Bridging the Gap between Mobile Application Contexts and Semantic Web Resources

Stefan Dietze, Alessio Gugliotta, John Domingue
Knowledge Media Institute,
Open University,
MK7 6AA, Milton Keynes, UK
{s.dietze, a.gugliotta, j.b.domingue}@open.ac.uk

ABSTRACT

Context-awareness is highly desired, particularly in highly dynamic mobile environments. Semantic Web Services (SWS) address context-adaptation by enabling the automatic discovery of distributed Web services based on comprehensive semantic capability descriptions. Even though the appropriateness of resources in mobile settings is strongly dependent on the current situation, SWS technology does not explicitly encourage the representation of situational contexts. Therefore, whereas SWS technology supports the allocation of resources, it does not entail the discovery of appropriate SWS representations for a given situational context. Moreover, describing the complex notion of a specific situation by utilizing symbolic SWS representation facilities is costly, prone to ambiguity issues and may never reach semantic completeness. In fact, since not any real-world situation completely equals another, a potentially infinite set of situation parameters has to be matched to a finite set of semantically defined SWS resource descriptions to enable context-adaptability. To overcome these issues, we propose Mobile Situation Spaces (MSS) which enable the description of situation characteristics as members in geometrical vector spaces following the idea of Conceptual Spaces (CS). Semantic similarity between situational contexts is calculated in terms of their Euclidean distance within a MSS. Extending merely symbolic SWS descriptions with context information on a conceptual level through MSS enables similarity-based matchmaking between real-world situation characteristics and predefined resource representations as part of SWS descriptions. To prove the feasibility, we provide a proof-of-concept prototype which applies MSS to support context-adaptation across distinct mobile situations.

KEYWORDS

Semantic Web, Mobile Computing, Internet Reasoning Service, Conceptual Spaces, Context.

INTRODUCTION

Current and next generation wireless communication technologies will encourage a widespread use of available resources – data and services - via a broad range of mobile devices resulting in the demand for a rather context-adaptive resource retrieval. Context-adaptation is a highly important feature across a wide variety of application domains and subject to intensive research throughout the last decade (Dietze, Gugliotta & Domingue, 2007; Schmidt & Winterhalter, 2004; Gellersen, Schmidt & Beigl, 2002). Whereas the context is defined as the entire set of surrounding situation characteristics, each individual situation represents a specific state of the world, and more precisely, a particular state of the actual context (Weißenberg, Gartmann & Voisard, 2006). Particularly, a situation description defines the context of a specific situation, and it is described by a combination of situation parameters, each representing a particular situation characteristic. Following this definition, context-adaptation can be defined as the ability of Information Systems (IS) to adapt to distinct possible situations.

To achieve this, we base on a promising technology for distributed and highly dynamic service oriented applications: Semantic Web Services (SWS). SWS technology (Fensel et al., 2006) addresses context-adaptation by means of automatic discovery of distributed Web services as well as underlying data for a given task based on comprehensive semantic descriptions. First results of SWS research are available in terms of reference ontologies – e.g. OWL-S (Joint US/EU ad hoc Agent Markup Language Committee, 2004) and WSMO (WSMO Working Group, 2004) – as well as comprehensive frameworks (e.g. DIP project1 results). However, whereas SWS technology supports the allocation of appropriate services for a given goal based on semantic representations, it does not entail the discovery of appropriate SWS goal representations for a given situation. Particularly in mobile settings, the current situation of a user heavily determines the intentional scope behind a user goal and consequently, the appropriateness of particular resources. For instance, when attempting to retrieve localized geographical information, the achievement of a respective goal has to consider the location and device of the user.

Despite the strong impact of a (mobile) context on the semantic meaning and intention behind a user goal, current SWS technology does not explicitly encourage the representation of domain situations. Furthermore, the symbolic approach - describing symbols by using other symbols without a grounding in the real world - of established SWS and Semantic Web (SW) representation standards in general, such as RDF (World Wide Web Consortium, W3C, 2004a), OWL (World Wide Web Consortium, W3C, 2004b), OWL-S (Joint US/EU ad hoc Agent Markup Language Committee, 2004), or WSMO (WSMO Working Group, 2004), leads to ambiguity issues and does not entail semantic meaningfulness, since meaning requires both the definition of a terminology in terms of a logical structure (using symbols) and grounding of symbols to a conceptual level (Cregan, 2007; Nosofsky, 1992).Moreover, while not any situation or situation parameter completely equals another, the description of the complex notion of a specific situation in all its facets is a costly task and may never reach semantic completeness. Apart from that, to enable context-adaptability, a potential infinite set of (real-world) situation characteristics has to be matched to a finite set of semantically defined parameter representations. Therefore, we claim, that fuzzy classification and matchmaking techniques are required to extend and exploit the current functionalities provided by SWS and match the specific requirements of context-aware mobile applications.

[1] DIP Project: http://dip.semanticweb.org

Conceptual Spaces (CS), introduced by Gärdenfors (Gärdenfors, 2000; Gärdenfors, 2004) follow a theory of describing entities at the conceptual level in terms of their natural characteristics similar to natural human cognition in order to avoid the symbol grounding issue. CS enable representation of objects as vector spaces within a geometrical space which is defined through a set of quality dimensions. For instance, a particular color may be defined as point described by vectors measuring the quality dimensions hue, saturation, and brightness. Describing instances as vector spaces where each vector follows a specific metric enables the automatic calculation of their semantic similarity, in terms of their Euclidean distance, in contrast to the costly representation of such knowledge through symbolic SW representations. Even though several criticisms have to be taken into account when utilizing CS (Section 0) they are considered to be a viable option for knowledge representation.

In this chapter, we propose Mobile Situation Spaces (MSS) as a specific derivation of Conceptual Situation Spaces (CSS). MSS utilize CS to represent situations and are mapped to standardized SWS representations to enable first, the situation-aware discovery of appropriate SWS descriptions and finally, the automatic discovery and invocation of appropriate Web services to achieve a given task within a particular situation. Extending merely symbolic SWS descriptions with context information on a conceptual level through MSS enables a fuzzy, similarity-based matchmaking methodology between real-world situation characteristics and predefined SWS representations within mobile environments. Since semantic similarity between situation parameters within a MSS is indicated by the Euclidean distance between them, real-world situation parameters are classified in terms of their distance to predefined prototypical parameters, which are implicit elements of a SWS description. Whereas current SWS technology addresses the issue of allocating services for a given task, our approach supports the discovery of SWS task representations within a given mobile situation. Consequently, the expressiveness of current SWS standards is extended and fuzzy matchmaking mechanisms are supported.

To prove the feasibility of our approach, a proof-of-concept prototype is provided which uses MSS to support context-adaptation by taking into account context parameters such as the current location and desired knowledge subject.

The paper is organized as follows. The following Section 2 provides background information on SWS, whereas Section 3 introduces our approach of Conceptual Situation Spaces which are aligned to current SWS representations. Section 4 illustrates the application of CSS to mobile settings by introducing MSS. Utilizing MSS, we introduce a context-adaptive prototype in Section 5. Finally, we conclude our work in Section 6 and provide an outlook to future research.

SEMANTIC WEB SERVICES AND WSMO

SWS technology aims at the automatic discovery, orchestration and invocation of distributed services for a given user goal on the basis of comprehensive semantic descriptions. SWS are supported through representation standards such as *WSMO* and *OWL-S*. We refer to the *Web Service Modelling Ontology (WSMO)*, a well established SWS reference ontology and framework. The conceptual model of WSMO defines the following four main entities:

- *Domain Ontologies* provide the foundation for describing domains semantically. They are used by the three other WSMO elements. WSMO domain ontologies not only support Web service related knowledge representation but semantic knowledge representation in general.

- *Goals* define the tasks that a service requester expects a Web service to fulfill. In this sense they express the requester's intent.
- *Web service* descriptions represent the functional behavior of an existing deployed Web service. The description also outlines how Web services communicate (*choreography*) and how they are composed (*orchestration*).
- *Mediators* handle data and process interoperability issues that arise when handling heterogeneous systems.

WSMO is currently supported through several software tools and runtime environments, such as the *Internet Reasoning Service IRS-III* (Cabral et al., 2006) and WSMX (WSMX Working Group, 2007). IRS-III is a *Semantic Execution Environment (SEE)* that also provides a development and broker environment for SWS following WSMO. IRS-III mediates between a service requester and one or more service providers. Based on a client request capturing a desired outcome, the goal, IRS-III proceeds through the following steps utilizing the set of SWS capability descriptions:

1. Discovery of potentially relevant Web services.
2. Selection of set of Web services which best fit the incoming request.
3. Invocation of selected Web services whilst adhering to any data, control flow and Web service invocation constraints defined in the SWS capabilities.
4. Mediation of mismatches at the data or process level.

In particular, IRS-III incorporates and extends WSMO as core epistemological framework of the IRS-III service ontology which provides semantic links between the knowledge level components describing the capabilities of a service and the restrictions applied to its use.

However, even though SWS technologies enable the dynamic allocation of Web services for a given goal, it does not consider the adaptation to different user contexts. In order to fully enable context-aware discovery of resources as required by mobile settings (Section 1), the following shortcomings have to be considered:

I1. *Lack of explicit notion of context*: current SWS technology does not entirely specify how to represent domain contexts. For example, WSMO addresses the idea of context: Goal and web service represent the user and provider local views, respectively; the domain ontologies define the terminologies used in each view; and the mediators are the semantic bridges among such distinct views. However, WSMO does not specify what a context description should define and how the context elements should be used.

I2. *Symbolic Semantic Web representations lack grounding to conceptual level*: the symbolic approach, i.e. describing symbols by using other symbols, without a grounding in the real world, of established SWS, and Semantic Web representation standards in general, leads to ambiguity issues and does not entail semantic meaningfulness, since meaning requires both the definition of a terminology in terms of a logical structure (using symbols) and grounding of symbols to a conceptual level (Cregan, 2007; Nosofsky, 1992).

I3. *Lack of fuzzy matchmaking methodologies*: Describing the complex notion of a specific situation in all its facets is a costly task and may never reach semantic completeness. Whereas not any situation and situation parameter completely equals another, the number of (predefined) semantic representations of situations and situation parameters is finite. Therefore, a possibly infinite set of given (real-world) situation characteristics has to be matched to a finite set of predefined parameter instance representations which are described within an IS. Consequently, fuzzy classification and matchmaking techniques are required to classify a real-world situation based on a limited set of predefined parameter descriptions.

CONCEPTUAL SITUATION SPACES

To address the issues I1 - I3 introduced in Section 0, we propose *Mobile Situation Spaces (MSS)* as a setting-specific realisation of our metamodel for *Conceptual Situation Spaces (CSS)* (Dietze, Gugliotta & Domingue, 2008).

CSS Formalisation

CSS enable the description of a particular situation as a member of a dedicated CS. As defined in (Weißenberg et al., 2006) a situation is defined as:

$$S^n = \{(t_1, t_2, cp_1, cp_2, .., cp_n) | cp_i \in CP\}$$

Where t_1 is the starting time of a situation, t_2 represents the end time of a situation and cp_i being situation parameters which are invariant throughout the time interval defined through t_1 and t_2. Referring to (Gärdenfors, 2004; Raubal, 2004), we define a CSS (*css:Conceptual Situation Space* in Figure 1) as a vector space:

$$C^n = \{(c_1, c_2, ..., c_n) | c_i \in C\}$$

with c_i being the quality dimensions *(css:Quality Dimension)* of C. In that, a CSS C represents a particular situation S whereas its situation parameters cp_i are represented through certain quality dimensions c_i. Please note, that we do not distinguish between dimensions and domains - beings sets of integral dimensions (Gärdenfors, 2004) - but enable dimensions to be detailed further in terms of subspaces. Hence, a dimension within one space may be defined through another conceptual space by using further dimensions (Raubal, 2004). In such a case, the particular quality dimension c_j is described by a set of further quality dimensions with

$$c_j = D^n = \{(d_1, d_2, ..., d_n) | d_k \in D\}.$$

In this way, a CSS may be composed of several subspaces and consequently, the description granularity of a specific situation can be refined gradually. To reflect the impact of a specific quality dimension on the entire CSS, we consider a prominence value p *(css:Prominence)* for each dimension. Therefore, a CSS is defined by

$$C^n = \{(p_1c_1, p_2c_2, ..., p_nc_n) | c_i \in C, p_i \in P\}$$

where P is the set of real numbers. However, the usage context, purpose and domain of a particular CSS strongly influence the ranking of its quality dimensions. This clearly supports our position of describing distinct CSS explicitly for specific domains only.

Particular members (*css:Member*) in the CSS are described through a set of valued dimension vectors (*css:Valued Dimension Vectors*). Symbolic representations of domain situations and parameters, such as *css:Situation Description* and *css:Situation Parameter,* refer to particular CSS *(css:Conceptual Situation Space)* whereas parameter instances are represented as members *(css:Member).*

Moreover, referring to Gärdenfors (2004) we consider prototypical members (*css:Prototypical Member*) within a particular space. Prototypical members enable the classification of any arbitrary member *m* within the a specific CSS, by simply calculating the Euclidean distances between *m* and all prototypical members in the same space to identify the closest neighbours of *m*. For instance, given a CS to describe apples based on their shape, taste and colour, a green apple with a strong and fruity taste may be close to a prototypical member representing the typical characteristics of the Granny Smith species. Figure 1 depicts the CSS metamodel.

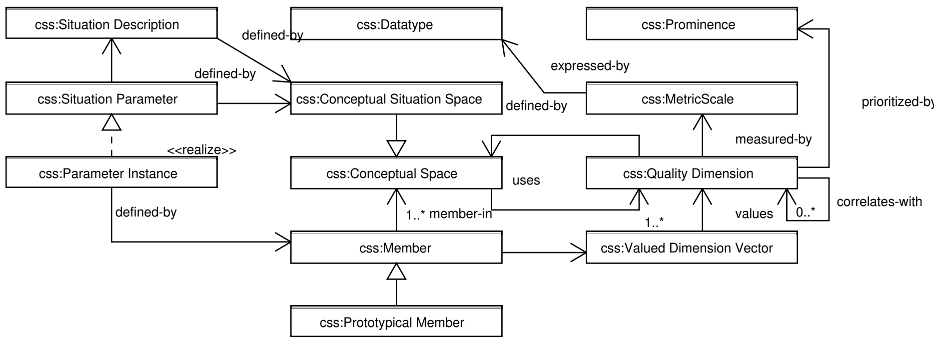

Figure 1. The CSS metamodel.

The metamodel introduced above has been formalized into a *Conceptual Situation Space Ontology (CSSO),* utilizing OCML (Motta, 1998). In particular, each of the depicted entities is represented as a concept within CSSO whereas associations are reflected as their properties in most cases. The correlation relationship indicates whether two dimensions are correlated or not. For instance, when describing an apple the quality dimension describing its sugar content may be correlated with the taste dimension. Information about correlation is expressed within the CSSO through axioms related to a specific quality dimension instance. CSSO is aligned to a well-known foundational ontology: the Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE) (Gangemi, Guarino, Masolo, Oltramari, Schneider, 2002) and, in particular, its module Descriptions and Situations (D&S) (Gangemi, Mika, 2003). The aspect of gradually refining a CSS through subspaces corresponds to the approach of DOLCE D&S to gradually refine a particular description by using parameters where each parameter can be described by an additional description.

With respect to (Raubal, 2004), we define the semantic similarity between two members of a space as a function of the Euclidean distance between the points representing each of the members. However, we would like to point out, that distinct distance metrics, such as the Taxicab or Manhattan distance (Krause, 1987), could be considered, even though the nature of the space and its possible metrics suggests the Euclidean distance as a useful metric to calculate similarities. Applying a formalization of CS proposed in Raubal (2004) to our definition of a CSS, we formalize the Euclidean distance between two members in a CSS as follows. Given a CSS definition *C* and two members represented by two vector sets *V* and *U*, defined by vectors v_0, v_1, …,v_n and u_1, u_2,…,u_n within *C*, the distance between *V* and *U* can be calculated as:

$$\left|d(u,v)\right|^2 = \sum_{i=1}^{n}(z(u_i) - z(v_i))^2$$

where $z(u_i)$ is the so-called Z-transformation or standardization (Devore, Peck, 2001) from u_i. Z-transformation facilitates the standardization of distinct measurement scales which are utilized by different quality dimensions in order to enable the calculation of distances in a multi-dimensional and multi-metric space. The z-score of a particular observation u_i in a dataset is calculated as follows:

$$z(u_i) = \frac{u_i - \bar{u}}{s_u}$$

where $\bar{u}$ is the mean of a dataset U and s_u is the standard deviation from U. Considering prominence values p_i for each quality dimension i, the Euclidean distance $d(u,v)$ indicating the semantic similarity between two members described by vector sets V and U can be calculated as follows:

$$d(u,v) = \sqrt{\sum_{i=1}^{n} p_i \left(\left(\frac{u_i - \bar{u}}{s_u} \right) - \left(\frac{v_i - \bar{v}}{s_v} \right) \right)^2}$$

Utilizing CSS for SWS Selection

Whereas the discovery of distributed Web services for a given user goal is addressed by current SWS technology, such as WSMO, and corresponding reasoners, the context-aware selection of a specific SWS goal representation for a given situation is a challenging task to be tackled when developing SWS-driven applications. By providing an alignment of CSS and SWS, we address this issue by enabling the classification of an individual situation along predefined situation descriptions - used within SWS descriptions - based on semantic similarity calculation. Therefore, CSS are aligned to WSMO to support the automatic discovery of the most appropriate goal representation for a specific situation. Since both metamodels, WSMO as well as CSS, are represented based on the OCML representation language (Motta, 1998), the alignment was accomplished by defining relations between concepts of both ontologies as depicted in Figure 2.

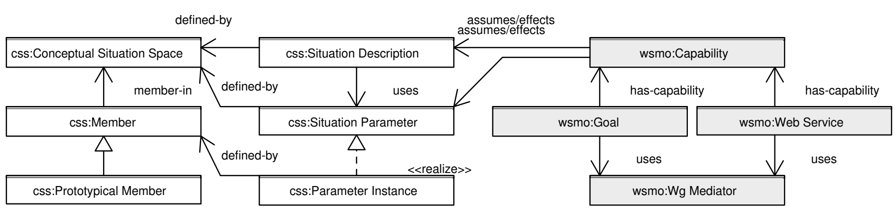

Figure 2. Alignment of CSS and WSMO.

Grey colored concepts in Figure 2 represent concepts of WSMO. A goal description (*wsmo:Goal*) utilizes particular situation parameters (*css:Situation Parameters*) to semantically describe its capabilities, i.e. its assumptions, effects, preconditions and postconditions in terms of semantic situation descriptions *(css:Situation Description)*. A WSMO runtime reasoning engine utilizes capability descriptions to identify SWS (*wsmo:Web Service*) which suit a given Goal. In contrast, the preliminary selection of the most appropriate goal description for a given situation is addressed by classification of situation parameters through CSS. For instance, given a set of real-world situation parameters, described as members in a CSS, their semantic similarity with predefined prototypical parameters *(css:Prototypical Member)* is calculated. Given such a classification of a particular real-world situation, a goal representation which assumes matching prototypical parameter instances is selected and achieved through the reasoning engine.

Deriving CSS for certain Application Contexts

As stated in Gärdenfors (2000), the definition and prioritization of quality dimensions within a CS is highly dependent on the purpose and context of the space. For instance, when describing an apple, dimensions may be differently weighted, dependent on whether the apple is subject to visual cognition exclusively or to full sensory perception, what would be the case if the apple is supposed to be eaten. Whereas in the first case, dimensions such as color and shape are highly ranked, taste and texture may additionally be important in the latter case.

Consequently, the derivation of an appropriate space for a certain purpose is considered an important task which usually should be carried out by a qualified individual such as an application designer. We particularly foresee a procedure consisting of the following steps:

S1. Identification of situation parameters eligible for representation as quality dimension c_i.
S2. Assignment of prominence values p_i to each quality dimension c_i
S3. Assignment of metrics to each quality dimension c_i.

With respect to *S1*, one has to take into account which aspects of a situation are relevant from an application perspective, i.e. which characteristics have an impact on the applied context adaptation strategy or rules. In the case of our intended usage of CSS for SWS selection, only parameters are important, which are considered within SWS capability representations (Section 0).

Since several dimensions might have a different impact factor on the entire space, *S2* is aimed at assigning a prominence value p_i to each dimension c_i. Prominence values should usually be chosen from a predefined value range, such as 0...1. However, since the assignment of prominences to quality dimensions is of major importance for the semantic meaning of calculated distances within a space, this step is not straightforward and most probably requires ex post re-adjustment.

During the final step *S3*, a quantitative metric has to be assigned to each previously defined dimension. Whereas certain dimensions naturally are described using qualitative measurements, such as a size or a weight, other dimensions are usually described using rather qualitative values. The latter applies for instance to the notion of a color. In case no quantitative metric can be assigned to a certain quality dimension c_i, a subspace has to be defined which refines the particular dimension through further dimensions. For instance, in the case of the color dimension, a subspace could be defined using the quantitative dimensions hue, saturation and brightness. Hence, the proposed procedure has to be repeated iteratively until a sufficient description depth has been achieved leading to the definition of a CSS C of the form (Section 0):

$$C^n = \{(p_1 c_1, p_2 c_2, \ldots, p_n c_n) | c_i \in C, p_i \in P\}$$

A MOBILE SITUATION SPACE

Following the steps introduced in Section 0, we derive a CSS aimed at representing situations in mobile settings. A mobile situation is defined by parameters such as the technical environment used by a user, his/her current objectives and particularly the current location. Since each of these parameters apparently is a complex theoretical construct, most of the situation parameters cannot be represented as a single quality dimension within the CSS, but have to be represented as dedicated subspaces which are defined by their very own dimensions (Section 0). Moreover, applying CSS to represent a particular concept is only reasonable in cases where similarity calculation is possible and semantically meaningful, i.e. a particular measurement can be applied to each quality dimension. For instance, the native language of a user is a crucial important situation parameter, but in this case, only a direct match is reasonable in order to provide appropriate information resources in the correct language to the user.

Therefore, this section focuses exemplarily on the representation of two parameters through a CSS subspace, which are of particular interest: the *location* and the *subject* a user is interested in. Due to the complex and diverse nature of a particular subject or spatial location, traditional symbolic representation approaches of the Semantic Web are supposed to fail since it is nearly impossible to define either a subject or a location in a non-ambiguous and comprehensive way by just following a symbolic approach.

Moreover, a one-to-one matchmaking between different locations and subjects is hard to achieve, since fairly not any instance of these parameters completely equals another one. Therefore, fuzzy similarity detections, as enabled through MSS, have to be utilized.

To represent spatial locations, we define a CSS subspace L with 2 quality dimensions l_i representing the latitude and longitude of a particular location

$$L^2 = \{(l_1, l_2) | l_i \in L\}$$

In order to represent a particular subject, we currently consider 4 dimensions (history, geography, culture, languages) which are used to describe the semantic meaning of a particular subject within subspace S:

$$S^4 = \{(s_1, s_2, s_3, s_4) | s_i \in L\}$$

Figure 3 depicts the key concepts of the ontology describing L and S as subspaces (*css:Location Space, css:Subject Space*) within the mobile space *(css:Mobile Situation Space)*.

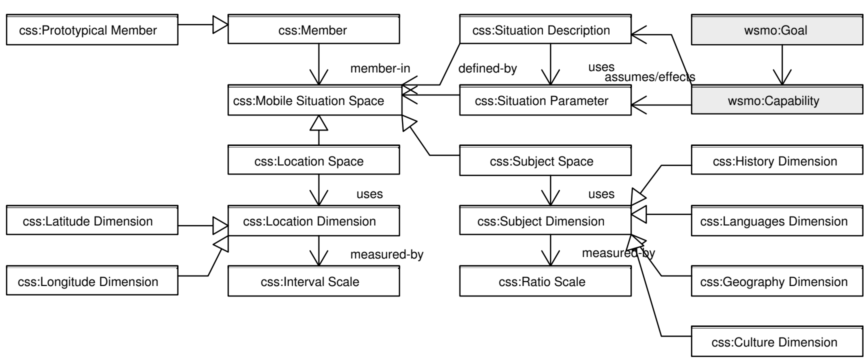

Figure 3. Key concepts representing mobile situation subspaces.

Moreover, Figure 3 depicts the relation of the subspace L (*css:Location Space*) and subspace S (*css:Subject Space*) with WSMO-based SWS descriptions, represented via grey-colored concepts (Section 0).

Instances of a situation parameter representing a subject are defined by particular members within the space S (*css:Subject Space*), which itself uses 4 quality dimension c_i, whereas instances of a parameter representing a spatial location are defined by members within the space L

(*css:Location Space*), which itself uses 4 quality dimension l_i. The metric scale, datatype and value range for each dimension s_i and l_i are presented in Table 1:

Table 1. Metric scale, range, and data type of quality dimensions l_i *and* s_i.

	Quality Dimension	Metric Scale	Data-type	Range
l_1	Latitude	Interval	Float	-90..+90
l_2	Longitude	Interval	Float	-180..+180
s_1	History	Ratio	Float	0..100
s_2	Culture	Ratio	Float	0..100
s_3	Geography	Ratio	Float	0..100
s_4	Language	Ratio	Float	0..100

As depicted in Table 1, each quality dimension l_i is ranked on an interval scale with value ranges being float numbers between -90 and +90 in case of the latitude and between -180 and +180 in case of the longitude. Furthermore, each quality dimension c_i is ranked on a ratio scale with value ranges being float numbers between 0 and 100. The authors would like to highlight, that no prominence values have been assigned since each dimension has an equal impact to define a particular member. It is obvious, that the assignment of prominence values is a highly subjective process, strongly dependent on the purpose, context and individual preferences. Therefore, future work is aimed at enabling users to assign rankings of quality dimensions themselves in order to represent their individual priorities regarding the service retrieval process.

To classify an individual mobile situation, we define prototypical members (*css:Prototypical Member*) in the Mobile Situation Space. For instance, to describe particular cities as members within *L,* we utilized geodata, retrieved from GoogleMaps[2], to describe a prototypical member for each location which is targeted by a particular SWS. A few examples of prototypical location members used in the current prototype application are represented in Table 2:

Table 2. Prototypical members within L.

Prototype	l_1 (Latitude)	l_2 (Longitude)
L1: Milton Keynes (UK)	52.044041	-0.699569
L2: London (UK)	51.500152	-0.126236
L3: Brighton (UK)	50.820931	-0.139846
L4: Paris (FR)	48.85667	2.350987
L5: Toulouse (FR)	43.604363	1.442951

An example of how such parameters are represented in a formal knowledge modeling language is given in Section 0. Moreover, we predefined several prototypical subjects in *S*, each representing the maximum value of a particular quality dimension s_i what resulted in the following 4 prototypical subjects.

[2] http://maps.google.com/.

Table 3. Prototypical members within S.

Prototype	s_1	s_2	s_3	s_4
S1: History	**100**	0	0	0
S2: Culture	0	**100**	0	0
S3: Geography	0	0	**100**	0
S4: Languages	0	0	0	**100**

Apart from the depicted subjects, each subject which is described as part of a symbolic SWS capability representation had been referred to an individual member in *S*.

SIMILARITY-BASED SWS SELECTION AND ACHIEVEMENT IN A MOBILE SETTING

To prove the feasibility of our approach, we provide a proof-of-concept prototype application, which utilizes MSS (Section 4) - based on the CSS metamodel introduced in Sections 0 - and supports context-adaptation in a mobile environment based on SWS and CSS.

Runtime support for CSS and SWS

The following Figure 4 depicts the general architecture adopted to support reasoning on MSS and SWS in distinct domain settings through a Semantic Execution Environment (SEE), which in our case is IRS-III (Section 0).

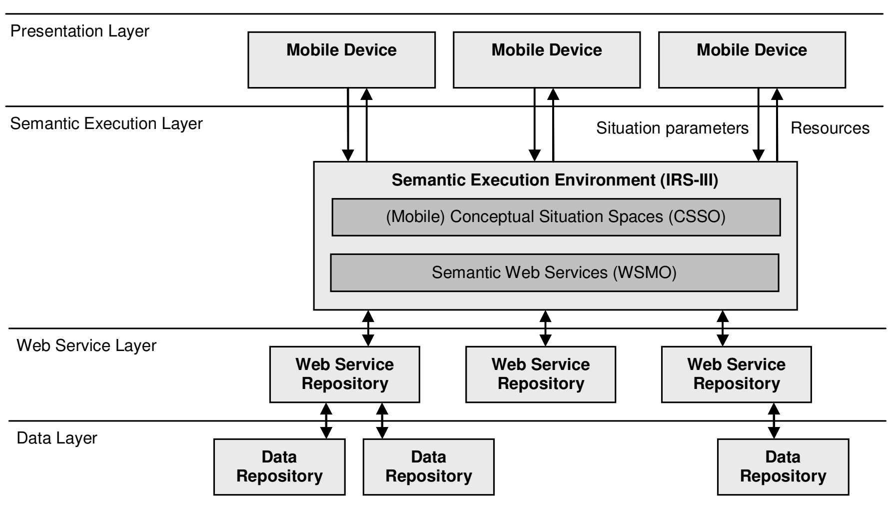

Figure 4. Architecture to support runtime reasoning on CSS and SWS.

Multiple mobile devices - such as PDAs, mobiles or any other portable device hosting a Web browser - can serve as user interface of the SEE, enabling the user (and the device itself) to provide information about his/her goal and the current real-world situation.

The SEE makes use of semantic representations of the CSS formalisation (CSS ontology, CSSO), specifically derived for mobile settings, and of SWS annotations based on WSMO in order to discover and allocate the most appropriate resource for a given user goal within a current situation. Ontologies had been represented using the OCML knowledge modeling language (Motta, 1998).

WSMO capabilities are represented by defining the assumptions and effects of available SWS and goals in terms of certain situation description or situation parameter instances (Section 0). Such situation descriptions are refined as particular prototypical members of an associated CSS, such as prototypical members of the MSS S and L introduced in Section 4.

As mentioned in Section 3, CSSO allows us to describe a specific mobile situation description instance in terms of a collection of situation parameter instances. Mobile situation description instances are automatically and gradually defined at runtime by the SEE as the result of the user interaction with the mobile device. On the basis of the detected context parameters, the SEE performs the following steps:

(i) Computation of similarities between the detected real-world context parameters – obtained from the user and its device - and symbolic representation of prototypical situation parameters;
(ii) Progressive update of the current mobile situation description with the closest prototypical situations parameters;
(iii) Determination of (WSMO) goal matching the refined situation description;
(iv) Achievement of selected goals by means of discovery and orchestration of available web services.

Consequently, we enable the classification of real-world context parameters along available predefined parameters in order to enable a similarity-based selection and orchestration WSMO goals.

Context classification and adaptation

As outlined in the previous section, the SEE automatically detects the semantic similarity of specific situation parameters with a set of predefined prototypical parameters to enable the allocation of context-appropriate resources. In this section, we further detail these aspects, since they are central in the contribution of this chapter. In particular, we specify the concepts of classification and adaptation.

Referring to CSS subspaces *L* and *S* described in Section 0, given a particular member *U* in *L* or *S*, its semantic similarity with each of the prototypical members is indicated by their Euclidean distance. Since we utilize spaces described by dimensions which each use the same metric scale and no prominence value, the distance between two members *U* and *V* can be calculated disregarding a Z-transformation (Section 0) for each vector:

$$d(u,v) = \sqrt{\sum_{i=1}^{n} (u_i - v_i)^2}$$

Please note, that it would be possible to calculate distances either between entire situations (members within *css:Mobile Situation Space*) or between particular parameter (members in subspaces such as *L* and *S*). Since individual semantic similarities between instances of parameters such as the current location or the desired subject are usually important knowledge when deciding about the appropriateness of resources for a given context, the current application calculates distances between each parameter, i.e. between members within each individual subspace.

The calculation of Euclidean distances using the formula shown above is performed by a standard Web service, which is annotated as SWS and invoked through IRS-III at runtime. Given a particular CSS description, a member (representing a specific parameter instance) as well as a set of prototypical member descriptions (representing prototypical parameter instances), similarities are calculated by the Web service at runtime in order to classify a given situation parameter.

For instance, a user is currently located in Eastbourne (UK) and is interested in historical information about the surrounding area. Consequently, the particular situation description (*css:MobileSituation Desccription*) includes a location parameter which is defined by a member *E* in the specific location space (*css:Location Space*) with the following vectors describing latitude and longitude of Eastbourne:

$$E = \{(e_1 = 50.766868, e_2 = 0.284804) | e_i \in L\}$$

To represent the current aim of the user, a user selects one of the subject prototypes (Section 0), in this case *S1* (Table 3), which is added to the situation description.

Figure 5 depicts a screenshot of a mobile device showing the application web-interface while supporting a user to semi-automatically locate him-/herself utilizing geodata dynamically retrieved from GoogleMaps. By providing incomplete knowledge about the current location, for instance the current city, full geospatial datasets, including the latitude and longitude of a location, are retrieved dynamically to enable similarity-based location matchmaking.

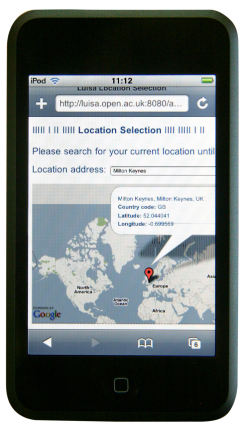

Figure 5. Mobile device showing semi-automatic location detection.

Based on the current situation description, SWS are selected which are able to address the situation. Whereas parameters which are not defined by members in a specific CSS require a direct match with a corresponding SWS description, a similarity-based match is computed for parameters which are described in a CSS, e.g. the location or the subject. Hence, distance calculation was utilized to identify similarities between current context parameters – such as *E* and *S1* – and prototypical parameters which had been defined as part of SWS capability descriptions in order to represent the parameters targeted by available SWS. In order to illustrate the representation of prototypical CSS members, the following OCML code defines a location parameter instance representing the geospatial location Brighton, as well as the respective prototypical member (*L3*) in the MSS *L*.

```
(def-instance brighton-location location
    ((has-instance-title "Brighton")
    (defined-by p2-location-brighton)))

(def-instance p2-location-brighton location-prototypical-member
    ((has-title "Location-Brighton ")
    (has-description "Prototype describing Brighton")
    (member-in location-space)
    (has-valued-dimension (brighton-valued-lat-vector brighton-valued-long-
    vector))))

(def-instance brighton-valued-lat-vector location-valued-dimension-vector
    ((values latitude-dimension)
     (has-value 50.820931)))

(def-instance brighton-valued-long-vector location-valued-dimension-vector
    ((values longitude-dimension)
     (has-value -0.139846)))
```

Listing 1. Partial OCML code defining location parameter instance and respective MSS member.

Calculating distances between *E* and targeted locations – represented as prototypical MSS members - led to the identification of the following distances to the three closest matches:

Table 4. Distances between E and targeted locations.

Prototype	Euclidean Distance
L1: Milton Keynes (UK	1.6125014961413195
L2: London	0.8406303029608179
L3: Brighton	0.42807759865356176

Since not any SWS targets historical interests (*S1*) exclusively – as desired by the user - no direct match between the situation and subjects targeted by available SWS was achieved. However, similarity calculation identified related subject areas, which partially target historical information. Table 5 indicates their vectors and distances to the required subject *S1*.

Table 5. Distances between S1 and targeted subjects.

Subject	Euclidean Distance
S5 (50,0,50,0)	70.71067811865476
S6 (65,0,0,35)	49.49747468305833
S7 (70, 30,0, 0)	35.35533905932738

The subjects *S5*, *S6* and *S7* as well as the locations *L1*, *L2*, and *L3* shown in Table 4 and Table 5 had been described as prototypical members in the MSS (Section 0) during the development of SWS representations targeting certain subjects and locations. By following our alignment from Section 0, this task could be performed by either the Web service provider or any SWS expert who is providing and publishing a semantic representation of available Web services.

As indicated by the Euclidean distances depicted in Tables 4 and 5, the closest matching SWS provides historical and cultural (*S7*) resources for the Brighton (*L3*) area, as these show the lowest distances. Provided these similarities, a user is able to select predefined parameters that best suit his/her specific preferences within the current situation. In that, the use of similarity-based classification enables the gradual refinement of a situation description and fuzzy matchmaking between real-world situations, and prototypical parameters predefined within a SWS description. For example, the following OCML code defines the partial capability description of a Web service that provides historic and cultural information for the area of Brighton:

```
(def-class lpmo-get-brighton-his-and-cult-LOs-ws-capability (capability)
?capability
((used-mediator :value lpmo-get-brighton-his-and-cult-LOs-mediator)
     (has-assumption :value
           (KAPPA (?web-service)
           (and (= (get-location (wsmo-role-value ?web-service 'has-
           situation)) " Brighton"))
           (= (get-subject (wsmo-role-value ?web-service 'has-situation))
           "S7")))))
```

Listing 2. Partial OCML code representing SWS capability in terms of assumed MSS members.

In fact, the assumption expression presented above describes that situation description representing the current situation (*has-situation*) consider the location *Brighton* and the subject *S7*.

As a result, in our approach, the actual mobile situation description (i.e. the actual context) is the result of an iterative process that involves several distance calculations to map symbolic representations and real world characteristics. Notice that this process actively involves the end users in providing observables and validating the distance calculations. According to the obtained situation parameters and the selected user goal, the SEE discovers and orchestrates annotated Web services, which show the capabilities to suit the given situation representation. Whereas discovery and orchestration are addressed by existing SWS technology, the context-aware selection of a specific SWS goal representation is addressed through CSS by enabling similarity-based classifications of individual situations as described in the previous sections.

RELATED WORK

Since our work relates to several different but related research areas, we report here related work on (i) Semantic Web Services, (ii) Context-adaptive systems, and (iii) Context-adaptation in mobile environments. Moreover, by comparing our approach with related work in (iii) we describe our contribution to the current state of the art in context-adaptive mobile and ubiquitous computing.

SWS: OWL-S (OWL-S Coalition. 2004) is a comparatively narrow framework and ontology for adding semantics to Web service descriptions. In order to identify problematic aspects of OWL-S and suggest possible enhancements, a contextualized core ontology of services has been described in Mika et al. (2004). Such an ontology is based on DOLCE (Gangemi et al., 2002) and its specific module D&S (Gangemi, Mika, 2003). Even though we followed a similar approach, we adopt WSMO (WSMO Working Group, 2004) instead of OWL-S as reference ontology for SWS. Moreover, the aim of our resulting ontology is not proposing changes to WSMO, but creating domain-specific models which incorporate WSMO-based SWS representations.

Context-adaptive systems: in Bouquet et al. (2003) the authors define contexts as the local models that encode a party's view of a domain. They distinguish contexts from ontologies, since the latter are shared models of some domain that encode a view which is common to a set of different parties. Contexts are best used in those applications where the core problem is the use and management of local and autonomous representations with a need for a lack of centralized control. For example, the notion of contexts is used in some applications of distributed knowledge management Bonifacio et al. (2003), pervasive computing environments (Chen, Finin & Joshi, 2003) and peer-to-peer applications (Serafini et al., 2003). According to the definition introduced in Bouquet et al. (2003), we propose a novel use of contexts. The local models encode party's view of SWS-based process descriptions.

Context-adaptation in mobile environments: Weissenberg et al. (2006) adopt an approach to context-adaptation in mobile settings which shows some similarities to ours: given a set of context parameters – based on sensor data – first a context is identified and then a matching situation. However, they rely on manually predefined axioms which enable such a reasoning compared to the automatic detection as proposed in this paper. Korpipaa et al. (2003) propose a related framework but firstly, require client-side applications to be installed and, secondly, relies on Bayesian reasoning for matching between measured lower-level contexts and higher-level context abstractions represented within an ontology. Hence, as a major lack, it is required to provide information about contexts and their relations within a Bayesian Network in order to perform the proposed reasoning. Gu, Wang, Pung & Zang (2004) propose a context-aware middleware which also distinguishes between lower-level and higher-level contexts. However, there is no mechanism to automatically identify relationships between certain contexts or context parameters. The same criticism applies to the approaches to a semantic representation of user contexts described in Toivinen, Kolari & Laako (2003) and Sathish, Pavel & Trossen (2006).

Finally, it can be highlighted, that current approaches to context-adaptation in mobile settings usually rely on the manual representation of mappings between a given set of real-world context data and predefined context representations. Since this approach is costly and time-consuming, our approach could contribute there significantly by providing a similarity-based and rather fuzzy method for automatically identifying appropriate symbolic context representations given a set of detected context parameters.

CONCLUSIONS

In this paper, we proposed an approach to support fuzzy, similarity-based matchmaking between real-world situation parameters in mobile settings and predefined semantic situation descriptions by incorporating semantic context information on a conceptual level into symbolic SWS descriptions based on Conceptual Situation Spaces. Given a particular mobile situation, defined by parameters such as the location and device of the user, the most appropriate resources, whether data or services, are discovered based on the semantic similarity, calculated in terms of the Euclidean distance, between the real-world situation and predefined resource descriptions as part of SWS representations. Even though we refer to the SWS framework WSMO in this paper, we would like to highlight, that our approach could be applied to other SWS reference ontologies such as OWL-S (OWL-S Coalition. 2004). Consequently, by aligning CSS to established SWS technologies, the expressiveness of symbolic SWS standards is extended with context information on a conceptual level described in terms of natural quality dimensions to enable fuzzy context-aware delivery of information resources at runtime. Whereas current SWS frameworks address the allocation of distributed services for a given (semantically) well-described task, Mobile Situation Spaces particularly address the similarity-based discovery of the most appropriate SWS task representation for a given situation. To prove the feasibility of our approach, a proof-of-concept prototype application was presented, which applies the MSS to enable context-adaptive resource discovery in a mobile setting.

However, although our approach applies CS to solve SWS-related issues such as the symbol grounding problem, several criticisms still have to be taken into account. Whereas defining situational contexts, respectively members within a given MSS, appears to be a straightforward process of assigning specific values to each quality dimension, the definition of the MSS itself is not trivial at all and strongly dependent on individual perspectives and subjective appraisals. Whereas the semantics of an object are grounded to metrics in geometrical vector spaces within a MSS, the quality dimensions itself are subject to ones perspective and interpretation what may lead to ambiguity issues. With regard to this, MSS do not appear to solve the symbol grounding issue but to shift it from the process of describing instances to the definition of a MSS. Moreover, distinct semantic interpretations and conceptual groundings of each dimension may be applied by different individuals. Apart from that, whereas the size and resolution of a MSS is indefinite, defining a reasonable space for a specific domain and purpose may become a challenging task. Nevertheless, distance calculation as major contribution of the MSS approach, not only makes sense for quantifiable parameters but also relies on the fact, that parameters are described in the same geometrical space.

Consequently, CS-based approaches, such as MSS, may be perceived as step forward but do not fully solve the issues related to symbolic Semantic Web (Services)-based knowledge representations. Hence, future work has to deal with the aforementioned issues. For instance, we foresee to enable adjustment of prominence values to quality dimensions of a specific space to be accomplished by a user him/herself, in order to most appropriately suit his/her specific priorities and preferences regarding the resource allocation process, since the prioritization of dimensions is a highly individual and subjective process. Nevertheless, further research will be concerned with the application of our approach to further domain-specific situation settings.

REFERENCES

Bouquet P, Giunchiglia F, van Harmelen F, Serafini L, Stuckenschmidt H (2003) C-OWL: Contextualizing Ontologies. ISWC-2003, LNCS vol. 2870, Springer Verlag , pp. 164-179.

Bonifacio M, Bouquet P, Mameli G, Nori M (2003). Peer-mediated knowledge management. In AAAI-03 Spring Symposium on Agent-Mediated Knowledge Management (AMKM-03).

Cabral, L., Domingue, J., Galizia, S., Gugliotta, A., Norton, B., Tanasescu, V., Pedrinaci, C. (2006): IRS-III: A Broker for Semantic Web Services based Applications. Proceedings of the 5th International Semantic Web Conference (ISWC), Athens, USA.

Chen H, Finin T, Joshi A (2003). An Ontology for Context Aware Pervasive Computing Environments. The Knowledge Engineering Review, vol. 18 pp. 197-207.

Cregan, A. (2007), Symbol Grounding for the Semantic Web. 4th European Semantic Web Conference 2007, Innsbruck, Austria.

Devore, J., and Peck, R. (2001), *Statistics - The Exploration and Analysis of Data*, 4th ed. Pacific Grove, CA: Duxbury, 2001.

Dietze, S., Gugliotta, A., Domingue, J., (2007) A Semantic Web Services-based Infrastructure for Context-Adaptive Process Support. Proceedings of IEEE 2007 International Conference on Web Services (ICWS), Salt Lake City, Utah, USA.

Dietze, S., Gugliotta, A., Domingue, J., (2008) Towards Context-aware Semantic Web Service Discovery through Conceptual Situation Spaces. Workshop: International Workshop on Context enabled Source and Service Selection, Integration and Adaptation (CSSSIA), 17th International World Wide Web Conference (WWW2008), Beijing, China.

Fensel, D., Lausen, H., Polleres, A., de Bruijn, J., Stollberg, M., Roman, D., Domingue, J. (2006): Enabling Semantic Web Services – The Web service Modelling Ontology, Springer 2006.

Gangemi, A., and Mika, P. (2003), Understanding the Semantic Web through Descriptions and Situations. In Meersman, R.; Tari, Z.; and et al., D. S., eds., Proceedings of the On The Move Federated Conferences (OTM'03), LNCS. Springer Verlag.

Gangemi, A., Guarino, N., Masolo, C., Oltramari, A., Schneider, L.(2002), Sweetening Ontologies with DOLCE. In: A. Gómez-Pérez , V. Richard Benjamins (Eds.) Knowledge Engineering and Knowledge Management. Ontologies and the Semantic Web: 13th International Conference, EKAW 2002, Siguenza, Spain, October 1-4, 2002.

Gärdenfors, P. (2000), Conceptual Spaces - The Geometry of Thought. MIT Press, 2000.

Gärdenfors, P. (2004), How to make the semantic web more semantic. In A.C. Vieu and L. Varzi, editors, Formal Ontology in Information Systems, pages 19–36. IOS Press, 2004.

Gellersen, H-W., Schmidt, A. and Beigl, M. (2002), Multi-Sensor Context-Awareness in Mobile Devices and Smart Artefacts. ACM Journal Mobile Networks and Applications (MONET), Vol 7, No 5, Oct. 2002.

Gu, T., Wang, X.H., Pung, H.K., and Zhang, D.Q. (2004), A middleware for context-aware mobile services, in IEEE Vehicular Technology Conference (VT)C, IEEE Computer Society Press: Los Alamitos, CA, USA, 2004.

Joint US/EU ad hoc Agent Markup Language Committee (2004), OWL-S 1.1 Release. http://www.daml.org/services/owl-s/1.1/.

Krause, E. F. (1987). Taxicab Geometry. Dover.

Korpipaa, P., Mantyjarvi, J., Kela, J., Keranen, H., Malm, E. (2003), Managing Context Information in Mobile Devices, IEEE Pervasive Computing, vol. 02, no. 3, pp. 42-51, Jul-Sept, 2003.

Mika P, Oberle D, Gangemi A, Sabou M (2004). Foundations for Service Ontologies: Aligning OWL-S to DOLCE, WWW04.

Motta, E. (1998). An Overview of the OCML Modelling Language, the 8th Workshop on Methods and Languages, 1998.

Nosofsky, R. (1992), Similarity, scaling and cognitive process models, Annual Review of Psychology 43, pp. 25- 53, (1992).

OWL-S Coalition: OWL-S 1.1 release. (2004). http://www.daml.org/services/owl-s/1.1/

Raubal, M. (2004) Formalizing Conceptual Spaces. in: A. Varzi and L. Vieu (Eds.), Formal Ontology in Information Systems, Proceedings of the Third International Conference (FOIS 2004).*Frontiers in Artificial Intelligence and Applications* 114, pp. 153-164, IOS Press, Amsterdam, NL.

Sailesh Sathish, Dana Pavel and Dirk Trossen (2006), Context Service Framework for Mobile Internet, International Worskshop on System Support for Future Mobile Computing Applications (FUMCA 2006), September 2006, Irvine, California, USA

Serafini L, Giunchiglia F, Mylopoulos J, Bernstein P (2003). Local relational model: a logical formalization of database coordination. In P. Blackburn, C. Ghidini, and R. Turner editors, Context'03.

Schmidt, A., Winterhalter, C. (2004), User Context Aware Delivery of E-Learning Material: Approach and Architecture, Journal of Universal Computer Science (JUCS) vol.10, no.1, January 2004.

Toivonen, S., Kolari, J., and Laakko, T. (2003) Facilitating mobile users with contextualized content, in Proc. Workshop Artificial Intelligence in Mobile Systems, 2003.

Weißenberg, N., Gartmann, R., and Voisard, A. (2006), An Ontology-Based Approach to Personalized Situation-Aware Mobile Service Supply. Geoinformatica 10, 1 (Mar. 2006), 55-90. DOI= http://dx.doi.org/10.1007/s10707-005-4886-9.

World Wide Web Consortium, W3C (2004a): Resource Description Framework, W3C Recommendation 10 February 2004, http://www.w3.org/RDF/.

World Wide Web Consortium, W3C (2004b): Web Ontology Language Reference, W3C Recommendation 10 February 2004, http://www.w3.org/TR/owl-ref/.

WSMO Working Group (2004), D2v1.0: Web service Modeling Ontology (WSMO). WSMO Working Draft, (2004). (http://www.wsmo.org/2004/d2/v1.0/).

WSMX Working Group (2007), The Web Service Modelling eXecution environment, http://www.wsmx.org/.