

0098-3004/$ - see front matter © 2006 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cageo.2006.09.003

at the end-user side (Di and McDonald, 1999). A Web service is a software system designed to support interoperable machine-to-machine interaction over a network.[1] It has a standard interface to enable the interoperation of different software systems, so that Web services developed by different organizations can be combined to fulfill users' requests. Geospatial problems usually involve large and heterogeneous data and multiple computation steps and service providers. Service composition, the process of creating the service chain though composing a collection of interoperable web services, is required. Automatic service composition, if successful, can be of great value to the geospatial user community because it can greatly broaden the uses of geospatial knowledge in social and economic activities and it can automatically provide answers to users' questions (Di, 2004).

The key to achieve automation relies mainly on solutions to three issues: (1) how to make geospatial Web services interoperable both syntactically and semantically; (2) how to automatically discover, based on the syntactic and semantic descriptions, the most appropriate data and services; and (3) assemble them to build the composite service (Di, 2005).

1.1. Interoperability of Web services

Interoperability is the capability to exchange information, execute programs, or transfer data among various functional units in a manner that requires the user to have little or no knowledge of the unique characteristics of those units (Percivall, 2002). There are two levels of interoperability[2]: syntactical interoperability and semantic interoperability (Percivall, 2002). The former requires that there is a technical connection, i.e. that the data can be transferred between Web services. It does not provide an interpretation for the content transferred in the connection. The latter assures that the contents of data and services are correctly understood when data/services are connected.

Syntactical interoperability of Web services is achieved mainly using two common Web service standards: Web service description language (WSDL)[3] and simple object access protocol (SOAP).[4] WSDL is used to describe a Web service in terms of its interfaces and SOAP formalizes the XML-based message transportation between Web services. In the geospatial community, the Open Geospatial Consortium (OGC) has defined a series of interface specifications for the interoperability of geospatial Web services, e.g. Web feature service (WFS) (Vretanos, 2005), Web map service (WMS) (de la Beaujardière, 2004), Web coverage service (WCS) (Evans, 2003), Web coordinate transformation service (WCTS) (Müller et al., 2004), and Web image classification service (WICS) (Yang and Whiteside, 2005). These specifications follow the principles for geospatial Web services defined in ISO 19119, and describe the structure of content transferred between Web services. For example, WCS defines the standard interface and message type for Web services providing coverage data, yet it does not formalize the conceptualization of content.

To achieve semantic interoperability, the conceptualization of content should be expressed formally and explicitly. This can be achieved by using ontologies. An ontology is a formal, explicit specification of a conceptualization that provides a common vocabulary for a knowledge domain and defines the meaning of the terms and the relations between them (Gruber, 1993). Ontologies are crucial to making the semantics of the exchanged content machine understandable. The Web ontology language (OWL),[5] recommended by W3C as the standard Web ontology language, is designed to represent semantics based on a flexible graph model composed of resource description framework (RDF)[6] triples. Furthermore, the Semantic Web services initiative (SWSI) introduces OWL-S[7] as the representative technology for describing the semantics of individual Web services. OWL-S can be used

---

[1] Web services architecture. W3C working group note 11 February 2004, W3C. http://www.w3.org/TR/ws-arch/.

[2] Some may argue the structural interoperability, e.g. mapping the elements in the output message structure of one service to the input message structure of the next dependable service (Section 3.5). This paper follows the definition of syntactical and semantic interoperability from the OGC abstract service architecture. It treats this kind of structure difference as the issue related to the semantic interoperability.

[3] WSDL 1.1. W3C note 15 March 2001. http://www.w3.org/TR/2001/NOTE-wsdl-20010315.

[4] SOAP Version 1.2, W3C working draft 17 December 2001. http://www.w3.org/TR/2001/WD-soap12-part0-20011217/.

[5] OWL Web ontology language reference, W3C. http://www.w3.org/TR/owl-ref.

[6] RDF: Concepts and abstract syntax. W3C recommendation 10 February 2004. http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/.

[7] OWL-based Web service ontology (OWL-S). http://www.daml.org/services/owl-s/1.1/.

to specify the semantics of the exchanged data (i.e. input/output), the functionality (through the reference to some service classification outside OWL-S), pre/postconditions, and other aspects of a Web service such as grounding information. These explicit specifications make the capabilities of individual Web services machine understandable so that automated Web service discovery is possible. Much research dealing with the problem of automatic service composition uses OWL-S as the vehicle for service description (Srivastava and Koehler, 2003; Sycara et al., 2003; Zhang, 2004). Most of it aims at application to e-business and the more general information technology world. The work reported in this paper explores Web service composition in the geospatial domain, through the introduction and design of OWL-based ontologies conveying semantic information on geospatial services and data. Although the primary concern is geospatial, the design philosophy and architecture are general enough to be applicable to the broader community of semantic Web services.

1.2. Integration of Web services

The service-oriented architecture (SOA) provides the basis for the integration of Web services. SOA is a way of reorganizing a portfolio of previously siloed software applications and support infrastructure into an interconnected set of services, each accessible through standard interfaces and messaging protocols (Papazoglou, 2003). There are three key actors in SOA: requester, provider, and broker. The requester is the user who requires the information services. The provider is the standards-based individual service. The broker is a metainformation repository (e.g., a registry, catalog, or clearinghouse). The interactions among these actors involve the operations of publishing, finding, and binding. Service composition introduces a new operation into SOA, chaining, which combines services into a dependent series to accomplish a larger task. SOA is the basis for automatic service composition, since service management functions such as registration, discovery, accessing, and execution are well positioned under this structure and these functions are the basic units in the whole automation process.

Syntactically interoperable Web services can be chained manually under SOA. While service compositions generated at the syntactic level can meet many e-business needs (Aissi et al., 2002), they often need more considerations in geospatial applications, especially in complicated modeling and decision-making processes such as natural hazard prevention and damage assessments, where the inherent complexity of the geospatial data needs to be identified. In this paper, we explore the semantic support in the different aspects of SOA. In particular, an automatic and intelligent composition architecture based on the "DataType" and "ServiceType" ontologies is designed and tested. A prototype system backed by NASA EOS data and a number of OGC-compliant services, with OWL/OWL-S ontology descriptions, is implemented.

2. A use case

To illustrate the proposed solution, we use the following example throughout the paper. Supposing a user asks the question: "What was the susceptibility of Dimond Canyon, California, United States to landslide on January 10, 2005?" One reasonable answer would be to return an image map containing the landslide susceptibility index value.

An OGC-compliant catalog service (see Section 3.3) can provide help in the search of such an image map with the conditions from the thematic (e.g. landslide susceptibility), spatial (e.g. Dimond Canyon, CA, United States), and temporal information (e.g. January 10, 2005). However, an image map of landslide susceptibility is a data product resulting from expert analysis. It is not always available and up-to-date for a given region and date. To assess the landslide susceptibility, the expert might have to collect terrain slope data, slope aspect data, land cover data, and normalized difference vegetation index (NDVI) data. Here, the same problem exists. For example, slope data must be computed from the digital elevation model (DEM) data that is available, while the production of land cover data involves an image classification process. It is possible for the expert to produce all these data products routinely and register them in the catalog. Yet it is obviously more flexible and intelligent to wrap each computation process as a building block; thus, not only can a high-level data product be produced on demand, but also the flexible composition of these building blocks is possible to satisfy different modeling requirements. Hence, only comparatively low-level data (e.g. DEM or Landsat enhanced thematic mapper (ETM) imagery) need to be updated routinely, which can greatly reduce the cost in data management and maintenance. A service chain can then be created to bind these services and data orderly to generate the landslide susceptibility product for answering this question. If we want to let the machine automate this process, the system should be able to identify, at each step in the service chaining process, semantically correct data and services without human interactions. By using ontologies to enrich the data and service description, the semantics of data and services are machine understandable and conceptually potential data and services can be discovered through the attached logical reasoning process. For example, if no services are available (considering the keyword match) that can produce an exact "NDVI" as requested by the landslide susceptibility service, through the ontologies the reasoning process can tell that a service which can generate ETM NDVI data is also applicable.

Tables 1 and 2 list the services and data that can be used to answer this question when introducing Web service as the vehicle for this kind of building blocks. In order to obtain the final answer, these services and data have to be discovered from the catalog and chained automatically. The next sections introduce geospatial semantics and how they are incorporated in a framework to enable the automation of this process. It has to be noted that this work, however, is designed to be general and thus is not restricted to only this example.

3. Geospatial semantics

Geospatial semantics are those that convey content information about geospatial data, entities, phenomena, functionalities, relationships, processes, services, etc. The scope of geospatial semantics can be extremely broad. A number of research projects have been started in this subject, e.g. SWEET (Raskin and Pan, 2005). We focus on defining data and service semantics that enable dynamic and automatic composition of geospatial Web service chains to achieve a complex geospatial goal that involves heterogeneous data and multiple services. In order to establish geospatial semantics, the semantics of Web service must first be understood. In the Web service domain, semantics can be classified into four types (Sheth, 2003): (1) data/information semantics, (2) functional/operational semantics, (3) execution semantics, and (4) quality of service (QoS) semantics.

Data semantics annotate the semantics of input and output data in a Web service operation. Functional semantics represent the semantics for a

Table 1
Services used in this example

Service	Description
Landslide susceptibility	The computational model for landslide susceptibility in this service takes into consideration the factors of terrain slope, terrain aspect, land cover types, and vegetation conditions (through the normalized difference vegetation index, or NDVI) by assigning each a weighting factor and then doing the map algebra computation
Slope	Computes the terrain slope from DEM data
Slope aspect	Generates the terrain aspect from DEM data
ETM NDVI	Calculates ETM NDVI from the near-infrared (NIR) and red bands of ETM images
WICS	Performs the image classification functions (supervised) that can generate the land cover types.
WCS	Provides the available geospatial data in the data archives

Table 2
Data used in this example

Data	Description
DEM	Terrain elevation data (Dimond Canyon on January 10, 2005[a])
Training image	Label image containing land cover types for the training function in the WICS (Dimond Canyon on January 10, 2005)
ETM image	Image to be classified as land cover types (Dimond Canyon on January 10, 2005)
NIR image	Near-infrared (NIR) band of ETM image for NDVI calculation (Dimond Canyon on January 10, 2005)
Red image	Red band of ETM image for NDVI calculation (Dimond Canyon on January 10, 2005)

[a] The temporal condition of available data might not be valid on that time. To illustrate the solution, the data are assumed to satisfy the temporal condition and registered in catalog with a long time period.

service function. Execution semantics specify the requirements of a service such as the preconditions and postconditions/effects. QoS semantics provide the quality criteria for service selection. For example, a service that calculates the terrain slope from DEM data may require the HDF-EOS data format as a precondition and DEM data as input. It generates the slope as output. The functional semantics for this service can be represented by using the slope entity class in an ontology called functional ontology, in which each concept/class represents a well-defined functionality (Cardoso and Sheth, 2005).

This research primarily focuses on automatic service composition based on geospatial data and functional semantics leaving the execution semantics and QoS semantics oriented composition to future work. The subsequent sections show how geospatial semantics schemas (geospatial ontologies for data and functionality of service) are designed and how they are incorporated into SOA for service integration.

3.1. Geospatial semantics schema

Based on the degree of generality, ontologies can be divided into three levels (Guarino, 1997): top-level ontologies, domain ontologies, and application ontologies. Top-level ontologies describe the general concepts independent of domain, for example, object or event. Domain ontologies describe the concepts in a generic domain such as the geospatial domain. Application ontologies are related to a specific domain or task that is intended for use in one application rather than across many applications. This paper concerns mainly the design of domain and application ontologies.

By conceptualizing a set of controlled, community-accepted domain vocabularies (Bermudez, 2004) such as GCMD,[8] domain ontologies provide a high-level representation of concepts in the geospatial domain and an organizational structure for classifying data and services. They are intended to be used across many geospatial applications, while application ontologies are instead designed to be used in a single application. A bridge between the application ontologies and the high-level domain ontologies needs to be built so that the knowledge discovery based on the relationship of entity classes can be undertaken in a large knowledge base instead of being limited to small applications. The UML style graph of Fig. 1[9] introduces some examples. For data semantics, “DataType” ontologies are developed including the GCMD “DataType” and the landslide susceptibility “DataType” ontologies. Landslide susceptibility “DataType” ontology contains the necessary entity classes in this application such as “Terrain_Elevation” and “Terrain_Slope”. When bridging the domain ontologies and application ontologies, the “subClassOf” axiom is added to signify that “Terrain_Slope” is a subcategory of “Topology” in the GCMD ontology, while the “equivalentClass”axiom is added to signify that “Terrain_Elevation” in the landslide susceptibility “DataType” ontology is equivalent to “Terrain_Elevation” in the GCMD ontology. For functional semantics, “ServiceType” ontologies are developed, including the GCMD “ServiceType” ontology and the landslide susceptibility “ServiceType” ontology.

In addition to the “DataType” and “ServiceType” ontologies, we have also included the “Association” ontologies to describe the relationships between services and data (Fig. 1). The introduction of an association ontology can significantly speed up the reasoning process because it reduces the search space through the association relationship expressed in the ontology. As illustrated in Fig. 1, the association ontology indicates that a “Terrain_Slope” data type is associated to an “Image_Processing” service type. The searching process for services can then start primarily with those services under the “Image_Processing” service type. The instances of the “GeoDTSTAssociation” class can be defined in a very flexible way, depending on the application case. A direct association can be defined if a service type is tightly coupled with a data type (in this situation, it is similar to the service output description in the OWL-S) (e.g. a “GeoDTSTAssociation” instance associating a FGDCVegetationClassificationData data type with a FGDCVegetationClassificationService service type), while a relaxed association can be described when a data type and a service type are loosely related (e.g. a “GeoDTSTAssociation” instance associating a FGDCVegetationClassificationData data type with a vegetationClassificationService service type). A direct association can point

---

[8] Global change master directory (GCMD). http://gcmd.nasa.gov/Resources/valids/keyword_list.html.

[9] In Fig. 1, “geodatatype” represents the target namespace of the “DataType”, and “geoservicetype” represents the target namespace of the “ServiceType” ontology.

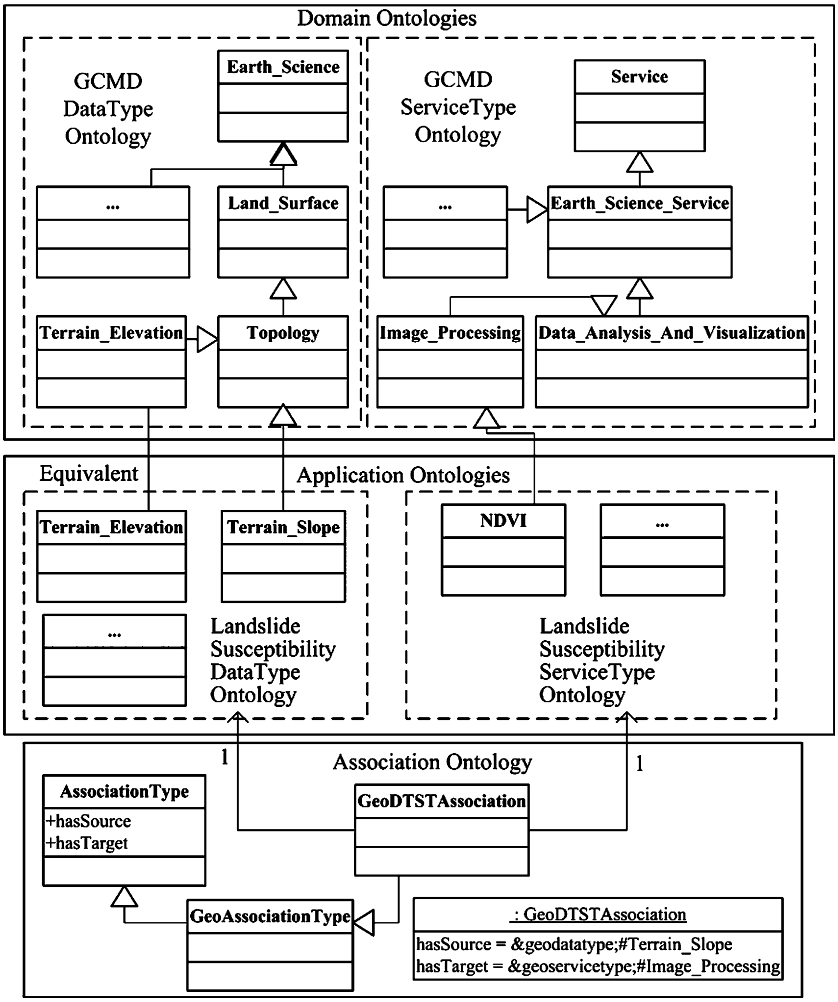


Fig. 1. Geospatial semantics schema.

to a specific service very quickly, but at the same time if the service under that service type is not available, the search process will terminate. A relaxed association can provide more possible available services (e.g. vegetationClassificationService includes multiple vegetation classification services). Thus, a further selection based on an input/output match can be performed on these available services. Furthermore, hierarchical associations among ontology concepts can be defined by extending the "Association" ontology (Fig. 1). It should be noted that, although useful, the association ontology is not a required component of our geospatial semantics, thus making the reasoning and composing engine more widely applicable and adaptable to other applications.

The entity classes in the "DataType" and "ServiceType" ontologies describe which entities can possibly exist in the geospatial domain, which in turn are used to represent the data and functional semantics in the geospatial Web service (e.g. input/output parameter type and service classification in OWL-S, see examples in Fig. 2). In addition, it provides the RDF structure (see Section 3.5) for the XSLT transformation in the service grounding of OWL-S. In these aspects, they can be treated as the conceptual schema for semantic description of geospatial Web services. We call them "geospatial semantics schema".

3.2. Geospatial semantics for providers

In SOA, service providers supply services over the Internet. As mentioned before, OWL-S is used to describe the semantics of geospatial Web services.

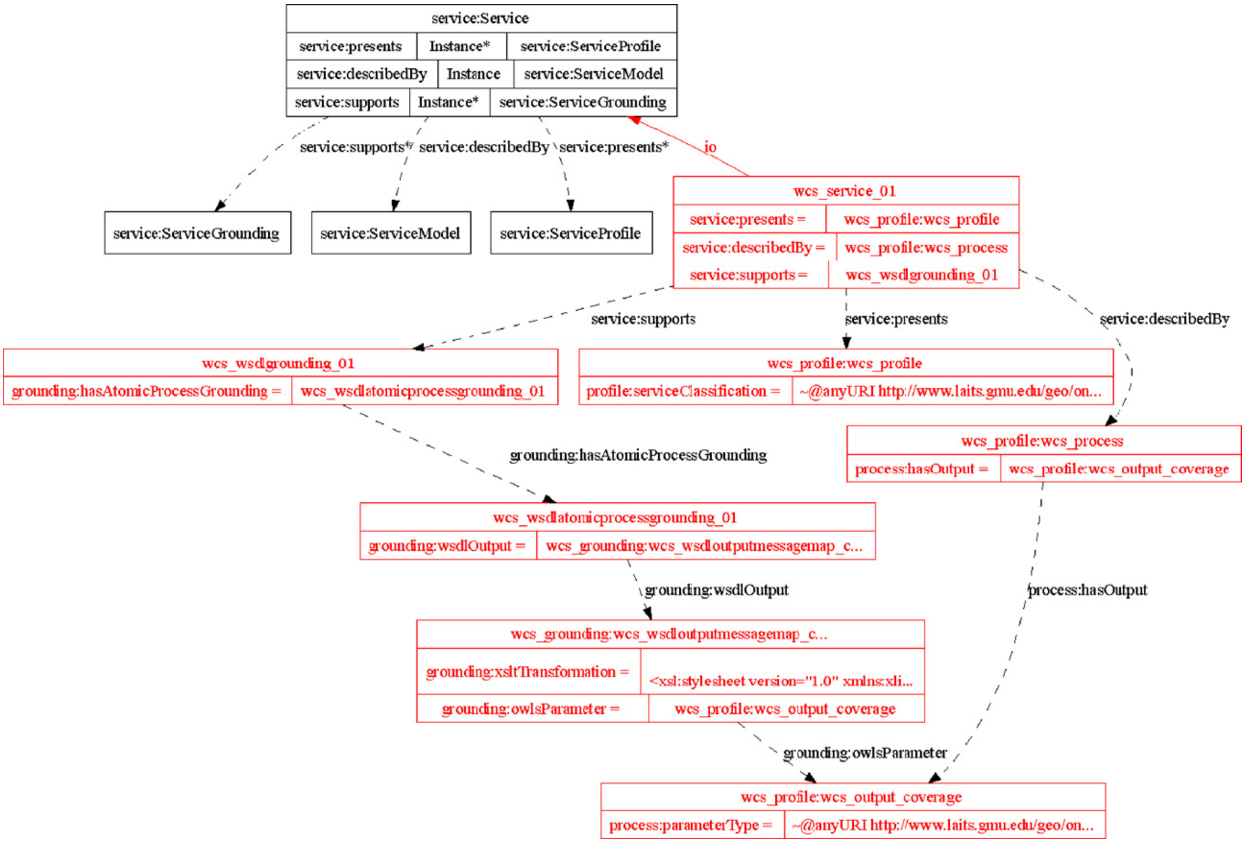


Fig. 2. OWL-S structure in the OntoViz.

A graph using OntoViz[10] (Fig. 2) illustrates how to describe a WCS using OWL-S.

OWL-S is structured in three main parts: (1) service profile: what a service does (advertisement), e.g. "WCS" as a "ServiceType" and "Coverage" as an output "DataType" in Fig. 2, (2) service model: how a service works (detailed description), e.g. a series of input parameters which are identified in the service model, and (3) service grounding: how to assess a service (execution), e.g. the output "DataType" of WCS is grounded to the output message of the GetCoverage operation defined in the WCS WSDL using an XSLT transformation. The service profile and service model concern the semantic description of the Web service using the geospatial semantics schema. The service grounding describes the relation of the semantic description to the syntactic description of a service.

One of the major efforts in the service grounding is to focus on the specification of the XSLT transformation between service messages and OWL-S parameters, since the ontology entity's RDF structure is not always consistent with the grounding message structure[11] of individual services. Two types of elements in the message structure should be differentiated in the grounding description: the elements whose values are passed along in service chains and those whose values are not passed along. Table 3 shows some examples. The "service" element in the WICS[12] GetClassification message does not get a value from its precedent services. Therefore, to enable automation, we set the grounding information for the "service" element of WICS with hardcoded text "WICS", while the "sourceURL" and "sourceFormat" elements can get values at runtime from the RDF structure of the "DataType" output in the precedent service WCS.

---

[10] http://protege.cim3.net/cgi-bin/wiki.pl?OntoViz.

[11] Most geospatial Web services provide access via HTTP GET, HTTP POST, and SOAP which can be described through WSDL interface. Thus we mainly discuss WSDL grounding of OWL-S in this paper.

[12] WICS version 0.0.20.

Table 3
Some examples on service grounding

	Grounding information

Type1 (values got from former service’s output, e.g. WCS OWL-S output message map)

```
<grounding:WsdlOutputMessageMap rdf:ID = “wcs_wsdloutputmessagemap_coverage”>
  <grounding:owlsParameter rdf:resource = “&wcs_profile;#wcs_output_coverage”/>
  <grounding:xsltTransformationString><![CDATA[
    <xsl:stylesheet version = “1.0” xmlns:xlink = “http://www.w3.org/1999/xlink”
xmlns:wcs = “http://www.opengis.net/wcs”
xmlns:xsl = “http://www.w3.org/1999/XSL/Transform”>
<xsl:template match = “//wcs:Coverage/wcs:CoverageRegion/wcs:CoverageData”>
<xsl:variable name = “X1” select = “@xlink:href”/>
<xsl:variable name = “X2” select = “wcs:Format”/>
<rdf:RDF xmlns:rdf = “http://www.w3.org/1999/02/22-df-syntax-ns#”
xmlns:mediator = “http://www.laits.gmu.edu/geo/ontology/domain/v2/mediator.owl#”
xmlns:iso19115 = “http://www.laits.gmu.edu/geo/ontology/domain/iso/v2/iso19115.owl#”
xmlns:geodatatype = “http://www.laits.gmu.edu/geo/ontology/domain/GeoDataType.owl#”>
  <geodatatype:Coverage>
  <mediator:hasMD_Metadata><iso19115:MD_Metadata><iso19115:distributionInfo>
    <iso19115:MD_Distribution>
      <iso19115:transferOptions>
        <iso19115:MD_DigitalTransferOptions> <iso19115:onLine>
          <iso19115:CI_OnlineResource> <iso19115:linkage>
            <xsl:value-of select = “$X1”/>
              </iso19115:linkage></iso19115:CI_OnlineResource>
              </iso19115:onLine></iso19115:MD_DigitalTransferOptions>
            </iso19115:transferOptions>
            <iso19115:distributionFormat>
              <iso19115:MD_Format><iso19115:name_MD_Format>
                <xsl:value-of select = “$X2”/>
              </iso19115:name_MD_Format></iso19115:MD_Format>
            </iso19115:distributionFormat>
          </iso19115:MD_Distribution>
  </iso19115:distributionInfo></iso19115:MD_Metadata></mediator:has MD_Metadata>
</geodatatype:Coverage></rdf:RDF></xsl:template></xsl:stylesheet>
  ]]></grounding:xsltTransformationString>
</grounding:WsdlOutputMessageMap>
```

Type 2 (values obtained from other ways such as hardcoded, e.g. one of WICS OWL-S input message map)

```
<grounding:wsdlInput>
<grounding:WsdlInputMessageMap rdf:ID = “wics_mindis_train_wsdlinputmessagemap1”>
<grounding:owlsParameter rdf:resource = “#wics_mindis_train_input_service”/>
<grounding:wsdlMessagePart
rdf:datatype = “&xsd;#anyURI”>&wics_wsdl;#service</grounding:wsdlMessagePart>
<grounding:xsltTransformationString><![CDATA[
<xsl:stylesheet version = “1.0” xmlns:xsl = “http://www.w3.org/1999/XSL/Transform”
xmlns:rdf = “http://www.w3.org/1999/02/22rdf-syntax-ns#”
xmlns:geodatatype = “http://www.laits.gmu.edu/geo/ontology/domain/GeoDataType.owl#”
xmlns = “http://www.opengis.net/wics”>
      <xsl:template match = “/”>
      <xsl:text>WICS</xsl:text>
      </xsl:template>
      </xsl:stylesheet>
  ]]></grounding:xsltTransformationString>
</grounding:WsdlInputMessageMap>
</grounding:wsdlInput>
```

In the past several years, OGC has made significant progresses on the standardization of geospatial Web services. Since our geospatial models include both OGC-compliant and non-OGC-compliant Web services, we develop two groups of OWL-S descriptions for the two categories of geospatial Web services. The OWL-S descriptions for OGC-compliant Web services focus on the semantic representation of the standard interfaces and messages. It is possible to define some common OWL-S grounding representation for all OGC service instances under the same standard interface and message with the premise of the same semantics. For example, different WCS service instances can share the common XSLT transformation information (example in Table 3) in service grounding.

Although the OGC service architecture abstract specification has listed a series of geographic services that could be standardized (Percivall, 2002), standard interface protocols are currently defined for only a very limited number of geographic services. A large number of geospatial software tools, most of which do not have standard interface protocols, are available either as freeware or commercial products. These tools can be developed into Web services with non-OGC-compliant interfaces. Under these circumstances, OWL-S descriptions for these services need to be developed individually, based either on a specific service instance (e.g. slope service) or on a small aggregation of service instances from a certain software package (e.g. GRASS[13]). Hence the message mappings in the service grounding for non-OGC-compliant services need to be described case by case.

3.3. *Geospatial semantics in brokers*

The broker contains information about information (metainformation) available over the Internet or in the holdings of digital libraries but not the information itself. The broker plays an important role in helping requesters to find the right services. Geospatial Web services are cataloged in a registry/broker with their properties and capabilities.

Currently, there are two prominent general models for registry services: the electronic business registry information model[14] (ebRIM) and the universal discovery description and integration[15] (UDDI) model. For the geospatial community, ebRIM is more general and extensible because it provides comprehensive facilities, based on the ISO 11179 set of standards, to manage metadata. OGC has developed and recommended an ebRIM profile for a Web-based geospatial catalog service, called the catalog service for the Web (CSW) (Martell, 2004). This profile introduces an ebRIM-based catalog information model for publication and discovery of geospatial information. The metadata for both geospatial Web services and geospatial data are registered in a CSW server.

There are some disadvantages in the current CSW, especially its support only to direct match of keywords without considering the underlying semantics when searching for data and services. It is desirable to enable a geospatial semantic search capability in a CSW. A few recent studies have been reported regarding mapping OWL elements to ebRIM elements (Dogac et al., 2004; Wei et al., 2005a). The basic idea is to use class, slot, and association elements in ebRIM to record corresponding OWL classes, properties, and related axioms such as subClassOf. We have designed the registration of our semantic schema, OWL-S descriptions, and data semantics (the "DataType" for the data) in CSW and implemented a semantic searching capability based on the registered semantic information (Yue et al., 2006). Currently, three types of semantic matching are supported: EXACT, SUBSUME, and RELAXED. Let OntR denote the requested concept and OntP denote the provider concept, the three matching conditions can be expressed as the following with the decreasing priority order:

EXACT	OntR equivalent to OntP
SUBSUME	OntP is a subClassOf OntR
RELAXED	OntR is a subClassOf OntP.

3.4. *Geospatial semantics for requesters*

A requester represents the consumer or user of Web services who needs geospatial information. A user may request a service to generate a data product, or may request a data product without knowing the specific service(s) needed to generate the product. The latter case is convenient to the general geospatial users. In our design, a user request is expressed by a concept in the "DataType" ontology, which represents the content or theme of the requested product. In addition to the "DataType", a geospatial query is often associated with other conditions, especially temporal and spatial constraints. Therefore, a complete query consists of at least three major elements, a "DataType" concept representing the content of the query, a

---

[13] http://grass.itc.it/.

[14] OASIS/ebXML registry information model v2.5. http://www.oasis-open.org/committees/regrep/documents/2.5/specs/ebrim-2.5.pdf.

[15] The UDDI technical white paper. http://uddi.org/pubs/uddi-tech-wp.pdf.

temporal domain, and a spatial domain. The following is an example of such a request in XML generated from the question in Section 2. This XML specifies the temporal/spatial ranges during/among which the information is requested. The ontology element of the XML specifies the type of information (i.e. "DataType").

```
<TimeRange>
    <Start>2005-01-10T00:00:00Z</Start>
    <End>2005-01-10T23:59:59Z</End>
</TimeRange>
<PlaceBoundingBox crs = "EPSG:4326">
    <WestBoundingLongitude>-122.262908</
WestBoundingLongitude>
    <SouthBoundingLatitude>37.597494</
SouthBoundingLatitude>
    <EastBoundingLongitude>-122.005009</
EastBoundingLongitude>
    <NorthBoundingLatitude>37.875999</
NorthBoundingLatitude>
</PlaceBoundingBox>
<Ontology></Ontology>
```

3.5. Geospatial semantics in service chain

There are many XML-based service composition languages such as the business process execution language for Web Services[16] (BPEL4WS), the Web services flow language[17] (WSFL) and the web service choreography interface[18] (WSCI). These languages rely on the XML and XML schema description of individual Web services for constructing the service chain. Certain schema-match mechanisms are required for enabling the chaining of Web services with heterogeneous interfaces and messages. For example, in order to chain a WCS service that provides DEM data and a slope service, a non-OGC-compliant service defined by the service provider that generates slope data from DEM (Fig. 3), we need first to extract the data URL and data format from the "Coverage" message structure defined in the OGC WCS schema[19], and then transfer them to the "souceURL" and "sourceFormat" parts of the DEM2SlopeRequest message in the slope service. Through the input/output XSLT transformation defined in the service grounding of OWL-S, this value-transfer process can be performed automatically at runtime.

When two services are chained, there must be a mapping between the message schemas of the services. One approach is to define direct schema mapping among all available services. In a Web environment where n services are available, the maximum possible number of such mappings is $C(n,2)$. For standards-compliant services, the mappings can be defined at the service-type level rather than at the service instance level, which reduces the number $C(n,2)$ to $C(m,2)$, where m, representing the number of service types to which the n service instances belong, is usually much smaller than n. For services not compliant with standards and thus without standard interface schemas, the number of direct schema mappings between each pairs of chainable services can be much larger. With the introduction of geospatial ontology, the mapping number can be reduced from $C(n,2)$ to n because messaging mappings are indirectly embodied in the mapping of the service message schema structure to a mediated RDF structure. The mediated RDF structure is defined by following the ISO 19115 standard, which provides a well-defined metadata structure. A light-weight RDF structure for all "DataType" entity classes, which acts as a relay structure to convey the element value of WSDL messages, is illustrated in Fig. 4. Fig. 4 shows that data URL and file format are identified in the "linkage" and "name_MD_Format," respectively.

van der Aalst (2003) compared several common service composition languages from the aspect of control flow. Twenty flow control constructs, such as sequence, parallel split, and choice, were identified as the considerations most often required when designing a service composition language. OWL-S provides a "Composite Process" ontology with control constructs for these pattern definitions. Composite processes are processes decomposable into other (non-composite or composite) processes. The decomposition can be specified by using control constructs. Since most control construct definitions originate from the service composition languages, it is possible for business processes defined in any of the service composition languages to map to the "Composite Process" ontology, thus achieving interoperability between service composition languages.

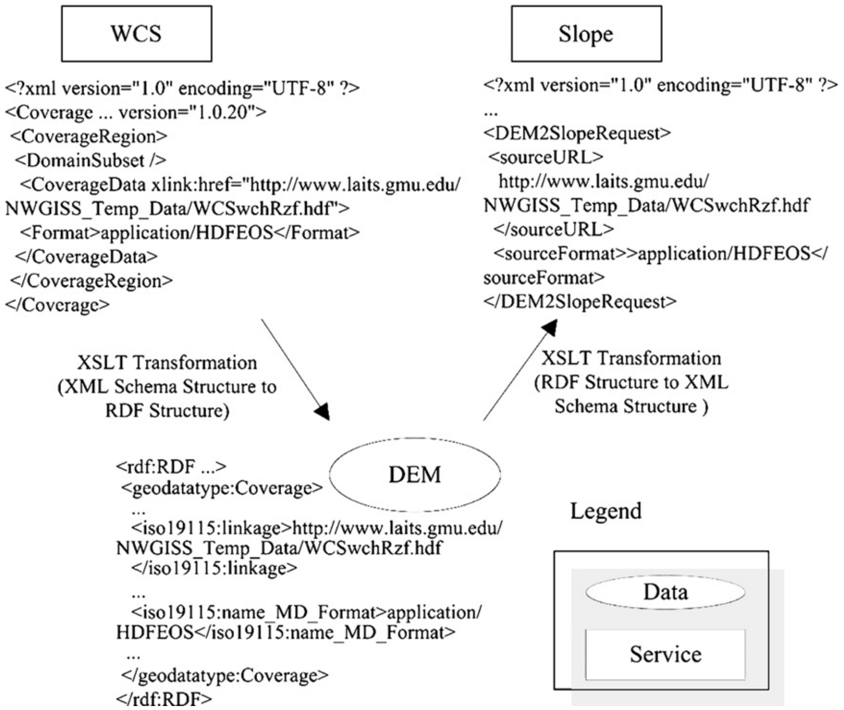
---

[16] Business process execution language for Web services (BPEL4WS). http://www-128.ibm.com/developerworks/library/specification/ws-bpel/.

[17] WSFL 1.0. http://www.ibm.com/software/solutions/webservices/pdf/WSFL.pdf.

[18] WSCI 1.0. http://www.w3.org/TR/wsci/.

[19] WCS version 1.0.20.



```
<?xml version="1.0" encoding="UTF-8" ?>
<Coverage ... version="1.0.20">
 <CoverageRegion>
  <DomainSubset />
   <CoverageData xlink:href="http://www.laits.gmu.edu/
NWGISS_Temp_Data/WCSwchRzf.hdf">
   <Format>application/HDFEOS</Format>
  </CoverageData>
 </CoverageRegion>
</Coverage>
```

```
<?xml version="1.0" encoding="UTF-8" ?>
...
<DEM2SlopeRequest>
 <sourceURL>
   http://www.laits.gmu.edu/
NWGISS_Temp_Data/WCSwchRzf.hdf
  </sourceURL>
  <sourceFormat>>application/HDFEOS</
sourceFormat>
</DEM2SlopeRequest>
```

```
<rdf:RDF ...>
 <geodatatype:Coverage>
  ...
  <iso19115:linkage>http://www.laits.gmu.edu/
NWGISS_Temp_Data/WCSwchRzf.hdf
  </iso19115:linkage>
  ...
  <iso19115:name_MD_Format>application/
HDFEOS</iso19115:name_MD_Format>
  ...
 </geodatatype:Coverage>
</rdf:RDF>
```

Fig. 3. Data flow in the service chain.

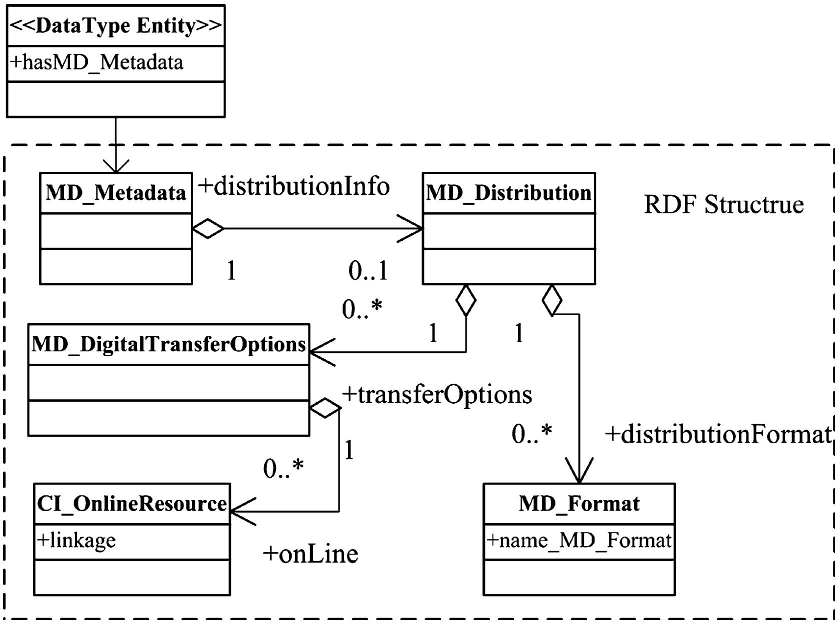


Fig. 4. A light-weight RDF structure for Data Type entity.

Also, a composite process in OWL-S resulting from service composition can be converted to any of the service composition languages to enable execution in the existing engine of these languages.

4. "DataType" driven automatic composition of service chains

Currently, our reasoning rules are primarily based on class hierarchical relationships defined in the service and data ontologies. According to semantic match priority discussed in Section 3.3, the preference order in a matching search for data and services is, in decreasing order, EXACT to SUBSUME to RELAXED.

An UML sequence diagram in Fig. 5 illustrates a simplified process. The following describes a more detailed process with an "Association" strategy. The semantic match options for the "DataType" and "ServiceType" can be set, respectively, in the *composer*.

(1) *User/requester* submits the XML query described in Section 3.4 to the *Composer* component.
(2) The *composer* constructs the CSW data query based on the input XML and "DataType" match option.
If the match option is "EXACT", the data with exact-matched "DataType"s are searched in CSW. If the match option is "SUBSUME", the data with exact-matched "DataType"s are searched in CSW first. If such data is not available, then the data with subsume-matched "DataType"s are searched in CSW. If the match option is "RELAXED", the data with exact-matched "DataType"s are searched in CSW first. If such data is not available, then the data with subsume-matched "DataType"s are searched. If the data is still not available, the data with relaxed-matched "DataType"s are searched in CSW finally. This match strategy has two advantages: (a) high precision—those data with higher match degree are always got firstly; and (b) efficiency—we obtain the matched "DataType" collection through the one-time reasoning and then perform the keyword match successively. This is more efficient than the orderly match between the requested "DataType" and each "DataType" of available data, because in the latter situation the reasoning process will repeat many times which will take lots of time in a large knowledge base.
(3) If matched data are not found in CSW, the *composer* searches for associated "ServiceType"s through the predefined

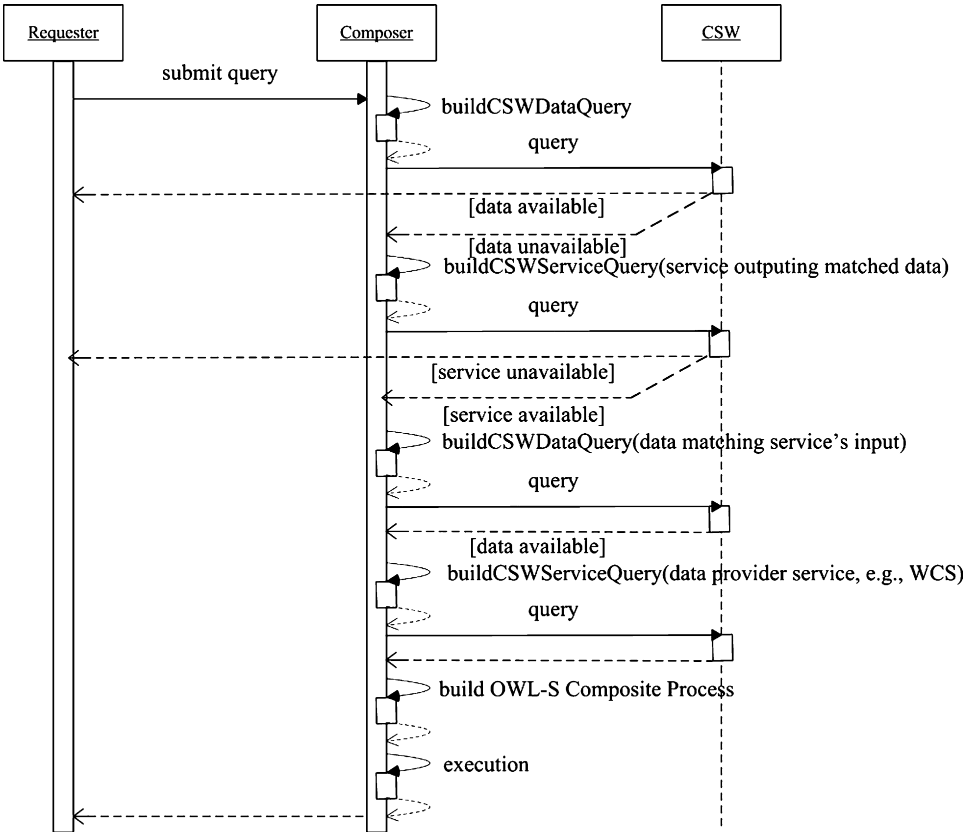


Fig. 5. "Data Type" driven automatic service composition.

“GeoDTSTAssociation” instances in the association ontology.
(4) The *composer* constructs the CSW service query based on the “ServiceType” match option and the “ServiceType” collection resulting from Step (3). The same match strategy in Step (2) is adopted here also.
(5) The *composer* checks the available services according to the matching between the service output “DataType”s and the requested “DataType”. If a matched service is found, the *composer* builds the CSW data query according to the input “DataType” of the selected service with those spatial and temporal constraints. This process continues recursively until all input data are available for the service chain. If finally some input data is not available, neither in the archives nor provided by the services, the chaining process will go back to an upper level, find another matched service, and repeat the above process again.
(6) When all binding data and services are available finally, the *composer* converts them into an OWL-S composite process, and executes them to delivery the product to the requester.

5. Prototype implementation and result analysis

5.1. Implementation

A prototypical system has been developed based on the concepts discussed in the above sections. The system is based on SOA with underlying geospatial services and knowledge base. The *composer* in Fig. 5 connects the three partners in SOA and plays the following three roles in the automatic composition of service chains:

(1) *Knowledge base*: Includes the geospatial domain ontology and service ontologies. An inference engine is attached for reasoning in semantic matching.
(2) *Chaining*: Composes service chains based on the knowledge base. The service chain is produced using the built-in OWL-S process model generator to tell the chain execution component about the logic flow in the service chain. The intermediate results can also be advertised as new Web services to the broker so that they can be reused.
(3) *Chain execution*: Performs the grounding of the logic flow in the service chain with individual Web services and executes those Web services. The service chain can also be converted to a service composition language representation and sent to the corresponding engine for execution.

OWL is used as the language for the geospatial semantics schema representation. The OWL-S API[20] is used for OWL-S parsing and grounding execution. The Jena Transitive Reasoner[21] is selected for reasoning based on our application knowledge base. The geospatial semantics schema and data semantics are registered in the grid-enabled CSW (Wei et al., 2005b) and can be queried through the CSW interface. The OWL-S manager, the *composer* in Fig. 5, works as a Web application with four types of primary functions:

(1) The OWL-S files management functions:
    (a) Set Schema (geospatial semantics schema): Provides the knowledge base for the reasoner.
    (b) Add Association: Adds associations between the “DataType” and “ServiceType”.
    (c) OWL-S Deploy: Loads a service ontology (i.e. an OWL-S) to the knowledge base.
    (d) OWL-S UnDeploy: Unloads a service ontology from the knowledge base.
    (e) Get Capabilities: Gets the service ontology repository in the knowledge base.
(2) The semantic matching function: Query matched data and services.
(3) The service chaining function: Perform automatic service chain composition based on the “DataType” driven process described in Section 4.
(4) The chain execution function: Execute the OWL-S composite process resulting from the chaining process.

OWL-S’s primary goal is service composition. Considering the execution of a service chain, a number of limitations in OWL-S have been identified, such as fault/error handling and event handling. These features are well defined in service composition languages such as BPEL4WS. We have developed and implemented a preliminary OWL-S to BPEL conversion tool, which also works as a Web application. The conversion results can be sent to a BPEL engine for execution.

---

[20] http://www.mindswap.org/2004/owl-s/api/.
[21] http://jena.sourceforge.net/inference/index.html.

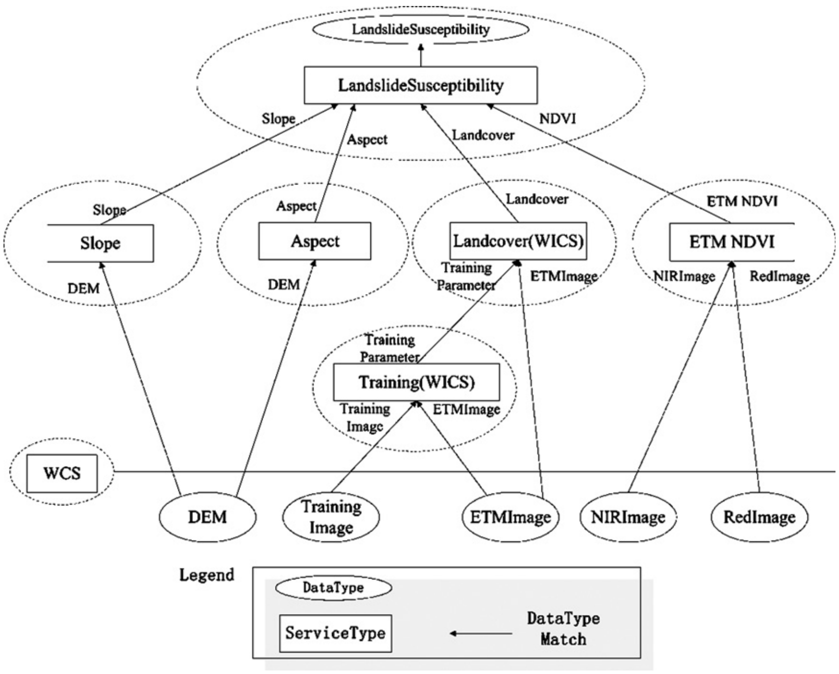


Fig. 6. Landslide susceptibility scenario.

An online demo of the implementation is available at http://www.laits.gmu.edu/geo/nga/index.html.

5.2. Result analysis

To test the effectiveness of this approach, the OWL-S manager has been applied to the problem described in Section 2. Fig. 6 shows the services involved and the data flows in this problem. Both OGC-compliant and non-OGC-compliant services are involved in the final service chain created by the automatic service composition process. In this case, the EXACT match cannot produce the landslide susceptibility data automatically because the ETM NDVI service's output ETM NDVI is not exactly the same as the NDVI input required by the landslide susceptibility computation service. Thus a SUBSUME match is required to achieve this goal. If users select the RELAXED match option, the goal still can be achieved since the match process considers the EXACT and SUBSUME matches first. The service chain in this use case can be automatically and dynamically generated whenever the CSW service is available and the data registration can be searched using the CSW service. The composite process can also be registered in the CSW as a virtual data product so that the composition process need not be repeated when a new request for the same data product is submitted. In addition to landslide susceptibility data, slope data, slope aspect data, land cover data, or ETM NDVI data can also be created on demand whenever the involved services and data are available from the Web.

6. Related work

In recent years, there have been several studies in the area of automatic service composition (Srivastava and Koehler, 2003; Sirin et al., 2003; Klusch et al., 2005). Sirin et al. (2003) present a semi-automatic method for Web service composition. OWL and OWL-S provide the semantics needed for service filtering and composition. The composition is based primarily on a match between two services such that the output of the first service provides the input of the second service. Both the functionalities and the non-functional attributes of the services are considered. Functionalities are identified using the hierarchy of service profile ontology in OWL-S. The match is conducted by attempting to match input/output types in the service profile ontology. When a match is found, non-functional attributes such as location can be used to filter service instances with that service profile. Our system uses a similar input/output-type matching approach. One of the most important characteristics of the geospatial domain is that an application often includes multiple modeling or processing steps involving large and heterogeneous data volumes, such as the landslide example presented in this paper. Our system must therefore be able to construct and execute a service composition involving complex constraints on input/output at each composite node (i.e. data or service), such as data format, map projection, spatial and temporal resolution. Such constraints are primarily dealt with by extending the non-functional attributes and we have designed the framework of such extensions. Spatial bounding box and temporal duration constraints have been implemented in our current prototypical system to guide data and service discovery at each composition node. Additional constraints can be added based on the framework.

Some efforts on geospatial Web service composition have been reported. One example is the geosciences network (GEON) (Jaeger et al., 2005). Geospatial Web services including data (GML representation) provider services and customized services with vector data processing functionalities are sampled to compose a workflow manually in the KEPLER system (Ludäscher et al., 2005). The KEPLER system provides a framework for workflow support in the scientific disciplines. The major feature of the KEPLER system is that it provides high-level workflow design while at the same time hiding the underlying complexity of technologies from the user as much as possible. Both Web service technologies and grid technologies are wrapped as extensions in the system. For example, individual workflow components, such as data movement, database querying, job scheduling, and remote execution, are abstracted into a set of generic, reusable tasks in the grid environment (Altintas et al., 2004). Thus, combining a knowledge representation technique (e.g. OWL and OWL-S), with the lower level, generic, common scientific workflow tasks in the KEPLER system, is a worthwhile technique for minimizing or eliminating human intervention in the generation and instantiation of workflow. OWL is introduced into SEEK, a similar and major contributor to KEPLER, to enable automatic structural data transformation in the data flow among services. This transformation is based on ontology and registration mapping of input and output structural types to their corresponding semantic types (Bowers and Ludäscher, 2004). The information conveyed in the registration mapping is the same as the XSLT transformation in the service grounding of OWL-S. Our framework is more open and interoperable than that method, because it formalizes semantic types with the mediated RDF structure and represents registration mapping explicitly in the grounding information of OWL-S. Using semantically augmented metadata to annotate data and service is important to automatic service and data discovery (Lutz et al., 2003; Lutz and Klien, 2006; Klien et al., 2006). Ontologies, related in both simple taxonomic and non-taxonomic ways, are employed using subsumption reasoning to improve service discovery and the recall and resolution of data. Our data and service ontologies are designed based on semantically meaningful taxonomy. The semantics are mainly reflected at class level. We will include more rich semantic support on data and service discovery.

There are a significant number of literatures addressing the problem of automatic service composition in the semantic Web areas through artificial intelligence (AI) planning. Many efforts on automating the Web service composition problem using AI planning have been reported (McIlraith and Son, 2002; Wu et al., 2003; Klusch et al., 2005). OWL-S is adopted for modeling Web services in these methods. Our current work provides a framework for automatic composition of geospatial Web services but does not include AI planning. We will introduce it in the next phase of development to enhance the capabilities of automatic service composition.

7. Conclusions and future work

This paper presents an approach to the automatic composition of geospatial Web services based on the geospatial semantics and SOA. We have explored the semantics for both geospatial Web services and geospatial information contents and formalized the semantics as geospatial ontologies. We have developed three types of ontologies in the geospatial domain, “DataType”, “ServiceType”, and “Association”, and used them as the semantic schema in SOA. Both OGC standards-compliant and non-standard Web services are used in semantic analyses, ontology design, and composition construction. A preliminary prototypical system involving semantic incorporation and coordination among the three major components of SOA has been implemented. An OWL-S engine, based on the

OWL-S API and providing basic functionalities for OWL-S management, has also been developed.

This work demonstrates that ontologies, both those for data and those for services, are useful for conveying geospatial semantics and the automatic construction of geospatial models. A mediated RDF structure in the "DataType" ontology can significantly reduce the number of required schema mappings between service messages. A number of observations point to important future directions.

First, the functional semantics of services need further consideration in the automation of service composition. For example, the OGC WCTS service performs a geometrical operation, which changes spatial reference coordinate systems without transforming the content or theme of input data. Therefore, the input data type and output data type of this service type are the same. As a result, current "DataType" driven service composition will not chain this service automatically, except that when RELAXED match option is used it might be chained because its output data type is a generic data type. This chaining result is incorrect and we will introduce rules to the current reasoning process to deal with this issue. The RELAXED match option also should be used carefully in the composition process and its result might need to be inspected by the human experts before use.

The second direction we intend to explore is the spatial reasoning. Our current inference is primarily based on a semantic match of the geospatial scientific theme ontologies. The example data is limited to raster imagery. Although some spatial and temporal constraints are considered in the service composition, a fully fledged consideration of the spatial characteristics of geospatial data, such as topological relation, is needed. The introduction of geospatial inference into the semantic Web area is still an emerging area of research. We will explore adding rules to the knowledge base to describe such spatial relationships as overlaps, within, and disjoint, and to enable inference based on these rules for data and service discovery in the chaining process.

The third direction is AI planning. There have been many efforts to automate Web service composition using AI planning (Srivastava and Koehler, 2003; Rao and Su, 2004; Peer, 2005). Our work provides a framework for automatic composition of geospatial Web services. Introduction of AI planning methods will enhance the capability of automatic service composition.

Acknowledgments

This work is supported by a grant from US National Geospatial-Intelligence Agency NURI program (HM1582-04-1-2021, PI: Dr. Liping Di). We are grateful to the two anonymous viewers and Dr. Barry Schlesinger for their detailed comments on this paper.

References

van der Aalst, W., 2003. Don't go with the flow: web services composition standards exposed, IEEE Intelligent Systems January/February, pp. 72–76.

Aissi, S., Malu, P., Srinivasan, K., 2002. E-Business process modeling: the next big step. IEEE Computer 35(5), pp. 55–62.

Altintas, I., Birnbaum, A., Baldridge, K., Sudholt, W., Miller, M., Amoreira, C., Potier, Y., Ludäscher, B., 2004. A framework for the design and reuse of grid workflows. In: Herrero, P., Prez, M.S., Robles, V. (Eds.), International Workshop on Scientific Applications on Grid Computing (SAG'04). Lecture Notes in Computer Science, vol. 3458. Springer, Berlin, Germany, pp. 119–132.

de la Beaujardière, J., 2004. Web map service (WMS). Version 1.3. OGC 04-024. Open Geospatial Consortium, Inc., 85pp.

Bermudez, E.L., 2004. Ontomet: ontology metadata framework. Ph.D. Dissertation, Drexel University, Philadelphia, USA, 177pp.

Bowers, S., Ludäscher, B., 2004. An ontology-driven framework for data transformation in scientific workflows. In: Rahm, E. (Ed.), Proceedings of the International Workshop on Data Integration in the Life Sciences (DILS 2004). Lecture Notes in Computer Science, vol. 2994. Springer, Berlin, Germany, pp. 1–16.

Cardoso, J., Sheth, A., 2005. Introduction to semantic web services and web process composition. In: Cardoso, J., Sheth, A. (Eds.), First International Workshop on Semantic Web Services and Web Process Composition (SWSWPC 2004). Lecture Notes in Computer Science, vol. 3387. Springer, Berlin, Germany, 14pp.

Di, L., 2004. 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, Istanbul, Turkey, 7pp (invited).

Di, L., 2005. A framework for developing web-service-based intelligent geospatial knowledge systems. Journal of Geographic Information Sciences 11 (1), 24–28.

Di, L., McDonald, K., 1999. Next generation data and information systems for earth sciences research. In: Proceedings of the First International Symposium on Digital Earth, vol. I. Science Press, Beijing, China, pp. 92–101.

Dogac, A., Kabak, Y., Laleci, G.B., 2004. Enriching ebXML registries with OWL ontologies for efficient service discovery. In: Proceedings of the 14th International Workshop on Research Issues on Data Engineering: Web Services for e-Commerce and e-Government Applications (RIDE' 04), Boston, USA, pp. 69–76.

Evans, J., 2003. Web coverage service (WCS). Version 1.0.0. OGC 03-065r6. Open Geospatial Consortium, Inc., 67pp.

Gruber, T.R., 1993. A translation approach to portable ontology specification. Knowledge Acquisition 5 (2), 199–220.

Guarino, N., 1997. Semantic matching: formal ontological distinctions for information organization, extraction, and integration. In: Pazienza, M.T. (Ed.), Information Extraction: a Multidisciplinary Approach to an Emerging Information Technology (SCIE 97). Lecture Notes in Computer Science, vol. 1299. Springer, Berlin, Germany, pp. 139–170.

Jaeger, E., Altintas, I., Zhang, J., Ludäscher, B., Pennington, D., Michener, W., 2005. A scientific workflow approach to distributed geospatial data processing using web services. In: 17th International Conference on Scientific and Statistical Database Management (SSDBM'05), Santa Barbara, CA, 27–29th June, pp. 87–90.

Klien, E., Lutz, M., Kuhn, W., 2006. Ontology-based discovery of geographic information services—an application in disaster management. Computers, Environment and Urban Systems 30 (1), 102–123.

Klusch, M., Gerber, A., Schmidt, M., 2005. Semantic web service composition planning with OWLS-Xplan. In: Agents and the Semantic Web. 2005 AAAI Fall Symposium Series, Arlington, VA, USA, 4th–6th November, 8 pp.

Ludäscher, B., Altintas, I., Berkley, C., Higgins, D., Jaeger, E., Jones, M., Lee, E., Tao, J., Zhao, Y., 2005. Scientific workflow management and the kepler system. Concurrency and Computation 18 (10), 1039–1065 (special issue on scientific workflows).

Lutz, M., Klien, E., 2006. Ontology-based retrieval of geographic information. International Journal of Geographical Information Science 20 (3), 233–260.

Lutz, M., Riedemann, C., Probst, F., 2003. A classification framework for approaches to achieving semantic interoperability between GI web services. In: Kuhn, W., Worboys, M.F., Timpf, S. (Eds.), Conference on Spatial Information Theory: Foundations of Geographic Information Science (COSIT 2003). Lecture Notes in Computer Science, vol. 2825. Springer, Berlin, Germany, pp. 200–217.

Martell, R., 2004. OGC^{TM} catalogue services—ebRIM(ISO/TS 15000-3) profile of CSW. Version 0.9.1. OGC 04-017rl. Open Geospatial Consortium, Inc., 87pp.

McIlraith, S.A., Son, T.C., 2002. Adapting Golog for composition of semantic web services. In: Proceedings of the Eighth International Conference on Principles and Knowledge Representation and Reasoning (KR'02), Toulouse, France, 22nd–25th April.

Müller, M.U., Warmerdam, F., Whiteside, A., Fellah, S., 2004. OpenGIS® web coordinate transformation service (WCTS) implementation specification. Version: 0.1.9. OGC 04-0XX. Open Geospatial Consortium, Inc., 116pp.

Papazoglou, M.P., 2003. Service-oriented computing: concepts, characteristics and directions. In: The Fourth International Conference on Web Information Systems Engineering (WISE 2003), Roma, Italy, pp. 3–12.

Peer, J., 2005. Web service composition as AI planning—a survey. University of St. Gallen, Switzerland, 63pp.

Percivall, G. (Ed.), 2002. The OpenGIS abstract specification, topic 12: OpenGIS service architecture. Version 4.3. OGC 02-112. Open Geospatial Consortium, Inc., 78pp.

Rao, J., Su, X., 2004. A survey of automated web service composition methods. In: Proceedings of the First International Workshop on Semantic Web Services and Web Process Composition (SWSWPC 2004), San Diego, CA, USA, pp. 43–54.

Raskin, R., Pan, M., 2005. Knowledge representation in the semantic web for earth and environmental terminology (SWEET). Computer and Geosciences 31 (9), 1119–1125.

Sheth, A., 2003. Semantic web process lifecycle: role of semantics in annotation, discovery, composition and orchestration. Invited talk at WWW 2003 Workshop on e-Services and the Semantic Web, Budapest, Hungary.

Sirin, E., Hendler, J., Parsia, B., 2003. Semi-automatic composition of web services using semantic descriptions. In: First Workshop on Web Services: Modeling, Architecture and Infrastructure. In conjunction with the Fifth International Conference on Enterprise Information Systems (ICEIS 2003), Angers, France, 22nd–26th April, pp. 17–24.

Srivastava, B., Koehler, J., 2003. Web service composition—current solutions and open problems. In: ICAPS 2003 Workshop on Planning for Web Services, Trento, Italy, pp. 28–35.

Sycara, K., Paolucci, M., Ankolekar, A., Srinivasan, N., 2003. Automated discovery, interaction and composition of semantic web services. Web Semantics: Science, Services and Agents on the World Wide Web, vol. 1(1), December, 27–46.

Vretanos, P., 2005. Web feature service (WFS) implementation specification. Version 1.1.0. OGC 04-094. Open Geospatial Consortium, Inc., 131pp.

Wei, L., Keqing, H., Wudong, L., 2005a. Design and realization of ebXML registry classification model based on ontology. In: Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'05), pp. 809–814.

Wei, Y., Di, L, Zhao, B., Liao, G., Chen, A., Bai, Y., Liu, Y., 2005b. The design and implementation of a grid-enabled catalogue service. In: 25th Anniversary of IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2005), COEX, Seoul, South Korea, 25th–29th July, pp. 4224–4227.

Wu, D., Parsia, B., Sirin, E., Hendler, J., Nau, D., 2003. Automating DAML-S web services composition using SHOP2. In: Proceedings of the Second International Semantic Web Conference (ISWC2003), Sanibel Island, FL, October, 16pp.

Yang, W., Whiteside, 2005. Web image classification service (WICS) implementation specification. Version: 0.3.3. OGC 05-017. Open Geospatial Consortium, Inc., 45pp.

Yue, P., Di, L., Zhao, P., Yang, W., Yu, G., Wei, Y., 2006. Semantic augmentations for geospatial catalogue service. In: Proceedings of the 2006 IEEE International Geoscience and Remote Sensing Symposium (IGARSS06), 31 July–4 August 2006, Denver, USA. 4 pp.

Zhang, R., 2004. Ontology-driven web services composition techniques. M.S. Thesis, University of Georgia, Georgia, USA, 63pp.