International Conference on Hydroinformatics

8-1-2014

Integration Of Water Supply Distribution Systems By Using Interoperable Standards To Make Effective Decisions

Gabriel Anzaldi

Wenyan Wu

Andreas Abecker

Edgar Rubión

Aitor Corchero

See next page for additional authors

Authors
Gabriel Anzaldi, Wenyan Wu, Andreas Abecker, Edgar Rubión, Aitor Corchero, Ambreen Hussain, and Michael Quenzer

INTEGRATION OF WATER SUPPLY DISTRIBUTION SYSTEMS BY USING INTEROPERABLE STANDARDS TO MAKE EFFECTIVE DECISIONS

GABRIEL ANZALDI (1), WENYAN WU (2), ANDREAS ABECKER (3), EDGAR RUBIÓN (1), AITOR CORCHERO (1), AMBREEN HUSSAIN (2), MICHAEL QUENZER (3)
(1): Barcelona Digital Technology Centre, Roc Boronat Street 117 5th floor, Barcelona, Spain
(2): Faculty of Arts and Creative Technologies, Staffordshire University, United Kingdom
(3): Disy Informationssysteme GmbH, Ludwig-Erhard Avenue 6, Karlsruhe, Germany

The integration of water control and monitoring systems, distributed along the water supply distribution chain, is a main challenge for unifying into a single solution all decisional processes in water resource management. This paper presents the work done in the EU-FP7 WatERP project (http://www.waterp-fp7.eu/), where an intelligent architecture, based on OGC® standards to enable information/knowledge exchange, is being designed, implemented and deployed by harmonizing existing protocols and tools. Most of the current communication architectures follow the OGC® stack to provide a standard mechanism for gathering sensor information and publishing geospatial processes in an interoperable way. However, these architectures do not cover all the water supply distribution chain and they need to enhance data understanding towards effectively supporting water managers' decision making. Our proposed architecture combines a Service Oriented Architecture (SOA) with a Multi Agent System (SOA-MAS) and complements it by a Water Management Ontology (WMO) to support interoperability and intelligent orchestration of system functionalities. The SOA supports interoperability between systems through OGC® standards (SOS, WPS, WaterML2). The WMO permits data exchange in an interoperable way by using a common and shared vocabulary throughout the architecture.

INTRODUCTION

Today, water managers use many unconnected, tools and data sources for their decision processes. Accordingly, water systems are communicated in different communication languages that mainly depend on software vendors. Therefore, information collection, data exchange and system monitoring take place independently in each part of the water supply distribution system. Further, the tracking of water flow and volume to ensure proper operations and to detect system abnormalities are done manually. Water managers need information from different parts of the system for their decision making and planning. As a solution, one has to harmonize water systems and communication languages in an interoperable platform that supports water managers' work to make more effective and efficient decisions regarding the whole chain. To improve decision making, several authors have proposed SOA-based interoperable software where web services based on hydrological procedures perform the connection between different involved systems. [1] proposes an XML/SOAP web service architecture with a specific XML language for data exchange between services and user interfaces. Most of the solutions in the literature are not standardised with respected to Database structure (customized Entity-Relationship model) such that the understanding of hydrological information depends on the concepts defined in non-standardised XML or databases. This lack is overcomed in approaches that use the OGC® stack complemented by an OpenMI, ODM data model and WaterML2 [2]. But these approaches cannot include new systems without manually re-modelling the architecture. Further, they are only focused on the management of hydrological sensor data without taking into account the interconnection of different management tools available in the water supply and distribution chain.

The WatERP architecture relies on a knowledge base (KB) to support the interoperability between systems distributed along the water supply distribution chain. The architecture can auto-manage the integrated building blocks (controlling, monitoring and management systems) in order to provide necessary information for each of them. It is based on open standards like OGC® SOS, WPS for process integration, and WaterML2 for data exchange. The architecture permits data exchange in an interoperable way by the use of an ontology which models the hydrological water domain (information and management) to enhance water data understanding and machine readability. The combination of SOA-MAS and KB provides the necessary interoperability through a common communication language currently not supported by WaterML2. The proposed architecture can be extended by more systems in a dynamic and automatic way. It permits the water managers to enhance their daily decision-making by combining the SOA-MAS with an ontological data understanding towards orchestration of decisional systems.

The next section "WatERP SOA-MAS Architecture" describes the interoperable WatERP architecture. The "WatERP Knowledge Base Interoperability" section describes data understanding and knowledge sharing. The subsequent one, "SOA Open Interface", explains the interoperability acquired by the use of the OGC® stack in an open environment. The "MAS Matchmaking Orchestration" section shows the implemented multi-agent process for intelligent orchestration of the whole architecture. Finally, conclusions and main outcomes are summarized.

SOA-MAS ARCHITECTURE

The SOA-MAS architecture [3] shall accomplish the following objectives: *(i)* link each decisional/informational system to help the integration in a collaborative framework, *(ii)* provide near real–time information flow, *(iii)* distribute intelligence to generate actions and alerts related to management processes, and *(iv)* perform orchestration of existing and new management tools. The architecture (Figure 1) has three layers: *(1) external data integration* enables information gathering from stakeholders' systems (water authorities, water utilities, water distributors, etc.); *(2) MAS framework* manages and orchestrates the whole architecture; and *(3) Open Interface* integrates and standardizes the use of building blocks (Decision Support Tools, Demand Management Tools, Hydrological Forecast Tools, etc.).

(1) The *external data integration layer* transforms data into WaterML2 through an XSLT wrapper, and stores it in an OGC® Sensor Observation Service Server (OGC® SOS Server). This server contains hydrological observations that are consumed by an internal OGC® SOS Client that feeds the Water Data Warehouse (WDW) and also populates the WMO. The WDW collects raw data and employs an Extract-Transform-Load (ETL) pipeline for data validation, data cleansing, and data harmonization. The WDW publishes cleaned observation data through

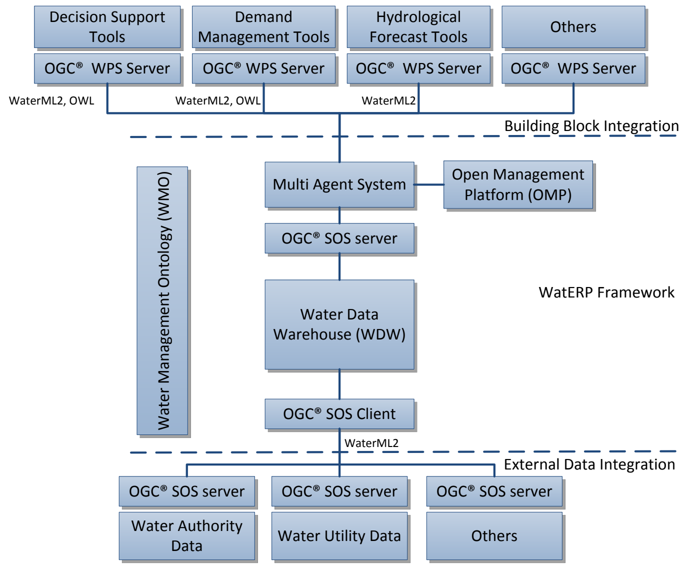

Figure 1. WatERP architecture.

the WDW OGC® SOS Server managed by the MAS in order to provide the time-series data to the rest of the architecture. Thereby, we avoid a direct dependency from the data sources. The architecture is interoperable with existing data infrastructures (e.g., water operators' databases) and only depends on the content provided by the OGC® SOS Server which gathers particular data from hydrological sources. In the case of a water-domain organization that would directly feed the WDW OGC® SOS Server with their hydrological systems/ sensors, they only need to integrate the observation results with the widely used OGC® standards.

(2) The *MAS* is responsible for managing and orchestrating the integrated building blocks, the ontology and the integrated observations stored in the WDW. The MAS has been implemented using the Jadex platform. Utility-based BDI (Belief-Desire-Intention) agents have been modelled because of their wide acceptance by the scientific community [4], and of their easy understandability (human-like process) which facilitates implementation and maintenance.

(3) The *WatERP Open Interface* was implemented as a public standard for connecting existing building blocks within the water supply distribution chain. The SOA architecture is taken as a basis because it offers technological interoperability, loose coupling and location transparency. OGC® WPS provides features to manage the discovery and binding of services.

Throughout the architecture two *data exchange formats* are used; WaterML2 to exchange hydrological time series and OWL files to exchange water domain knowledge. Because OGC® WPS allows easy exchange of any type of data, the architecture can support other formats.

WATER MANAGEMENT ONTOLOGY INTEROPERABILITY

The WMO provides a common information access and supports decision-making. The KB represents the water domain knowledge by defining *(i)* human-made interactions and decision making in the water supply and distribution chain; *(ii)* water resource availability; *(iii)* ecological, cultural and social functions of water resources and potential impacts of changes on hydrological regimes; *(iv)* current water infrastructure/assets and the economic value of water; *(v)* administrative, policy or regulatory issues of relevance; and *(vi)* sectorial use and water hierarchy. The novelty of the WMO over the state-of-the-art lies in the incorporation of human-made in-

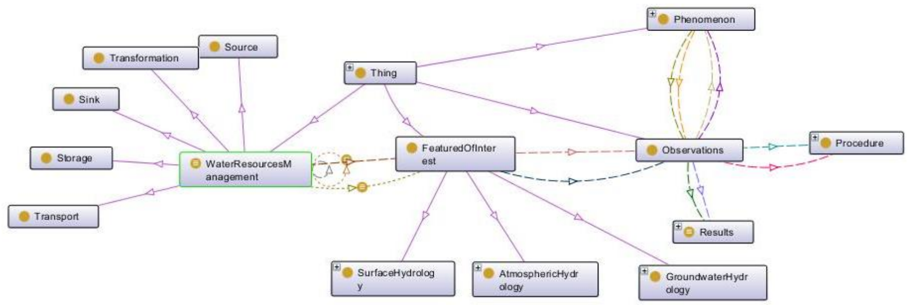

Figure 2.WMO highlights

teractions and decision-making processes carried out in the water domain. This allows alignment of water-physical objects ("*FeaturesOfInterest*") with decisional concepts. Thus, the WMO can represent the decisional correspondence (direct or indirect) between physical hydrological elements and how these decisions affect the mentioned real-world elements. This decisional correspondence is supported by the real-objects situation ("*FeaturesOfInterest*") that gathers hydrological information by an observation-and-measurement process described by "*observations*", "*procedures*", "*phenomena*" and "*results*" (Figure 2). Through this process, the ontology is aligned with the OGC® directives of sensor measures for hydrological systems [5] and provides easy information access through URIs for each ontology term (e.g., "*http://www.waterp-fp7.eu/WatERPOntology.owl#Phenomenon*"). This representation also supports data provenance by describing the real nature of the data in the observation process.

The presented KB has been constructed by defining standard water domain concepts that allow understanding of terms from other relevant water-domain vocabularies such as SSN, SWEET, HY_FEATURES and WaterML2. The WatERP ontology reuses these concepts, and aligns them with the current WatERP definitions towards enhancing the informational interoperability between systems. These relevant terms have been included into the ontology by the definition of semantic annotations. These annotations correspond with "*alignedWithCUAHSI*", "*alignedWithWaterML2*", "*alignedWithSSN*", "*alignedWithHY_FEATURES*" and "*alignedWithSWEE*T" annotation properties defined inside the WMO. Based on the annotations properties, the WMO materializes the alignments during the ontological population in order to define internally needed "*owl:equivalentClass*" and/or "*owl:equivalentProperty*" to let the ontology know which classes have the same meaning. The "*owl:sameAs*" has been used to indicate which instances refer to the same individual. Then, in the reasoning and querying stage, the WMO can merge knowledge from the above mentioned ontologies in order to provide the required information for the water manager by merging external concepts and vocabularies with the WMO.

The WMO contributes to interoperability by *(i)* the incorporation of standardized concepts into the ontology, *(ii)* the definition of ontological alignments with external representative water-domain ontologies to facilitate knowledge re-usability; *(iii)* the representation of human-interactions to support decision-making processes in the water supply distribution chain; and *(iv)* easy information access by the use of URIs to represent each ontology term. The WMO is consumed by the SOA-MAS through a SESAME triple-store where the ontological resources are stored. The SOA-MAS provides specific information about the water domain (e.g., volume of the reservoirs) when it is required (defined as input in a process description) to execute a building-block process (e.g., DS tools, DM tools, OMP). Hence, the MAS interacts with the SESAME front-end by SPARQL queries sent in HTTP/JSON or REST. As a result, the SESAME front-end responses are sent to the SOA-MAS architecture in JSON or OWL format, and then the resulting information is given to other building blocks and/or user interfaces.

SOA OPEN INTERFACE

The SOA integration ([6], [7], [8]) is extended for the management of water resources to standardize communication, integrate building blocks, and facilitate the extensibility of the framework. The SOA Open Interface allows exchange of information between systems, binding of building blocks and incorporation of new ones into the architecture.

The Open Interface is based on the OGC® WPS server implementation to integrate building blocks in the WatERP framework following OGC® best practices [9]. OGC® WPS facilitates the discovery and binding by the compulsory definition of the operations: *(i)* "*getCapabilities*"; *(ii)* "*describeProcess*"; and *(iii)* "*execute*". The "*getCapabilities*" operation allows MAS to obtain service metadata and list the processes that a building block can execute. The "*describeProcess*" operation provides detailed information (Figure 3) about a requested building-block process, i.e. the necessary inputs, their allowable formats, and the outputs that can be produced. The "*execute*" operation is aimed to run the specified building-block process, once required input parameters have been passed. These three operations provide the needed mechanism to discover building block's processes ("*getCapabilities*"), to know the required input and produced output parameters and their formats ("*describeProcess*"), and to execute the process.

A semantic necessity has been found in the OGC® WPS – "*describeProcess*" operation which (Figure 3) only provides *syntactic* and structural information for carrying out the matchmaking orchestration (e.g., Type of format, encoding and needed XML schema). This information is not enough to perform matchmaking orchestration because it is not possible to differentiate between two processes with similar syntax without understanding their *semantic* nature. For example, temperature forecast and demand forecast could be represented by the same format (WaterML2), but the nature of the information is widely different. To overcome this issue, semantic annotations [10] in the "*Metadata*" nodes [11] have been applied. The annotations add semantic information to the inputs and outputs in the process description, thus avoiding misunderstandings of data. Figure 4 shows the "*Metadata*" node describing an output parameter corresponding to an estimated temperature-phenomenon time series. Therefore, the "*Keyword*" nodes contain the needed ontological resource associated with the observation, and the "*Type*" node indicates the ontological source used to describe the nature of the parameter.

The Open Interface defines guidelines for enhancing interoperability between systems meanwhile the MAS orchestration is facilitated. These guidelines concern: *(i)* the implementation of the building block inside an OGC® WPS Server; *(ii)* providing semantic annotations in

```
<Format>
  <MimeType>text/xml</MimeType>
    <Encoding>UTF-8</Encoding>
    <Schema>http://www.opengis.net/waterml/2.0</Schema>
</Format>
```

Figure 3. Response for the "*describeProcess*" operation

```
<ows:Metadata>
  <ows:Keywords>
    <ows:Keyword>Temperature</ows:Keyword>
    <ows:Keyword>Estimated</ows:Keyword>
    <ows:Type codeSpace=”http://www.waterp-fp7.eu/WatERPOntology.owl#”>WatERPOntology</ows:Type>
  </ows:Keywords>
</ows:Metadata>
```

Figure 4. Response for the *"describeProcess"* operation with semantic annotations

the “*getCapabilities*” operation response inside “*ServiceIdentification*” node [11] to facilitate the classification of building block (DS tools, DM tools, HF tools); *(iii)* providing semantic annotations to the “*describeProcess*” operation response on “*Input*” and “*Output*” nodes to let the MAS understand the parameter nature; and *(iv)* knowing the nature of an observation by defining semantic annotations aligned with the WMO.

MAS MATCHMAKING ORCHESTRATION

MAS matchmaking orchestration ([12], [13]) shall manage information flows and system executions towards achieving water-managers’ information needs for decision-making. The MAS matchmaking process consists of fitting integrated process inputs and outputs to support decision-making in an unassisted way. It relies on the SOA architecture to discover and bind available processes published by a specific OGC® WPS Server for a concrete building block.

The matchmaking is carried out looking in the whole architecture for similar information by matching semantic and syntactic parts between two building blocks. Two building blocks are linked if the published operation that one of the building blocks can provide (via the specific OGC® WPS), accomplishes both; the semantic annotations and OGC® WPS definitions that are required for executing the other building block. This matchmaking is carried out by looking for capabilities using the agents network: *(i)* an agent for interacting with the user to gather water-manager requirements (OMP-Agent); *(ii)* an agent to interact with the ontological instantiation (SESAME-Agent); *(iii)* an agent to manage the SOS Servers (SOS-Agents); *(iv)* an agent to manage the building-block integration (OGCWPS-Agent); *(v)* agents to manage building blocks (DMS-Agent, DSS-Agent and hydrological forecasts agent - HF-Agent); and finally *(vi)* a Jadex agent (YellowPages-Agent) aimed at managing the registered services on the Yellow page.

The MAS matchmaking orchestration is applied in two fluxes: (i) *building block integration* is executed when a new building block (with a specific OGC® WPS) is registered in the platform, and (ii) *invocation process* is initiated when the water manager asks for specific information by using the OMP (WatERP Open Management Platform).

The *building block integration* process is used for cataloguing new building block processes. It is also able to analyze if new processes have the information necessary for execution. It consists of several steps that go from adding a new specific agent to the architecture until the integrated building block can be called by the MAS (Figure 5, aX steps): When a new building block has been noticed on MAS, the MAS creates an OGCWPS-Agent that is bound with the new OGC® WPS Server (step a1). The next step (a2) finds out the type of the new building block by calling the “*getCapabilities*” operation of the associated OGC® WPS-Server. This allows the MAS to know the type of the new block based on defined semantic annotations (DS

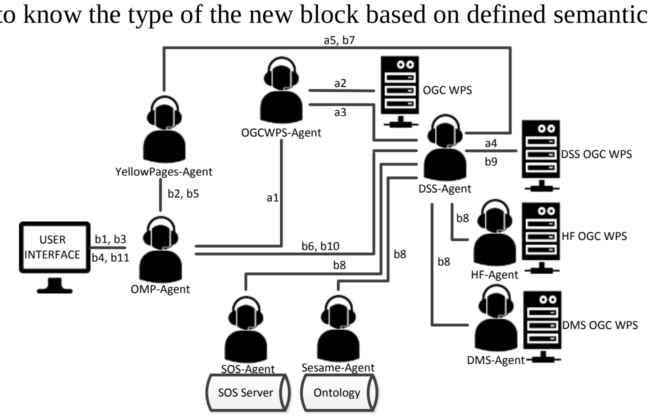

Figure 5. Building block integration process (aX steps) and invocation process (bY steps)

tools, DM tools, HF tools) in the "*ServiceIdentification*" node. After that, the OGCWPS-Agent instantiates a more specific agent (DSS-Agent, DMS-Agent, HF-Agent, etc.) to attend the building block, and delegates the responsibility of analysing the building-block processes (a3). Calling "*describeProcess*", the instantiated agent asks for the available processes that the building block can serve (a4). The returned available processes (with their inherited input/ output parameters) are checked by the specific agent by using the YellowPages-Agent (a5). It checks if the information required to execute a process is available. In case that the process requirements can be satisfied by the MAS, the specific process is registered on the YellowPages-Agent. The inclusion of a new process in the yellow pages can generate new processes that accomplish the necessary inputs to execute some required building blocks (waiting in the non-executable process). Therefore, when a new process is added in the yellow pages, the non-executed process must be reviewed. To finish the *building block* process, the defined specific agent enters a stand-by mode until its execution will be required for the MAS.

The *invocation process* provides to the water manager required information by executing building blocks. This process (Figure 5- bY steps) is performed by the interaction of the water manager with the user interface (OMP). The defined need (e.g., "*re-allocate water resources*") in the OMP is transformed into a goal in the OMP-Agent (step b1). Then, the OMP-Agent asks the YellowPages-Agent for the process(es) that can achieve this goal (b2). The YellowPages-Agent looks up published process(es) inside the Yellow pages and returns the building blocks which can fulfil the goal. Once the OMP-Agent receives the suitable processes, these are returned to the OMP (b3). The water manager selects the suitable available process. The selected process is transferred as a goal that receives the OMP-Agent (b4). Once again, the OMP-Agent asks the YellowPages-Agent for the specific agent (DSS-Agent, DMS-Agent, HF-Agent, etc.) that is responsible to execute the process (e.g., DSS-Agent) (b5). The OMP-Agent invokes this specific agent (e.g., DSS-Agent) in order to delegate the execution of the required process (b6). Later, the specific agent asks the YellowPages-Agent for agents that are able to satisfy the requirements obtained by the *"getDescription"* operation for the current process to execute (e.g., "*inflows/outflows variables and demand forecasting*" defined in an XML file) (b7). The YellowPages-Agent (b8) finds the agents (e.g., SOS-Agent and DMS-Agent) that provide the requirements and returns them to the specific agent (e.g., DSS-Agent). Then, the specific agent invokes the agents which are able to provide the required information (e.g., SOS-Agent and DMS-Agent) through the "*execute*" operation on the DMS-Agent and "*getObsevation*" on SOS-Agent case. Steps b7 and b8 can be performed again by the agents invoked (e.g., SOS-Agent and DMS-Agent) if they require extra data to provide to the initiator agent (e.g., DSS-Agent) the required answer (recursive agent invocation). When all input parameters are acquired, the initiator agent invokes the *"execute"* operation by using the gathered parameters (e.g., Inflows, Outflows and Demand Forecasting) (b9). Then, the operation is executed and the results are sent to the OMP-Agent (b10) that visualizes the operation's result into the OMP (b11).

CONCLUSIONS

The proposed architecture harmonizes the communication between systems that control, monitor and manage the water supply distribution chain by using a SOA-MAS approach together with a knowledge base driven by the WMO. The SOA-MAS architecture permits to dynamically and automatically manage building blocks. It goes beyond the state of the art by the enhancement of the OGC® definition through including a semantic alignment between the WPS-XML process description and the ontological vocabulary constructed for the hydrological domain. The SOA-MAS architecture is able to understand the meaning of the needed/resulting parameters that a building block needs/offers meanwhile data provenance is maintained throughout the whole architecture. The KB has been constructed with the aim of supporting system interoperability by providing a common vocabulary that serves to categorize water domain information (observation-and-measurement process) and governance aspects (human-made interactions, economical aspects, etc.). The KB improves the current state-of-the-art by the definition of a vocabulary that combines the hydrological observations and sensing process with an operational and management perspective for the water supply and distribution chain.

ACNOWLEDGEMENTS

This work has been supported by the European Commission through the 7th Framework Programme under the WatERP project, "Where Water Supply meets Demand" (ref. GA318603).

REFERENCES

[1] *Kanwar R., Narayan U., and Lakshmi V., (2010), "Web Service based Hydrologic Data Distribution System", Computers & Geosciences, Vol. 36, (2010), pp. 819-826*

[2] *Goodall J., Robinson B., and Castronova A., "Modelling water resource systems using service-oriented computing paradigm", Environmental Modelling & Software, (2010), Vol. 26, pp. 573-582*

[3] *Anzaldi G., Chomat C., Rubión E., Corchero A., Moya A., Moya C., Ciancio J., Helmbrecht J., "A holistic ICT solution to improve matching between supply and demand over the water supply distribution chain", 8th Conference on sustainable development of energy, water and environment systems, (2013)*

[4] *Rao, A. S., Georgeff M. P., "Decision procedures of BDI", Journal of logic and computation, (1998), pp. 293-233*

[5] *Taylor P., "OGC® Implementation Specification: WaterML 2.0: Part 1- Timeseries", (10-126r3), Open Geospatial Consortium Inc., (2012)*

[6] *Granell C., Díaz L., Gould M., Pascual V. Guimet J., Pepe M., Carrara P., "Developing geoprocessing services for a hydrological model application", 27th EARSeL Symposium on GeoInformation in Europe, (2007), pp. 369-376*

[7] *Lecca G., Petitdidier M., Hluchy L., Ivanovic M., Kussul N., Ray N., Thieron V., "Grid computing technology for hydrological applications", Journal of Hydrology, Vol. 403, (2011), pp. 186-199*

[8] *Castronova A., Goodall J., Elag M., "Models as web services using the Open Geospatial Consortium (OGC®) Web Processing Service (WPS) standard", Environmental Modelling & Software, Vol. 41, (2013), pp. 72-83*

[9] *Schut P., "OpenGIS Web Processing Service", Version 1.0.0(05-007r7), Open Geospatial Consortium Inc., (2007)*

[10] *Houbie F., Duchesne P., Maué P., "Semantic annotations in OGC® standards", Version 2.0 (08-167r2), Open Geospatial Consortium Inc., (2012)*

[11] *Whiteside A., "OGC® Web Services Common Specification", Version 1.1.0 (06-121r3), Open Geospatial Consortium Inc., (2007)*

[12] *Ebrahimi B., Bertels K., Vassiliadis S., Sigdel K., "Matchmaking within multi-agent systems", ProRisk 2004, Vendhoven, Netherlands, (2004), pp 118-124*

[13] *Maximilien E., Singh H., "Multiagent system for dynamic web services selection", 1st WS Service-Oriented Computing and Agent-Based Engineering at AAMAS, (2005) , pp 25-29*