

Towards Semantic Composition of Geospatial Web Services – Using WSMO in Comparison to BPEL

Moses Gone¹, Sven Schade²

¹ Department of Geomatics, Computer Science and Mathematics,
University of Applied Sciences Stuttgart, Germany,
algone@googlemail.com

² Institute for Geoinformatics, University of Münster, Germany,
schades@uni-muenster.de

Abstract: Geospatial Web Services (GWS) provide access to geospatial data and expose basic processing functionality using Web Service technology. The complexity of geo-processing tasks necessitates the combination of basic Web Services to user-oriented and application-specific service compositions. Semantic heterogeneity and lack of automation have been identified as core problems that hinder the full exploitation of GWS. Current composition approaches do not address these problems. This paper presents an analysis of the use of Business Process Execution Language (BPEL), a state-of-the-art composition approach. Its major limitations in composing GWSs are highlighted. Semantic Web Services (SWS) composition using the Web Services Modeling Ontology (WSMO) is proposed as improvement. As basis for comparison a use-case application is developed in BPEL and in WSMO.

1 INTRODUCTION

Geospatial Web Services (GWS) are meant to enable seamless integration of distributed geo-processing and data provision functionality (OGC, 2001). Through GWS, the components of Geographic Information Systems (GIS) are decomposed and become accessible via Spatial Data Infrastructures (SDI) (Nebert, 2004; Shi, 2004). The separated components can be accessed and integrated over a network. The GWS in an SDI are loosely coupled and may be offered by different providers. The Open Geospatial Consortium¹ (OGC) broadly standardizes Web Service interfaces as an evolutionary framework for GWS. One of the most prominent OGC standards is the Web Feature Service (WFS) Implementation Specification (OGC, 2005b), an interface definition for the provision of vector data. Implementation specifications for frequently required processing functionalities, like coordinate transformation, are standardized, too (OGC, 2005a). While a

¹ see: <http://www.opengeospatial.org/>.

Web Processing Service (WPS) specification is under way (OGC, 2005c), processing GWS are currently implemented using non-standard interfaces. According to W3C², these proprietary interfaces are described using the Web Service Description Language (WSDL) and the services are communicating via the Simple Object Access Protocol (SOAP).

The complexity of processing applications requires combining GWS and W3C Web Services. Joining Web Services to support the execution of complex tasks is coupled with semantic heterogeneity problems (Zeng et al., 2003; Lutz, 2005; Lutz and Klien, 2006). These problems challenge Web Service composition, which are implemented by syntax-based approaches such as Business Process Execution Language (BPEL) (May, 2006). Semantic Web technology has been proposed as solution (Hakimpour et al., 2005). Semantic Web aims to make information on the Web machine-processable and understandable. The developments in the field of Semantic Web technology present an opportunity for adding value to GWS by enhancing their integration and automation.

This paper evaluates current service composition possibilities and examines how recent Semantic Web technology may improve the composition of GWS. BPEL composition is used for pointing out problems with the state-of-the-art. The limitations of BPEL composition approach are addressed by a semantic composition. The Web Services Modeling Ontology (WSMO) (Roman and Scicluna, 2005) is adopted as a basis for comparison. Opposed to other Semantic Web technology, WSMO provides an integrated solution to Web Service discovery, composition, and execution. By serving a modeling ontology for Web Services, WSMO is a general conceptual framework which remains open for extensions. The capabilities of BPEL and WSMO are highlighted by using a scenario, which was developed in an EU-funded project called SWING³. The scenario includes a use case and a set of available Web Services, which provide core functionalities in data provision and processing.

The remainder of this paper is structured as follows. Section 2 introduces the scenario and the application logic. BPEL and the WSMO approach are defined in section 3. The section includes pointers to related work. In section 4 an implementation of the application logic in BPEL is described and a possible implementation using the Web Services Modeling Ontology (WSMO) is documented. An implementation using the recent version of the reference implementation of WSMO, the Web Service Execution En-

² see: <http://www.w3.org/TR/2006/CR-wsdl20-primer-20060327/>.

³ The SWING project (FP6-26514) aims at deploying Semantic Web Service technology in the geospatial domain. Further information is available from: <http://www.swing-project.net>.

gine (WSMX) (Haller and Scicluna, 2005b) is sketched. Section 5 compares both approaches. The shortcomings of the BPEL approach are analyzed, requirements for semantic-aware GWS composition are identified, and it is discussed how WSMO addresses these requirements. The conclusions are drawn and future work is outlined in the final section.

2 SCENARIO AND PROCON APPLICATION LOGIC

The application logic serves a simple example of combining six decoupled Web Services from different providers. It bases on a use case that has been specified in the SWING project. The set of Web Services is provided by the project, too.

2.1 Use Case and Service Setting

The use case was developed by BRGM⁴, one of the partners in the SWING project. In brief, the use case concerns service composition, for supplying consumption-production information about aggregates. Six data and processing services are available for implementation (Table 1).

Table 1: Services available for the implementation.

| Name | Type | Description |
|-------------|--------------------------|--|
| Admin | OGC – WFS | Provides administrative data, like polygons for communities and population density per department. |
| Quarry | OGC – WFS | Provide data about quarries' location and allowed production. |
| INSEE | W3C – Data Service | Provides population data. |
| BRGM SEC | W3C – Data Service | Provides average consumption per capita value. |
| Consumption | W3C – Processing Service | Takes two numbers as input and returns their product. It should be used to multiply the average aggregate consumption value with a population count. |
| Aggregator | W3C – Processing Service | Parses the production values from a collection of quarry features and aggregates the values to give production value. |

An application implementing the use case enables the user to retrieve data about production and consumption of aggregates for a specific department (i.e. administrative region) in France. The application should employ OGC

⁴ see: <http://www.brgm.fr/>.

Web Services in combination with W3C Web Services. OGC Web Services are REST⁵-based (Fielding, 2000), and support http GET and POST requests with key-value-pair (respectively XML) encoding of request parameters. Contrary, W3C Web Services use SOAP and the interfaces are described via the Web Service Description Language (WSDL)⁶.

2.2 ProCon Application Logic

Based on the available services, the following application logic is used to implement the composition (Figure 1).

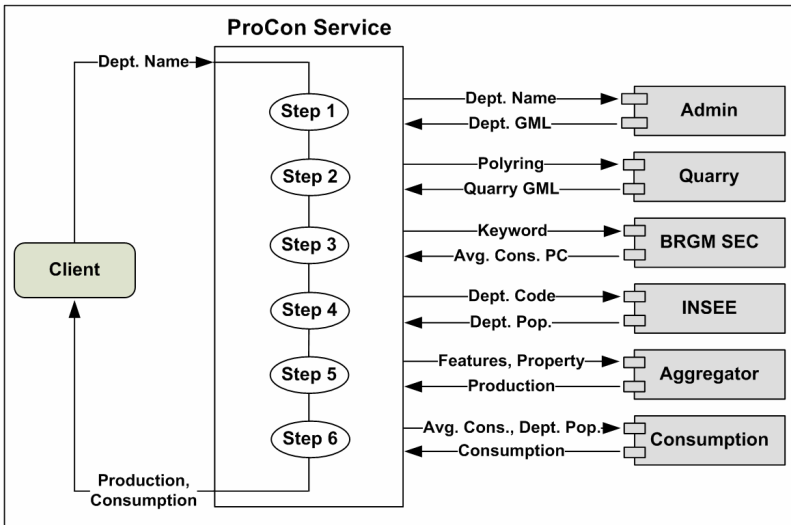


Figure 1: ProCon application logic.

The composition takes a department name as input. It computes the production rate and the consumption rate for the department (in tons per year). The consumption rate is computed by retrieving the department's population value from the INSEE service and multiplying it with a constant average consumption per capita. The production rate for each quarry in France is served by the Quarry WFS. The production rate for a specific department is computed by selecting the quarries inside the department and summing up the production rates of these quarries. The production and consumption values of the requested department are returned by the composition. In the

⁵ see: <http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm>.

⁶ see: <http://www.w3.org/TR/wsd1>

remainder, this application logic is called ProCon application logic. The composition service to implement the logic is called ProCon Service.

3 TWO POSSIBLE APPROACHES

WSMO and BPEL are related to existing work in Web Service, Semantic Web Service (SWS) and GWS composition. The basics of BPEL are outlined, the core functionalities of WSMO are described, and WSMX is introduced. Notably, both approaches were designed in favor of W3C Web Services. Geospatial Web Services, as defined by OGC, were not explicitly considered in the development of BPEL and WSMO.

3.1 Related Work

Several service composition techniques have been developed in parallel. On the one side, the business world has developed a number of XML-based standards to formalize the specification of Web Services, their discovery, composition and execution. This approach is primarily syntactical. On the other side, the Semantic Web community focuses on reasoning about Web resources using ontologies. The aim is automating composition.

Studies have been carried out on GWSs within a Service Oriented Architecture (SOA) environment (Aktas et al., 2006; Deng et al., 2006). This work does not include semantics within compositions. Lemmens has argued that, service discovery and composing mechanisms need much more expressive descriptions than WSDL (as used within BPEL) and OGC Capability Documents (Lemmens, 2004). Other studies address these shortcomings using SWS composition approaches (Sirin et al., 2003; Hakimpour et al., 2005; Haller et al., 2005b). They examine and demonstrate how current SWS technology fits into Enterprise Application Integration within SOAs. Accordingly, Web Services discovery, composition and execution to support publishing of geospatial data and processes have been explored (Klien et al., 2004; Jägery et al., 2005; Aerts et al., 2006).

Most of the studies mentioned so far focus on discovery and retrieval of GWS. Recently, attention is shifting towards GWS composition (Einspanier et al., 2003; Granell et al., 2004; Lemmens, 2004; Lutz, 2005). For example, the KEPLER scientific workflow system offers a Web Service composition and execution framework for distributed environmental modelling (Jägery et al., 2005). While Zhao and colleagues entirely achieve composition using BPEL (Zhao et al., 2005), Tanasescu and others combine WSMO and IRS III for composing Semantic GWS (Tanasescu et al., 2006). A similar approach has been used by Liping and others, who combine OWL-S and BPEL (Liping et al., 2006). In Liping's work, ontolo-

gies are used to give meaning to diverse data sources and geo-processing services. The work at hand considers semantics regarding client-service and service-service interaction in addition.

3.2 Orchestrating Services Using BPEL

A BPEL process aggregates and coordinates Web Services to collaborate in long-lived conversations. The work presented here uses only a subset of the constructs that are available in BPEL. Detailed reference can be found in the WS-BPEL 2.0 specification⁷.

Defining a process in BPEL involves creating a new composite Web Service together with a description, which defines the relationships between the composite service and its *Partner Web Services*. Partner Web Services involved in the execution of a composite service are added by defining *roles* and WSDL *port types* associated with those roles. *Variables* used in the process are defined for each message involved.

A process definition relies on XML Schema and WSDL for the definition of data types and service interfaces. The process can *receive* messages sent by partners in the process, *invoke* services synchronously or asynchronously and it can *reply* to the client if it is a request-response message exchange type.

3.3 Composing Web Services with WSMO

Currently several approaches try to bring Web Services usage to its full potential. All suggest possibilities to automate the three tasks of service discovery, composition and execution. An approach combining all of these aspects is the Web Service Modeling Ontology (WSMO) (Roman and Scicluna, 2005). WSMO defines a conceptual SWS model. It defines four basic conceptual modeling elements:

- *WSMO Ontologies* define a common agreed-upon terminology for an application domain. Following Guarino, an ontology is defined as “an engineering artifact, constituted by a specific vocabulary used to describe a certain reality, plus a set of explicit assumptions regarding the intended meaning of the vocabulary words” (Guarino, 1998).
- *WSMO Web Services* are descriptions that formalize various aspects of a Web Service. A service *capability* defines the Web Service functionality in terms of *non-functional properties*, *pre-conditions* and *post-conditions*. The *choreography* defines how a service interacts with its

⁷ see: http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsbpel.

clients. The *orchestration* describes how the provided functionality is realized internally, i.e. it specifies the interaction with the composed Web Services. WSMO Web Service descriptions use the available WSMO Ontologies. Invoking a Web Service means direct execution.

- *WSMO Goals* describe the user's desires with respect to the requested functionality. A Goal description consists of a requested *capability* and required *choreography* interfaces. WSMO Goals use WSMO Ontologies. Achieving a WSMO Goal means that the appropriate service has to be discovered, and executed by using choreography and orchestration.
- *WSMO Mediators* are used to deal with heterogeneities inherent in open and flexible architectures, such as the Service Oriented Architecture (SOA). They act as connectors between WSMO components. The use of Mediators is not elaborated in the work at hand.

Web Services *choreography* concerns the interactions of services with their users. In WSMO, choreography is the interface for consuming a Web Service, similar to a WSDL end point, and represents the external visible behavior of a service, including its communication structure and grounding. WSMO Choreography is based on Abstract State Machines (ASMs). It inherits the core principles of ASMs (Stollberg and Fensel, 2005).

Web Service composition in WSMO is achieved through *orchestration*. An orchestration defines how one Web Service makes use of other Web Services in order to achieve required functionality. According to the W3C definition, an orchestration defines the sequence and conditions in which one Web Service invokes other Web Services in order to realize a required function (W3C, 2004). A WSMO Orchestration is state-based and describes the desired workflow together with control and data flow. The *orchestration* description consists of a vocabulary that denotes the information space from the perspective of the Web Service. It consists of guarded transition of the form: if <condition> then invoke <Goal, Web Service or Mediator ID>.

The reference implementation of a WSMO is the Web Service Execution Engine (WSMX). WSMX uses the Web Service Modeling Language (WSML) as its internal language. Conditions are WSML *axioms*, which describe the states for which the transition rule should fire.

4 IMPLEMENTATION

This section describes the implementation of the ProCon Service using BPEL. The BPEL process is created using the NetBeans 5.5 Enterprise

Pack SOA tools⁸. Possibilities of implementing the same functionality following WSMO are discussed and an implementation in the currently available WSMX Version (0.2)⁹ is sketched.

4.1 Implementation of the Application Logic with BPEL

Before the ProCon Service can be implemented with BPEL, OGC services need to be made accessible via SOAP interfaces. In addition to the already available services (Section 2, Table 1), a WFS Proxy service is developed (Table 2). The Proxy WFS is implemented using Java Web Services technology.

Table 2: Additional services used in the BPEL implementation.

| Name | Type | Description |
|----------------|-----------------------------|--|
| WFS Proxy | (W3C) Proxy – Service | An intermediate service for invoking the Admin WFS and the Quarry WFS. This service is specific to the BPEL composition. |
| ProCon Service | (W3C) Composition – Service | A composite that utilises the other services. |

4.1.1 Enabling Invocation of OGC Services within BPEL

Current implementations of OGC services are REST-based, whereas BPEL requires SOAP services. A wrapper, which acts as a proxy to the OGC services is used to solve this heterogeneity. For ProCon, a proxy service was implemented with two operations; *getDepartments* and *getQuarries*.

The *getDepartment* method is used to invoke the AdminWFS. The method takes a String as argument (the department name) and returns a GML file that represents the department feature. Geography Mark-up Language (GML) (OGC, 2004) is the pre-defined output of each WFS. The GML file is wrapped into a SOAP message. The *getQuarries* method is used to wrap the *getFeature* request for the Quarry WFS into SOAP. The Quarry WFS service returns a GML file that represents quarries within the requested department.

4.1.2 BPEL Process Description

The implementation of the application logic is divided into two subtasks. The first task involves computing the consumption value per department. The second involves computing the production value for the same department. The results of these two subtasks are then combined and returned.

⁸ see: http://developers.sun.com/jsenterprise/nb_enterprise_pack/.

⁹ Recent implementations of WSMX are available from: <http://www.wsmx.org/index.html>.

To compute consumption value for a requested department, the *getDepartments* and *getQuarry* operation on the WFS Proxy are invoked. The `<flow>` construct is used to facilitate concurrent invocation of two services: INSEE and BRGM SEC. Invocation of these two services results in average consumption per capita value, returned from the BRGM SEC service and population count returned from the INSEE service (see also Figure 1, step 3 and 4). These two inputs are required to invoke a multiplier operation provided by the Consumption Service. This returns consumption value for the queried department.

Production is derived from aggregating the allowed production attributes of the quarry features. The Quarry WFS is invoked via the WFS Proxy service using the *getQuarries* operation (Figure 1, step 2). The output is a GML feature collection of quarries within a certain department. The quarries GML and two Strings (feature name and property name) are assigned to a variable representing input to the aggregate operation on the Aggregation Service (Figure 1, step 5). The two results, production and consumption values are assigned to a single variable, which is returned to the client.

4.2 GI Web Service Description in WSMO

This section shows how the use case application could be achieved within WSMO as the SWS framework. A prototypical implementation of the application logic is introduced using WSMX. In order to realize the ProCon Service with WSMX, the service composition is equipped with a WFS interface. This interface encapsulates the calculation of production-consumption values. Examples for GWS descriptions, specifically WFS descriptions, and Goal descriptions are given.

4.2.1 Defining Geospatial Web Services using WSMO

A semantic composition framework must provide meaningful service descriptions and enhance the composition process by providing some level of automation to the composition process. In WSMO, GWS descriptions contain the following elements:

- *Non-functional properties* are used to describe service metadata similar to those specified by OGC Capability Documents.
- *Capabilities* of the WSMO service description is used to semantically define the inputs and the outputs required for a geospatial service using:
 - *Pre-conditions* define the request to the service. The precondition states that a valid request, for example a *getFeature* request, must exist before the WFS service can be invoked.

- *Post conditions* define constraints on the output of a service invocation. If WFS is successfully executed, it will return a GML Feature Collection.
- *Interface* specifies the behaviour of the Web Service. It consists of, at least one, choreography and may include an orchestration in addition.
 - *Choreography* describes how a service can be consumed by a client. In case of WFS, each instance of a `getFeatureRequest` causes an invocation of the OGC service and the result is translated into a WSMML encoding of a GML Feature Collection. Adapters have to be defined, in order to implement Web Service requests from WSMX to OGC Services and to parse responses back into WSMX. Specific adapters can be build based on the generic *Adapter Framework* (DERI, 2005).
 - *Orchestration* describes the interaction with further services. This is, for example the case for the WFS, which encapsulates the ProCon application logic.

Considering the WFS description, “`getFeatureRequest`” is a concept, which is part of a globally available *WFS ontology* describing in- and outputs of WFSs. The “`FeatureCollection`” concept is part of an extensive *GML ontology* that captures the GML semantics. Application specific concepts are part of domain *ontology*. The example use case covers the domain of aggregate production (also termed quarrying) and consumption. Concepts of the domain ontology, like “`IndustrialSite`” or “`Basin`”, are used to capture the semantics of application specific feature types, i.e. models for the geo-spatial data that is returned by the WFS.

4.2.2 Defining Geospatial Task as WSMO Goal

Capturing a user goal sufficiently is one of the main tasks in using frameworks such as WSMO. For instance, the application user might pose a question such as “Give me production and consumption values for department X”. Such a goal must be formulated in a way that a service advertising such capability is found. The discovered service needs to be invoked based on available information and correct production and consumption values for “department X” have to be returned. In case no single service can satisfy the goal, services should be found that in combination resolve the user's goal.

As any WSMO Goal, the Goal for discovering and invoking the ProCon application logic resembles a service description, including both the capability and the interface section. The choreography of a Goal does not only express the users preferred mode of interaction with a WFS, it is also used

in service discovery. A service will be selected if its choreography interface matches that of the Goal.

4.2.3 Service Choreography and Orchestration in ProCon

Achieving ProCon orchestration involves two main tasks in WSMO. From the service-side, the services are described semantically using concepts defined in ontologies. This includes their properties, capabilities, and choreography interfaces. The six basic services only have a choreography interface. The application logic it encapsulated in a new Web Service with a WFS interface, the ProCon Service (Figure 2).

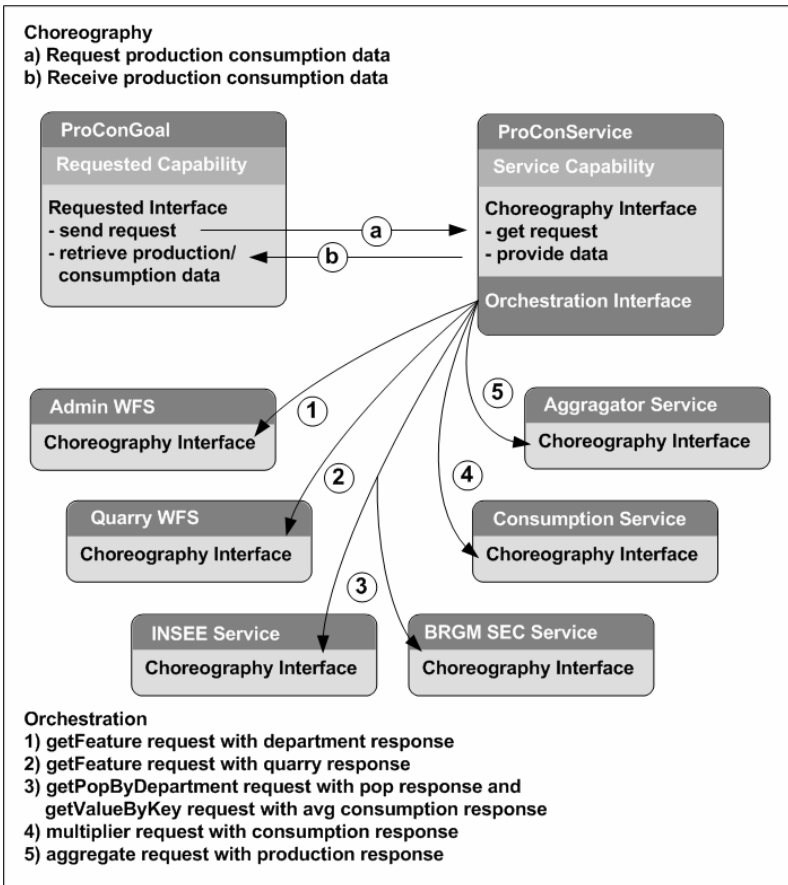


Figure 2: ProCon application orchestration in WSMO.

This service is described as a composite service, with a choreography and one orchestration interface. The choreography interface encapsulates the functionality and provides a `getFeature` request for production-consumption features with an according Feature Collection as response. The ProCon Service needs the orchestration interface since it does not provide the production and consumption data on its own, but combines other services.

4.2.4 Execution of the ProCon Web Service

Figure 3 gives a glass-box view of the ProCon service. The orchestration is described as an *Abstract State Machine* (ASM) (Haller and Scicluna, 2005a). It defines the interaction between the Web Service, which encapsulates the ProCon application logic and the six basic Web Services.

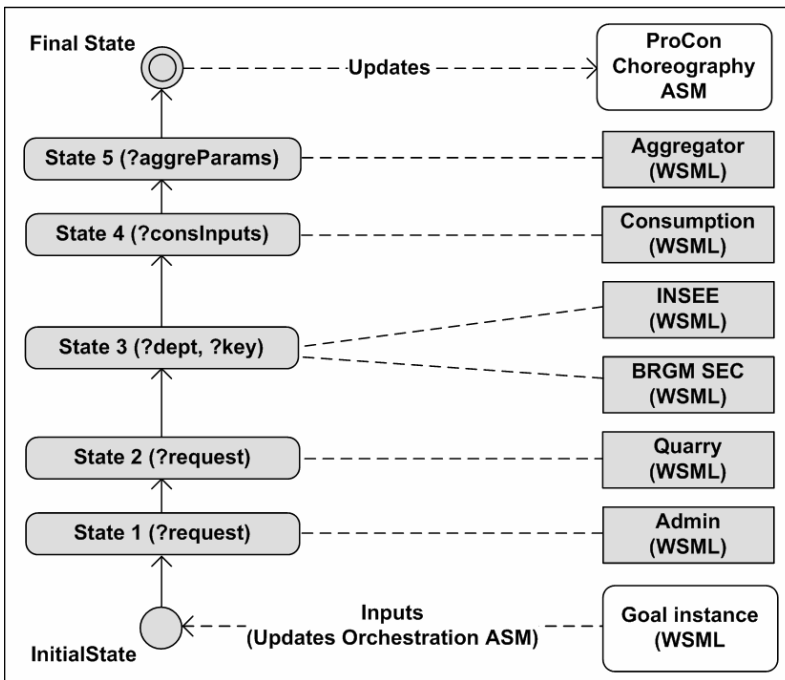


Figure 3: State transitions within the ProCon Service orchestration.

Assuming a suited Goal instance exists, a state is created in the ASM that enables the execution of Admin WFS (see also Figure 2, step 1). Using transition rules of the form “if <condition> then invoke <Web Service>”, the Admin WFS Service is invoked. It returns a GML Feature Collection as expected by the ProCon Service. The Feature Collection that is returned,

triggers a state change, i.e. a second transition rule is fired. This leads to the invocation of the Quarry WFS (see also Figure 2, step 2). The new state allows invoking the BRGM SEC and INSEE services and retrieve average consumption value and population data (see also Figure 2, step 3). This succession of states continues until the *Final State* is reached as defined by the post-condition of the ProCon Composite service. At this point then the user Goal can be resolved (see also Figure 2, step b).

4.2.5 Prototypical Implementation of the ProCon Service

The recent version of WSMX does not implement *orchestration*. Neither is complete discovery available. Reasoning is only supported for keywords (being part of *non-functional properties*) and on *post-conditions*. Anyhow, *Ontologies*, *Web Service* descriptions and *Goals* can be implemented.

```

webService _ "http://swing.brgm.fr:8080/axis/services/INSEE"
  nonFunctionalProperties[]

  importsOntology
    _ "http://swing.brgm.fr/INSEE"

  capability INSEECapability

  postcondition getPopulationFromDepartment
    definedBy
      ?response memberOf ins#INSEEgetPopulationByDepartmentResponse.

  interface ins#INSEEinterface

    importsOntology[]
    choreography INSEEEchor
    stateSignature sig
      importsOntology
        _ "http://swing.brgm.fr/INSEE"

    in
      concept ins#getPopulationFromDepartmentRequest
      withGrounding _ "http://swing.brgm.fr/axis/service/INSEE/In"

    out
      concept ins#getPopulationFromDepartmentResponse
      withGrounding _ "http://swing.brgm.fr/axis/service/INSEE/Out"

    transitionRules rules
      forall (?request) with (?request[ins#type hasValue "detailed"]
        memberOf ins#getPopulationFromDepartmentRequest) do
        add(?request memberOf ins#getPopulationFromDepartmentResponse)
      endForAll
  
```

Listing 1: WFS description in WSMO.

A first prototype of the application logic has been developed in the scope of the SWING project. Prototypic adapters for the WFS-WSMX connection, required Web Service descriptions (Listing 1), and the Goal (Listing 2) are

provided by SINTEF¹⁰, one of the project partners. A first set of ontologies capturing GML and domain knowledge have been developed by the Institute for Geoinformatics of the University of Münster¹¹. The WSMML ontologies are available from a web page hosted by the BRGM¹².

In the current WSMX environment, *choreography* needs to be used for implementing execution. It can be done by defining a set of transition rules in the *choreography* section of the Web Service description (bottom of Listing 1). The transition rules fire as soon as instances of requests (including all required parameters, like WFS version, bounding box etc.) are available. Consequently, the Goal must already contain some of the required parameters (bottom Listing 2). These parameters depend on the service instances used inside the composition, i.e. the person specifying the Goal requires knowing exactly which services are combined inside the ProCon Service.

```
goal proconGoal
  nonFunctionalProperties
    wsmstudio#version hasValue "0.5.4"
  endNonFunctionalProperties

  importsOntology[]

capability proc_capability

  importsOntology[]

postcondition proc_postcond

interface procon_Int
  choreography choreo
  stateSignature Sig
  in
    concept _ "http://localhost:8080/procon/procon_ont#proconRequest"

  out
    concept _ "http://localhost:8080/procon/procon_ont#proconResponse"

  transitionRules _#

ontology _ "http://localhost:8080/procon/procon_goal"
instance myProconRequest memberOf wfs#GetFeatureRequest
  wfs#Service hasValue "WFS"
  wfs#Version hasValue "1.1.1"
  wfs#TypeName hasValue "productionConsumption"
  wfs#OutputFormat hasValue "GML3"
```

Listing 2: Goal description for invoking a WFS service.

¹⁰ The official Web page is available from: <http://www.sintef.no>.

¹¹ The official Web page is available from: <http://www.uni-muenster.de/en/>.

¹² <http://swing.brgm.fr/ontologies>

5 DISCUSSION AND COMPARISON

Based on the implementation, shortcomings of the BPEL approach are highlighted. Requirements for semantic-aware GWS composition are identified and it is outlined how WSMO addresses these requirements. A discussion of the current status of WSMX is included.

5.1 Limitations of BPEL for GWS Composition

Although BPEL has good control over the workflow at design time, and it provides visual tools to model the process. It defines composition processes using variables defined in WSDL and XML schema. These syntactic definitions do not capture the semantics of the service capabilities. Based on syntactic descriptions, the meaning can only be derived from the names of the operations, for example *aggregate*, *multiplier*, *getValueByKey* etc. In some cases it might not be possible to understand the meaning of an operation by merely looking at the operation name. For instance, in the BRGM SEC service operation *getValueBykey*, it is not obvious what the operation does; this might mean that a certain key is required, but the key can mean anything.

BPEL offers only limited support for in the composition process. Compositions must be defined manually, by selecting the Web Services, by specifying the work-flow, and by making sure these services can cooperate. To be able to optimize the results of a composition, services should be composable at run-time. To be able to use a service efficiently, there need to be ways of determining when a service can be executed. This can be expressed by defining the conditions that hold before a service is executed and after a service has been executed.

The interaction between the composite service (ProCon service) and the aggregated services is hard-coded in the BPEL process. If user needs change, the BPEL process has to be rewritten. The whole composition might fail, as soon as one or more Partner Web Services are down, or the internal status of a Partner Web Service changes after the process is deployed.

Specific to GWS, OGC Web Services are REST-based and do not have WSDL description, and instead use OGC Capability Documents as service description. In this regard, OGC Web Services cannot be used directly within a BPEL process, which limits GWS composition using BPEL.

5.2 Addressing Composition Problems in WSMO

SWS frameworks such as WSMO allow attaching meaning to terminology used in service and data description by using ontologies. For instance, service inputs, outputs and operations can be clearly defined, and hence ambiguities and heterogeneities in terminologies are reduced.

The service composition does not have to exist at the time when a user poses the request, i.e. formulates the Goal. Compositions may employ different services, which is more flexible than using pre-selected services. Goals allow the specification of processes and tasks, for which suitable Web Services can be detected dynamically at execution time. WSMO included discovery mechanisms for identifying Web Services with suited semantics.

A service choreography description allows services and their clients to choose the appropriate times that they want to send a message or that they want to be able to receive a message, unlike in BPEL where such decisions are hard-coded in the service or client implementation. Definitions in a service description ensure that services can only be executed under certain conditions.

The generic approach followed by WSMO enables a seamless interaction between W3C and OGC Web Service. Services of both types can be used as components inside the same composition. In contrast to BEPL, WSMO services do not entirely rely on WSDL description. Services do not necessarily need to be WSDL- and SOAP-based in order being composable within this framework. WSMO choreography can include grounding to OGC services directly without using a proxy as was the case with BPEL.

As an additional benefit, a service can have different choreographies in WSMO, i.e. a single service can be used in different ways. Multiple choreographies enable multiple modes of interactions for the same Web Service. This enhances services reuse, for instance, the ProCon application may have three choreography interfaces, one that returns to the user production and consumption, one that returns just production data, and one that returns just consumption data.

5.3 Limitations of WSMO for GI services

As with any other SWS technology, the process building the required *Ontologies* (for example the GML ontology and domain ontologies) is manual and relies heavily on domain experts' knowledge. On the one hand, designing and developing ontologies takes considerable amounts of time. On the other hand, it adds value to the definition of data and process semantics.

Annotating Web Services using any of these Ontologies is a cumbersome process and is still largely carried out manually.

Considering WSMO, the user's desire is defined by Goals. However, it is hard to capture the user's objective sufficiently. The challenge of building Goals increases with the complexity of the processing task, i.e. of the required composition. Formulating Goals in WSML requires expert users who are familiar with the technologies and which know the available ontologies fairly well.

In order to connect GWS, like the WFS, to WSMO, specific adapters need to be implemented. In contrast to the proxy services required if using BEPL, one adapter per service type is sufficient, for example connecting WFSs requires one WFS adapter. The generic Adapter Framework provided by WSMX eases the development. Anyhow, WSMX is primarily designed for business applications with simple data models. Handling GML-encoded data within the WSMX execution engine challenges the adapter for converting between WSML and GML.

Regarding the available WSMX implementation, the WSMO discovery is not yet fully implemented and orchestration is under specified. In the presented solution, the user needs to include parameters in the Goal, which depend on the service instance realising the service composition. The WSMX engine is currently not capable of creating a composition of service instances on the fly. Implementing orchestration as suggested by WSMO would eliminate this drawback.

6 CONCLUSION

The emergence of Web Services, specifically OGC Web Services, has enabled syntactic interoperability in the execution of geo-processing tasks. Sets of decoupled services, which provide basic data and processing functionalities, can be used to compose new Web Services implementing application-specific tasks.

Two composition approaches, BPEL and WSMO, were discussed and examined. BPEL was used in order to highlight the problems of state-of-the-art service composition approaches. A use-case application (ProCon) was developed and implemented in BPEL and in WSMX.

SWS addresses the problems of current composition approaches by using ontologies for semantic descriptions of data and processing services, by allowing for dynamic service composition, and by enabling automatic service execution. Semantic service composition is an important approach for achieving interoperability not only within the geospatial domain.

It was discussed how limitation of using BPEL approach may be addressed and eliminated by semantic composition approaches, such as the Web Service Modelling Ontology (WSMO). Being generic and extendable, WSMO offers a complete conceptualisation for the description, composition and execution of SWSs. A prototypic implementation of the ProCon application logic in WSMX illustrated that W3C and OGC Web Services can be combined using the WSMO approach. Nevertheless, the currently available WSMX implementation does not offer all functionalities, which are required for dynamic composition.

In this work only composition of W3C services and WFSs was implemented. The use of different kinds of OGC services, including Web Map Services (WMS), Web Coverage Services (WCS), and Web Coordinate Transformation Services (WCTS) still needs to be examined. The Web Processing Services (WPS) deserves special consideration. Work on implementing WSMX is ongoing at DERI¹³.

ACKNOWLEDGEMENTS

The work was carried out while one of the authors (Sven Schade) was funded by the European Commission under the SWING project (FP6-26514). We would like to thank all project members. Lastly we would like to convey our sincere gratitude to colleagues at MUSIL¹⁴, University of Münster for their support and encouragement.

REFERENCES

- Aerts, K., Maesen K. and A. Van Rompaey (2006). A Practical Example of Semantic Interoperability of Large-Scale Topographic Databases Using Semantic Web Technologies. In: Proc. of the 9th Conference on Geographic Information Science (AGILE 2006), Visegrad, Hungary.
- Aktas, M.s. et al. (2006). Information Services for Grid/Web Service Oriented Architecture (SOA) Based Geospatial Applications. Technical Report, Advanced Information Systems Technology Program of NASA's Earth-Sun System Technology Office.
- Deng, M. et al. (2006). The Development of a Prototype Geospatial Web Service System for Remote Sensing Data. In: Proc of Geo-Imagery Bridging Continents - XXth ISPRS Congress, Istanbul, Turkey, pp. 218-222.

¹³ see: <http://www.deri.org/>

¹⁴ see: <http://musil.uni-muenster.de/>

- DERI (2005). WSMX Adapter Framework. Available from: http://wiki.wsmx.org/index.php?title=WSMX_Adapter_Framework, lastly accessed: 15th of June 2007.
- Einspanier, U. et al. (2003). Toward a Process Model for GI Service Composition. In: Proceedings of GI-Tage 2003, Münster, Germany.
- Fielding, R. (2000). Architectural Styles and the Design of Network-based Software Architectures, PhD. Thesis, UCLA, Irvine, Chapter 6.
- Granell, C., Poveda J., and M. Gould M. (2004). Incremental Composition of Geographic Web Services: An Emergency Management Context. In: Proceedings of the 7th Conference on Geographic Information Science (AGILE 2004). 2004. Heraklion, Greece.
- Guarino, N (1998). Formal Ontology in Information Systems. In: Proceedings of FOIS'98. 1998. Trento, Italy: IOS Press Amsterdam, pp. 81-97.
- Hakimpour, F., et al. (2005). Semantic Web Service Composition in IRS-III: The Structured Approach. In: Proceedings of Seventh IEEE International Conference on E-Commerce Technology (CEC'05), pp. 484-487.
- Haller, A., Gomez, J.M. and C. Bussler (2005). Exposing Semantic Web Service principles in SOA to solve EAI scenarios: WSMX - A Semantic Service-Oriented Architecture. In: Proceedings of The 14th International World Wide Web Conference (WWW 2005), Chiba, Japan.
- Haller, A. and J. Scicluna (2005). WSMX Choreography. WSMX Working Draft. Available from: <http://www.wsmo.org/TR/d13/d13.9/v0.1/#H332>, lastly accessed: 15th of June 2007.
- Jäger, E., et al. (2005). A Scientific Workflow Approach to Distributed Geospatial Data Processing using Web Services. In Proc. of the 17th International Conference on Scientific and Statistical Database Management (SSDBM 2005), Santa Barbara, CA, USA.
- OGC (2001). Introduction to OGC Web Services. In: OGC Interoperability Program White Papers. Open Geospatial Consortium Inc.
- OGC (2004). Geography Markup Language (GML) Encoding Specification (GML), Version 3.1.1., Open Geospatial Consortium Inc.
- OGC (2005a). Web Feature Service Implementation Specification, Version 1.1., Open Geospatial Consortium Inc.
- OGC (2005b). Coordinate Transformation Services Revision 1.00., Open Geospatial Consortium Inc.
- OGC (2005c). Web Processing Service Discussion Paper, Version 0.4.0., Open Geospatial Consortium Inc.

- Klien, E. et al. (2004). An Architecture for Ontology-Based Discovery and Retrieval of Geographic Information. In: Proceedings of the 7th Conference on Geographic Information Science (AGILE 2004). 2004. Heraklion, Greece.
- Lemmens, R. (2004). Applicability of the semantic web to geo service chaining. In: Proceedings of OpenGIS information communities and semantics working group meeting, Southampton.
- Liping, D. et al. (2006). Ontology-driven Automatic Geospatial-Processing Modeling based on Web-service Chaining. In: Proceedings of European Semantic Technology Conference, College Park, MD, USA.
- Lutz, M. (2005). Ontology-based Descriptions for Semantic Discovery and Composition of Geoprocessing Services. *Geoinformatica* 11(1), pp. 1-36.
- Lutz, M. and E. Klien (2006). Ontology-based Retrieval of Geographic Information. *International Journal of Geographical Information Science* 20(3), pp. 233-260.
- May, B (2006). Developer Guide to BPEL Designer. Available from: http://www.netbeans.org/kb/55/bpel_gsg.html, lastly accessed: 15th of June 2007.
- Nebert, D. (2004). Developing Spatial Data Infrastructures: The SDI Cookbook. Version 2.0.
- Roman, D. and J. Scicluna (2005). WSMO Working Draft: WSMO Orchestration. 2005 [cited 2007 4-01-2007]; Available from: <http://www.wsmo.org/2004/d15/v0.1/>, lastly accessed: 15th of June 2007.
- Sirin, E., Hendler, J. and B. Parsia (2003). Semi-automatic Composition of Web Services using Semantic Descriptions. Web Services: Modeling, Architecture and Infrastructure workshop in conjunction with I-CEIS2003.
- Stollberg, M. and D. Fensel (2005). Ontology-based Choreography and Orchestration of WSMO Services - WSMO Deliverable 14, Version 0.2. Available from: <http://www.wsmo.org/TR/d14/v0.2/>, lastly accessed: 15th of June 2007.
- Tanasescu, V. et al. (2006). A Semantic Web Services GIS based Emergency Management Application. In: Proceedings of Workshop on Semantic Web for eGovernment of ESWC 2006, ESWC Budva, Montenegro.

- Zhao, P., Deng, D., L. Di (2005). Geospatial Web Service Client, In: Proceedings of Annual Conference of the American Society for Photogrammetry and Remote Sensing (ASPRS 2005), Baltimore, Maryland.
- Zeng, L. et al. (2003). Quality Driven Web Services Composition. In: Proceedings of The 12th international conference on World Wide Web. ACM Press: Budapest, Hungary, pp. 411 - 421.

