

# Enriching Service Semantics through Conceptual Vector Spaces

Stefan Dietze<sup>1</sup>, John Domingue<sup>1</sup>

<sup>1</sup> Knowledge Media Institute,  
Open University,  
MK7 6AA, Milton Keynes, UK  
{s.dietze, j.b.domingue}@open.ac.uk

**Abstract.** Semantic Web Services (SWS) aim at the automated discovery and orchestration of Web services on the basis of comprehensive, machine-interpretable semantic descriptions. In that, SWS strive for automated interoperability and reusability of heterogeneous services through matchmaking of semantic capability and interface descriptions. However, to do so, established SWS reference models build on the general assumption that either (a) SWS providers subscribe to a common vocabulary to annotate their services or (b) alignments between distinct vocabularies are established. This is due to the fact that SWS descriptions are lacking sufficient meaningfulness to automatically infer relationships between syntactically different semantic annotations. In order to address these issues and to overcome the need for (a) and (b), we propose a representational approach which allows to enrich standard SWS descriptions through vector spaces, which are represented as a dedicated ontology being aligned with existing SWS standards. As a result, similarities between instances used to annotate SWS become automatically computable by means of spatial distances. Hence, our approach significantly contributes to solve the interoperability problem between heterogeneous SWS as well as SWS reference models.

**Keywords:** Semantic Web Services, Interoperability, Vector Spaces.

## 1 Introduction

The ongoing shift to service-orientation in software development leads to an increasing availability of a broad variety of Web services, ranging from SOAP-based ones to rather light-weight approaches based on REST [9] or XML-RPC [25]. This raises the need to automatically discover and orchestrate appropriate services for a given need. *Semantic Web Services (SWS)* [8] aim at addressing this challenge on the basis of comprehensive, machine-interpretable semantic descriptions. Since Web services usually are provided by distinct and independent parties, the actual Web service interfaces as well as their semantic representations are highly heterogeneous. This strongly limits the interoperability and re-usability of services. In order to cope with heterogeneity, established SWS reference models such as WSMO [26], OWL-S

[16] or SAWSDL<sup>1</sup> build upon the assumption, that either (a) SWS providers subscribe to a common vocabulary to annotate their services or (b) alignments between distinct vocabularies used by different SWS are established while somehow automatic mediation approaches are still limited and underdeveloped [17]. This is due to the fact that SWS descriptions are lacking sufficient meaningfulness to automatically infer relationships – particularly *semantic similarity* [1] relationships – between independent and syntactically different semantic annotations, such as concepts and instances which are part of different SWS. However, since this is a fundamental requirement to enable matchmaking across heterogeneous SWS [22][27], large-scale interoperability is not facilitated.

In this paper, we propose a representational approach which enriches the expressiveness of SWS approaches with formal representations following the *Conceptual Spaces (CS)* [10] approach. In particular, we propose an ontology which is aligned to SWS reference models and facilitates a grounding of SWS descriptions into multiple vector spaces. We will demonstrate that refining heterogeneous SWS descriptions in multiple shared CS supports computation of semantic similarities and implicitly facilitates matchmaking and discovery of heterogeneous SWS.

The remainder of the paper is organized as follows: Section 2 introduces the SWS matchmaking problem, while our representational approach based on refinement of SWS ontologies in CS is proposed in Section 3. In Section 4, we introduce application of our approach to an existing SWS reference model. Finally, we discuss and conclude our work in Section 6.

## 2 Semantic Web Services and the Matchmaking Problem

We report below some abstract definitions of SWS as used throughout the remainder of the paper, together with background information on current matchmaking and mediation approaches.

**Semantic Web Services:** a SWS description (either the description of the Web service or the description of the service request) is formally represented within a particular ontology that complies with a certain SWS reference model such as OWL-S [16] or WSMO [26]. By adopting a common formalisation of an ontology [6], we define a populated *service ontology*  $O$  – as utilised by a particular SWS representation – as a tuple:

$$O = \{C, I, P, R, A\} \subset SWS$$

With  $C$  being a set of  $n$  *concepts* where each concept  $C_i$  is described through  $l(i)$  *concept properties*  $pc$ .  $I$  represents all  $m$  *instances* where each instance  $I_{ij}$  represents a particular instance of a concept  $C_j$  and consists of  $l(i)$  *instantiated properties*  $pi$  instantiating the concept properties of  $C_j$ . Hence, the properties  $P$  of an ontology  $O$  represent the union of all concept properties  $PC$  and instantiated properties  $PI$  of  $O$ .

Given these definitions, we would like to point out that properties here exclusively refer to so-called data type properties. Hence, we define properties as being distinctive to relations  $R$ . The latter describe relations between concepts and instances.

---

<sup>1</sup> <http://www.w3.org/2002/ws/sawsdl/spec/>

In addition,  $A$  represents a set of *axioms* which define constraints on the other introduced notions. Since certain parts of a SWS ontology describe certain aspects of the Web service (request), such as its capability  $Cap$ , interface  $If$  or non-functional properties  $Nfp$  [4], a SWS ontology can be perceived as a conjunction of ontological subsets:

$$Cap \cup If \cup Nfp \subset SWS$$

The semantic capability description consists of further subsets, describing the assumptions  $As$ , effects  $Ef$ , preconditions  $Pre$  and postconditions  $Post$ . However, given the lack of a clear distinction between assumption/effect and pre-/postcondition, we prefer the exclusive usage of assumptions/effects:

$$As \cup Ef = Cap \subset O \subset SWS$$

**SWS discovery as a similarity computation problem:** SWS discovery across distributed SWS requires semantic level mediation, i.e. the mediation between heterogeneous SWS descriptions to overcome the need for either manual mappings or the subscription to a common vocabulary. That is perceived to be a fundamental requirement to further exploit SWS approaches on a Web scale. SWS discovery requires to identify SWS which are best suitable to satisfy a certain request. In that, in order to identify whether a particular SWS  $S_1$  is potentially relevant for a given request  $S_2$ , a SWS broker has to compare the capabilities of  $S_1$  and  $S_2$ , i.e. it has to identify whether the following holds true:

$$As_2 \subset As_1 \cup Ef_2 \subset Ef_1$$

However, in order to compare distinct capabilities of available SWS which each utilise a distinct vocabulary, these vocabularies have to be aligned. For instance, to compare whether an assumption expression  $As_1 \equiv \neg I_1 \cup I_2$  of one particular SWS $_1$  is the same as  $As_2 \equiv I_3 \cup \neg I_4$  of another SWS $_2$ , where  $I_i$  represents a particular instance, matchmaking engines have to perform two steps: (a) identification of relationships between concepts/instances involved in distinct SWS representations; (b) evaluation whether the semantics of the two SWS expressions match each other. Whereas current SWS execution environments exclusively focus on (b), SWS discovery also requires mediation between different ontologies, as in (a), and could also be perceived as a particular instantiation of the *ontology mapping* problem [27][3]. I.e. following [6] the goal is, to establish formal relations between a set of knowledge entities  $E_1$  from an ontology  $O_1$  – used to represent a particular SWS  $S_1$  – with entities  $E_2$  which represent the same or a similar semantic meaning in a distinct ontology  $O_2$  (SWS  $S_2$ ). In that, SWS discovery strongly relies on identifying *semantic similarities* [1] between entities across different SWS ontologies. Hence, the identification of similarities is a necessary requirement to solve the discovery problem for multiple heterogeneous SWS representations [27][21]. However, while similarity detection across distinct SWS representations requires semantic meaningfulness, the symbolic approach – i.e. describing symbols by using other symbols, without a grounding in the real world – of established SWS representation standards, leads to ambiguity issues and does not fully entail semantic meaningfulness [5][14]. Moreover, describing the complex notion of specific SWS capabilities in all their facets is a costly task and may never reach semantic completeness.

Given the lack of inherent similarity representation, current approaches to ontology mapping could be applied to facilitate SWS mediation. These approaches aim at semi-

automatic similarity detection across ontologies mostly based on identifying linguistic commonalities and/or structural similarities between entities of distinct ontologies [3][7][15]. However, such approaches require manual intervention, are costly and error-prone, and hence, similarity-computation remains as central challenge. In our vision, instead of semi-automatically formalising individual mappings or subscribing to common vocabularies, methodologies to automatically compute or implicitly represent similarities across distinct SWS representations are better suited to facilitate SWS interoperability.

### 3 Enriching Service Semantics through Conceptual Vector Spaces

To overcome the issues introduced in the previous Section, we propose a representational approach enriching SWS representations through multiple vector spaces following the *Conceptual Spaces (CS)* [9] theory. CS represent entities in terms of their quality characteristics similar to natural human cognition in order to bridge between the neural and the symbolic world [9]. In that, CS are represented through multidimensional geometrical vector spaces where instances are supposed to be represented as vectors, i.e. particular points in a CS. Describing instances as vectors which each vector follow a specific metric enables the automatic calculation of their semantic similarity by means of distance metrics such as the Euclidean, Taxicab or Manhattan distance [12] or the Minkowsky Metric [22]. However, CS do not provide any notion to represent any arbitrary relations [20], such as *part-of* relations which usually are represented within symbolic knowledge models.

#### 3.1 Conceptual Groundings for SWS

We propose a representational approach which combines symbolic SWS representation with groundings in multiple CS (Figure 1) to enable the implicit representation of semantic similarities across heterogeneous SWS representations provided by distinct agents.

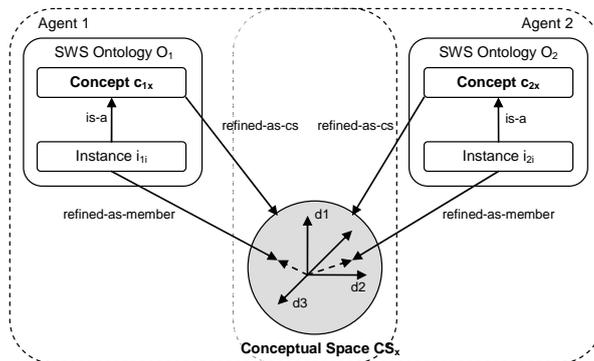


Fig. 1. Representing heterogeneous SWS representations through shared Conceptual Spaces.

Hence, we facilitate similarity based mediation at the semantic level and consequently support the SWS discovery task. Whereas CS allow the representation of semantic similarity as a notion implicit to a constructed knowledge model, it can be argued, that representing an entire SWS through a coherent CS might not be feasible, particularly when attempting to maintain the meaningfulness of the spatial distance as a similarity measure. Therefore, we claim that CS are a particularly promising model when being applied to individual concepts – as part of SWS descriptions – instead of representing an entire ontology in a single CS. In that, we would like to highlight that we consider the representation of a set of  $n$  concepts  $C$  of a SWS ontology  $O$  through a set of  $n$  CS (Figure 1). Hence, instances of concepts are represented as members (i.e. vectors) in the respective CS. While still taking advantage from implicit similarity information within a CS, our hybrid approach – combining SWS descriptions with multiple CS – allows overcoming CS-related issues by maintaining the advantages of ontology-based SWS representations.

Please note that our approach relies on the agreement on a common set of CS for a given set of distinct SWS ontologies, instead of a common agreement on the ontologies themselves. Hence, whereas in the latter case two agents have to agree on a common ontology at the concept and instance level, our approach requires just agreement at the concept level, since instance similarity becomes an implicit notion. Moreover, we assume that the agreement on ontologies at the concept level becomes an increasingly widespread case, due to, on the one hand, increasing use of upper-level ontologies such as DOLCE [11], SUMO [23] or OpenCyc<sup>2</sup> which support a certain degree of commonality between distinct ontologies. On the other hand, SWS ontologies often are provided within closed environments, for instance, virtual organisations, where a common agreement to a certain extent is ensured. In such cases, the derivation of a set of common CS is particularly applicable and straightforward.

In order to refine and represent SWS descriptions within a CS, we formalised the CS model into an ontology (CSO)<sup>3</sup>, currently being represented through OCML [13]. The ontology enables the instantiation of a set of CS to represent a given set of concepts as part of SWS descriptions. Referring to [19], we formalise a CS as a vector space defined through quality dimensions  $d_i$  of CS. Each dimension is associated with a certain metric scale, e.g. ratio, interval or ordinal scale. To reflect the impact of a specific quality dimension on the entire CS, we consider a prominence value  $p$  for each dimension [19]. Therefore, a CS is defined by

$$CS^n = \{(p_1d_1, p_2d_2, \dots, p_nd_n) \mid d_i \in CS, p_i \in \mathfrak{R}\}.$$

However, usage context, purpose and domain of a particular CS strongly influence the ranking of its quality dimensions. This clearly supports our position of describing distinct CS explicitly for individual concepts. Please note that dimensions could be detailed further in terms of subspaces. Hence, a dimension within one CS may be defined through another CS by using further dimensions. In such a case, the particular quality dimension  $d_j$  is described by a set of further quality dimensions. In this way, a CS may be composed of several subspaces and consequently, the description

---

<sup>2</sup> <http://www.opencyc.org/>

<sup>3</sup> <http://people.kmi.open.ac.uk/dietze/ontologies/conceptual-spaces.lisp>

granularity can be refined gradually. Furthermore, dimensions may be correlated what is expressed through axioms related to a specific quality dimension instance.

A member  $M$  – representing a particular instance – of the CS is described through a set of valued dimension vectors  $v_i$ :

$$M^n = \{(v_1, v_2, \dots, v_n) | v_i \in M\}$$

With respect to [19], 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, different distance metrics, such as the Taxicab or Manhattan distance [12], could be considered, dependent on the nature and purpose of the CS. Given a CS definition  $CS$  and two members  $V$  and  $U$ , defined by vectors  $v_0, v_1, \dots, v_n$  and  $u_1, u_2, \dots, u_n$  within  $CS$ , the distance between  $V$  and  $U$  can be calculated as:

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

where  $\bar{u}$  is the mean of a dataset  $U$  and  $s_u$  is the standard deviation from  $U$ . The formula above already considers the so-called Z-transformation or standardization [22] which facilitates the standardization of distinct measurement scales utilised by different quality dimensions in order to enable the calculation of distances in a multi-dimensional and multi-metric space.

### 3.2 Representing SWS Capabilities through Conceptual Spaces

Following our vision, the provisioning of SWS representations is a highly heterogeneous and distributed procedure that is accomplished autonomously by distinct agents. In particular, we distinguish two groups of involved agents: (C1) *distributed SWS providers and consumers* and (C2) *centralised SWS maintainers*. The existence of C2 is implied by the broker-based nature of SWS technologies.

Specifically, the overall procedure of providing SWS following our approach is based on the following steps:

- S1. Provisioning of a central SWS runtime environment (C2).
- S2. Provisioning of SWS representations  $S^n$  (C1).
- S3. Providing appropriate  $CS_i$  for each distinct real-world entity represented within an available SWS ontology  $O$ .
  - S3.1. Representing concept properties  $pc_{ij}$  of  $C_i$  as dimensions  $d_{ij}$  of  $CS_i$  (C2).
  - S3.2. Assignment of metrics to each quality dimension  $d_{ij}$  (C2).
  - S3.3. Assignment of prominence values  $p_{ij}$  to each quality dimension  $d_{ij}$  (C2).
  - S3.4. Representing all instances  $I_{ik}$  of  $C_i$  as members in  $CS_i$  (C1).

Whereas  $S1$  and  $S2$  are foreseen within the SWS vision in general,  $S3$  represents an additional activity aiming at providing the representational facilities required to realise our mediation approach. Referring to our formalisations of  $O$  and  $CS$  (Sections 2 and 3), we are able to simply instantiate a specific  $CS_i$  by applying a transformation function

$$trans : C_i \Rightarrow CS_i$$

The function is aimed at instantiating all elements of a CS, such as dimensions and prominence values (*S3.1 – S3.3*). In particular, *S3.1* aims at representing each concept property  $pc_{ij}$  of  $C_i$  as a particular dimension instance  $d_{ij}$  together with a corresponding prominence  $p_{ij}$  of a resulting space  $CS_i$ :

$$trans : \{(pc_{i1}, pc_{i2}, \dots, pc_{in}) | pc_{ij} \in PC_i\} \Rightarrow \{(p_{i1}d_{i1}, p_{i2}d_{i2}, \dots, p_{in}d_{in}) | d_{ij} \in CS_i, p_{ij} \in P\}$$

We particularly distinguish between data type properties and relations. The latter represent relations between concepts, but these are not represented as dimensions since such dimensions would refer to a range of concepts (instances) instead of quantified metrics, as required by *S3.2*. Hence, we propose to maintain the relationships represented within the original SWS ontology  $O$  without representing these within the resulting  $CS_i$ . In that, the complexity of  $CS_i$  is reduced to enable the maintainability of the spatial distance as appropriate similarity measure.

The assignment of metric scales to dimensions (*S3.2*) which naturally are described using quantitative measurements, such as size or weight, is rather straightforward. In such cases, interval scale or ratio scale are used. Otherwise, the respective dimensions need to be refined by means of subspaces (Section 3.1) until appropriate metric scales can be assigned. Since different dimensions might have distinct impact on the entire space  $CS_i$ , *S3.3* is aimed at assigning a prominence value  $p_{ij}$  to each dimension  $d_{ij}$ . Prominence values should be chosen from a predefined value range, such as 0..1. Since the assignment of prominences to quality dimensions is of major importance for the expressiveness of the similarity measure within a space, most probably this step requires incremental ex-post re-adjustments until a sufficient definition of a CS is achieved.

With respect to *S3.4*, each SWS provider (C1) has to represent all instances  $I_{ik}$  of a concept  $C_i$  as member instances in the created space  $CS_i$ :

$$trans : I_{ik} \Rightarrow M_{ik}$$

This is achieved by transforming all instantiated properties  $pi_{ikl}$  of  $I_{ik}$  as valued vectors in  $CS_i$ .

$$trans : \{(pi_{ik1}, pc_{ik2}, \dots, pc_{ikn}) | pi_{ikl} \in PI_{ik}\} \Rightarrow \{(v_{ik1}, v_{ik2}, \dots, v_{ikn}) | v_{ikk} \in M_{ik}\}$$

Hence, given a particular CS, representing instances as members becomes just a matter of assigning specific measurements to the dimensions of the CS. The accomplishment of the proposed procedure, particularly *S3*, results in a set of CS (member) instances where each CS (member) instance refines a particular concept (instance) of the SWS ontology. Please note that applying the procedure proposed here requires an additional effort.

#### 4 Similarity-based SWS Discovery for WSMO and IRS-III

The representational model described above had been implemented by and aligned to established SWS technologies based on WSMO [26] and the Internet Reasoning Service IRS-III [2]. However, please note that in principle the representational approach described above could be applied to any SWS reference model and is

particularly well-suited to support rather light-weight approaches such as SAWSDL or WSMO Lite [24].

#### 4.1 The IRS-III Service Ontology

The IRS-III Service Ontology – represented through OCML [13] – provides semantic links between the knowledge level components describing SWS and the conditions related to their use. It is based on WSMO [26][8] and contains the following main items:

- *Goal-related information.* A goal represents the user perspective of the required functional capabilities and includes a description of the requested Web service capability.
- *Web service functional capabilities.* They represent the provider perspective of what the service does in terms of inputs, output, pre-conditions and post-conditions, assumptions and effects. Pre-/postconditions and assumptions/effects are expressed by logical expressions that constrain the state or the type of inputs and outputs.
- *Web service interface.* The interface is defined by choreography and orchestration. The choreography specifies how to communicate with a Web service. A grounding describes how the semantic declarations are associated with a syntactic specification, such as WSDL. The orchestration of a Web service specifies the decomposition of its capability in terms of the functionality of other Web services.
- *Mediators.* A mediator specifies which top elements are connected and which type of mismatches can be resolved between them.

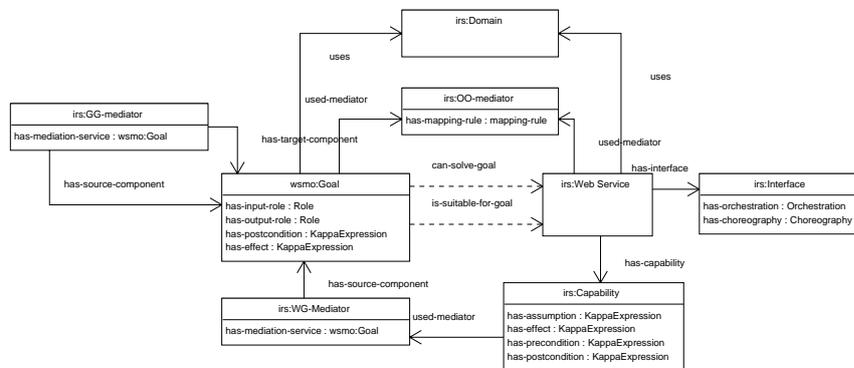


Fig. 2. IRS-III Service Ontology – core concepts and relations.

While the IRS-III Service Ontology considers Meta-classes for the top-level SWS concepts individual SWS definitions (goals, mediators, Web services) are defined as subclasses rather than instances. A class better captures, indeed, the concept of a reusable service description and taxonomic structures can be used to capture the constitution of a particular domain. At invocation time, particular instances of the respective goal, mediator and Web services automatically generated.

### 4.2 Introducing Similarity-based SWS Selection based on Conceptual Spaces

In order to facilitate the representational approach described in Section 3, we aligned the CSO (Section 3) with the IRS-III Service Ontology to allow for the refinement of individual concepts – being used as part of formal SWS descriptions – as formally expressed CS. In that, instances being used to represent SWS characteristics such as interfaces or capabilities can be refined as vectors to enable similarity computation between individual SWS and SWS requests.

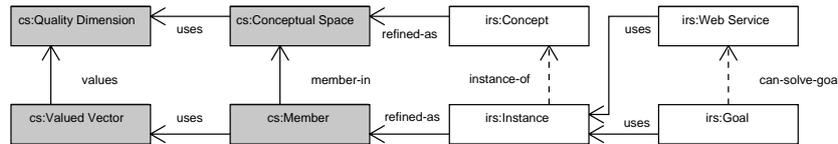


Fig. 3. Core concepts of the CS Ontology aligned to the IRS-III Service Ontology.

Figure 3 depicts the core concepts of CSO and their alignment with the IRS-III Service Ontology. Concepts (instances) as being used by IRS service or goal descriptions are refined as CS (members) within the CSO. In that, following the procedure proposed in Section 3.2, service capabilities are refined in multiple CS.

In order to facilitate automated similarity computation between SWS and SWS requests, we extended the matchmaking capabilities of IRS-III through a set of additional functions which introduce similarity computation as part of the SWS selection and matchmaking algorithm. Given the ontological refinement of SWS descriptions into CS as introduced in Section 3 this new functionality enables to automatically achieve IRS-III goals without being restricted to complete matches between a particular goal achievement request and the available SWS. When attempting to achieve a goal, our new function is provided with the actual SWS request  $SWS_i$ , named *base*, and the SWS descriptions of all  $x$  available services that are potentially relevant for the base – i.e. linked through a dedicated mediator:

$$SWS_i \cup \{SWS_1, SWS_2, \dots, SWS_x\}$$

Each SWS contains a set of concepts  $C=\{c_1..c_m\}$  and instances  $I=\{i_1..i_n\}$ . We first identify all members  $M(SWS_i)$  – in the form of valued vectors  $\{v_1..v_n\}$  refining the instance  $i_l$  of the base as proposed in Section 3. In addition, for each concept  $c$  within the base the corresponding conceptual space representations  $MS=\{MS_1..MS_m\}$  are retrieved. Similarly, for each  $SWS_j$  related to the base, members  $M(SWS_j)$  – which refine capabilities of  $SWS_j$  and are represented in one of the CS  $CS_1..CS_m$  – are retrieved:

$$CS \cup M(SWS_i) \cup \{M(SWS_1), M(SWS_2), \dots, M(SWS_x)\}$$

Based on the above ontological descriptions, for each member  $v_l$  within  $M(SWS_i)$ , the Euclidean distances to any member of all  $M(SWS_j)$  which is represented in the same space  $MS_j$  as  $v_l$  are computed. In case one set of members  $M(SWS_j)$  contains several members in the same MS – e.g.  $SWS_j$  targets several instances of the same kind – the algorithm just considers the closest distance since the closest match determines the appropriateness for a given goal. For example, if one SWS supports several different locations, just the one which is closest to the one required by  $SWS_i$  determines the appropriateness.

Consequently, a set of  $x$  sets of distances is computed as follows  $Dist(SWS_i)=\{Dist(SWS_i, SWS_1), Dist(SWS_i, SWS_2) \dots Dist(SWS_i, SWS_x)\}$  where each

$Dist(SWS_i, SWS_j)$  contains a set of distances  $\{dist_1..dist_n\}$  and any  $dist_i$  represents the distance between one particular member  $v_i$  of  $SWS_i$  and one member refining one instance of the capabilities of  $SWS_j$ . Hence, the overall similarity between the base  $SWS_i$  and any  $SWS_j$  could be defined as being reciprocal to the mean value of the individual distances between all instances of their respective capability descriptions and hence, is calculated as follows:

$$Sim(SWS_i, SWS_j) = \left( \overline{Dist(SWS_i, SWS_j)} \right)^{-1} = \left( \frac{\sum_{k=1}^n (dist_k)}{n} \right)^{-1}$$

Finally, a set of  $x$  similarity values – computed as described above – which each indicates the similarity between the base  $SWS_i$  and one of the  $x$  target SWS is computed:

$$\{Sim(SWS_i, SWS_1), Sim(SWS_i, SWS_2), \dots, Sim(SWS_i, SWS_x)\}$$

As a result, the most similar  $SWS_j$ , i.e. the closest associated SWS, can be selected and invoked. In order to ensure a certain degree of overlap between the actual request and the invoked functionality, we also defined a *threshold similarity value*  $T$  which determines the similarity threshold for any potential invocation.

A first prototypical application – accessible through an AJAX-based interface<sup>4</sup> – deploying the above functionality has been developed. The application supports similarity-based selection between a number of Web services which deliver video material and make use of the youtube-API<sup>5</sup> as well as the data feeds provided by BBC- Backstage<sup>6</sup>.

## 5 Discussion and Conclusions

We proposed a representational model which enriches the expressiveness of SWS technologies with metric-based representation in CS. As a result, the semantic meaningfulness of SWS representations is increased allowing to automatically infer about similarity-relationships between instances as used by heterogeneous SWS. We introduced a formal ontology which is aligned to the IRS-III Service Ontology and could potentially be utilised in the context of other established SWS reference models such as SAWSDL or OWL-S. In that, our two-fold representational approach provides a means to facilitate SWS interoperability. In addition, we extended the matchmaking algorithm of an existing SWS Broker, IRS-III, with new capabilities allowing for rather similarity-based matchmaking – based on our two-fold representational model – to overcome the need for strict complete SWS matchmaking. Furthermore, our approach is supported by a formal method on how to derive CS representations for individual concepts of any arbitrary SWS representations.

The proposed approach has the potential to significantly reduce the effort required to support interoperability between distinct heterogeneous SWS ontologies by overcoming the need to either subscribe to a common vocabulary or to align distinct

<sup>4</sup> <http://kmi-lisp05.open.ac.uk/demo>

<sup>5</sup> <http://code.google.com/intl/en/apis/youtube/>

<sup>6</sup> <http://backstage.bbc.co.uk/>

SWS ontologies. While our approach supports automatic similarity-computation between SWS ontology instances it requires a common agreement on shared CS. However, incomplete similarities are computable between partially overlapping CS. Given the nature of our approach – aiming at mediating between sets of concepts/instances which are used to annotate particular SWS – we argue that our solution is particularly applicable to SWS frameworks which are based on rather light-weight service semantics such as WSMO-Lite [24] or OWL-S [16]. Moreover, by representing SWS through vectors which are independent from the underlying representation language, we claim that our approach also has the potential to bridge between SWS across concurrent SWS reference models and modeling languages.

However, the authors are aware that the proposed approach requires considerable effort to establish CS-based representations. Future work has to investigate on this effort in order to further evaluate the potential contribution of the proposed approach. Moreover, while overcoming issues introduced in Section 2, further issues remain. For example, whereas defining instances, i.e. vectors, within a given CS appears to be a straightforward process of assigning specific quantitative values to quality dimensions, the definition of the CS itself is not trivial and dependent on individual perspectives. Moreover, whereas semantics of instances are grounded to metrics within a CS, the quality dimensions themselves are subject to ones interpretation what might lead to ambiguity issues. Nevertheless, distance calculation relies on the fact that resources are described in equivalent geometrical spaces. However, particularly with respect to the latter, traditional ontology and schema matching methods could be applied to align heterogeneous spaces. In addition, we would like to point out that the increasing usage of upper level ontologies and the progressive reuse of ontologies, particularly in loosely coupled organisational environments, leads to an increased sharing of (SWS) ontologies at the concept level. As a result, our proposed hybrid representational model becomes increasingly applicable by further enabling similarity-computation at the instance-level towards the vision of interoperable SWS.

## 6 References

1. Bicer, V., Kilic, O., Dogac, A., Laleci, G.B. Archetype-based semantic interoperability of web service messages in the health care domain. *International Journal of Semantic Web and Information Systems (IJSWIS)*, 1(4):1–23, October 2005.
2. Cabral, L., Domingue, J., Galizia, S., Gugliotta, A., Norton, B., Tanasescu, V., Pedrinaci, C.: IRS-III: A Broker for Semantic Web Services based Applications. In *proceedings of the 5<sup>th</sup> International Semantic Web Conference (ISWC 2006)*, Athens, USA (2006).
3. Choi, N., Song, I., and Han, H. (2006), A survey on ontology mapping, *SIGMOD Rec.*, Vol. 35, No. 3. (September 2006), pp. 34-41.
4. Cimpian, E., Mocan, A., Stollberg, M. (2006) Mediation Enabled Semantic Web Services Usage. *Proceedings of the 1st Asian Semantic Web Conference (ASWC 2006)*, September 6, 2006, Beijing, China, 2006 - Springer
5. Cregan, A. (2007), *Symbol Grounding for the Semantic Web*. 4th European Semantic Web Conference 2007, Innsbruck, Austria.
6. Ehrig, M., and Staab, S. QOM - Quick Ontology Mapping. *International Semantic Web Conference 2004*, 683-697

7. Euzenat, J., Guegan, P., and Valtchev, P. OLA in the OAEI 2005 Alignment Contest. K-Cap 2005 Workshop on Integrating Ontologies 2005, 97-102.
8. 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.
9. Fielding, R.T.: Architectural Styles and the Design of Network-based Software Architectures. PhD thesis, University of California, Irvine (2000)
10. Gärdenfors, P. (2000), Conceptual Spaces - The Geometry of Thought. MIT Press, 2000.
11. 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.
12. Krause, E. F. (1987). Taxicab Geometry. Dover.
13. Motta, E. (1998). An Overview of the OCML Modelling Language, the 8th Workshop on Methods and Languages, 1998.
14. Mrissa, M., Ghedira, C., Benslimane, D., Maamar, Z., Rosenberg, F., Dustdar, S. A context-based mediation approach to compose semantic web services. ACM Trans. Internet Techn., 8(1), 2007.
15. Noy, N.F., and Musen, M.A. The PROMPT Suite: Interactive Tools for Ontology Merging and Mapping. International Journal of Human-Computer Studies 59 (2003), 983-1024.
16. OWL-S 1.0 Release. <http://www.daml.org/services/owl-s/1.0/>.
17. Paolucci, M., Srinivasan, N., Sycara, K. Expressing WSMO Mediators in OWL-S, in proceedings of Semantic Web Services: in 3rd International Semantic Web Conference (ISWC 2004).
18. Pease, A., Niles, I., Li, J.(2002), The suggested upper merged ontology: A large ontology for the semanticweb and its applications. In: AAAI-2002Workshop on Ontologies and the Semantic Web. Working Notes (2002).
19. 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.
20. Schwering, A. (2005). Hybrid Model for Semantic Similarity Measurement, in R. Meersman and Z. Tari (Eds.): CoopIS/DOA/ODBASE 2005, LNCS 3761, pp. 1449 – 1465, 2005..
21. Spencer, B., Liu, S. Inferring data transformation rules to integrate semantic web services. In S. A. McIlraith, D. Plexousakis, and F. van Harmelen, editors, Int'l Semantic Web Conference, volume 3298 of Lecture Notes in Computer Science, pages 456–470. Springer, 2004.
22. Suppes, P., D. M. Krantz, et al. (1989). Foundations of Measurement - Geometrical, Threshold, and Probabilistic Representations. San Diego, California, USA, Academic Press, Inc.
23. SUMO: Suggested Upper Merged Ontology (<http://www.ontologyportal.org/>).
24. Vitvar, T., Kopecký, J., Viskova, J., Fensel, D. WSMO-Lite Annotations for Web Services. ESWC 2008: 674-689.
25. Winer, D.: XML-RPC Specification (June 1999), <http://www.xmlrpc.com/spec>.
26. WSMO Working Group, D2v1.0: Web service Modeling Ontology (WSMO). WSMO Working Draft, (2004). (<http://www.wsmo.org/2004/d2/v1.0/>).
27. Wu, Z., Ranabahu, A., Gomadam, K., Sheth, A.P., Miller, J.A., Automatic Composition of Semantic Web Services using Process Mediation, Proceedings of the 9th International Conference on Enterprise Information Systems (ICEIS'07), Funchal, Portugal Jun. 2007. pages 453-461.