Enhancing Service Discovery via Multi-Ontology Management and Annotation

Guang-hui HAN[1,b], Cheng ZENG*[2,3,a]

[1]Information Engineering Department, Wuhan Commercial Service College, Wuhan, China

[2]State Key Laboratory of Software Engineering, Wuhan University, Wuhan, China

[3]School of Mathematics & Computer Science, Hubei University, Wuhan, China

[a] zengcheng8@gmail.com, [b] hgh.mail@163.com

Abstract. Ontology, which adds semantic information to services, is an effective tool to automatize service discovery and composition. Though service discovery systems have been developed extensively, most possible scenario would still be that all the services adhere to a single ontology. However, even in a same domain, different domain experts or users can conceptualize the same real world entities in different views which will lead to multiple domain ontologies. Therefore how to manage multi-ontology and then how to use them to annotate service are the key technologies in service discovery system. In this paper based on ISO SC32 19763-3: MFI-3 (Metamodel Framework for Interoperability: Metamodel for Ontology Registration) we discuss in detail the mechanism of multi-ontology management and annotation to enhance the service discovery. Finally a prototype of Ontology Management Platform (OMP) is realized to serves annotation.

Introduction

With the deployment of more and more services on the Internet, it becomes much more difficult to find appropriate services. Service discovery systems, which have been developed extensively, play the key role in Semantic Service Registry and Repository (S2R2). However, service discovery need to know the meaning of service interface parameters such as Input, Output or Operation. WSDL [1], which is a kind of syntactic description, is not enough for this purpose. Therefore, Ontology is usually used to add semantic information to solve this problem [2]. It can help services to agree on a set of concepts and relationships allowing a shared understanding of the common domain knowledge. This allows matching service requests to service advertisements with less human intervention [5, 6]. Meanwhile, with the adoption of semantic analysis on ontology, the discovery of services is enriched to improve the responsiveness of the S2R2 and help users with candidate services that are close enough to what they would have liked to get.

In most cases, the semantic discovery systems adhere to single global ontology which is always constructed and standardized by Domain Experts [7]. However, even in the same domain, different domain experts or users can conceptualize the same real world entities in different views which will lead to multiple domain ontologies. Besides, the scope of domain ontology is usually arbitrary and cannot be formally defined. This is because multiple ontologies can model the same function differently. Another reason may be the co-existence of independent organizations.

For example, in the first 9 months of 2010, Google has acquired about 40 companies. Each company may adhere to their own ontology. Thus differences and overlaps between the models used for ontologies can exist. Moreover the agreements that are captured in relation to a domain may differ when seen from different perspectives and at different levels of details. Since the service requesters and providers operate independently, they usually choose any one of these available ontologies that best aligns to the service to annotate. Therefore, in such an environment where the service requesters and providers adopt different ontologies, a discovery system that supports services using different ontologies is extremely important.

From the description mentioned above, we can divide service discovery systems into two categories as shown in Fig.1: keyword-based discovery and ontology-based discovery. And the latter often refers to the service discovery in single-ontology environment and in multi-ontology environment. In the paper, we will focus on how to manage multi-ontology and then how to use them to annotate service in multi-ontology environment.
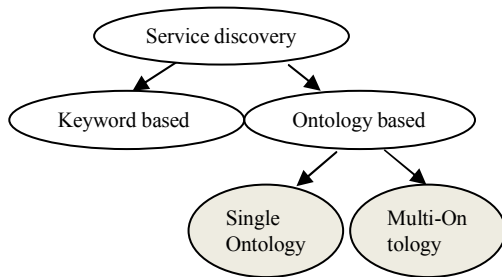

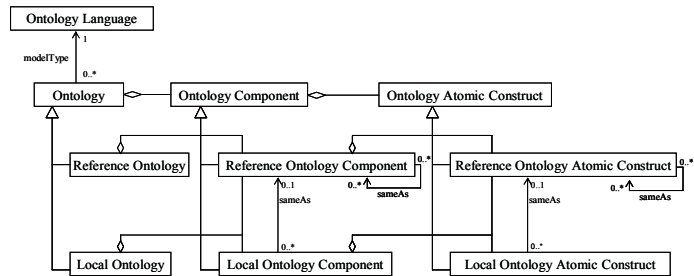Fig 1 Categories of service discovery

Fig.2. Basic Model of MFI-3

Motivated by the multi-ontology environment, a new approach of multi-ontology management and annotation in S2R2 is presented by means of ISO SC32 19763-3: MFI-3 (Metamodel Framework for Interoperability: Metamodel for Ontology Registration) [3]. Based on the approach, we construct and register the different ontologies in the same domain in order to serve annotation to web service. And more important, we specially design the Ontology Management Platform (OMP), in which different ontologies and the relationship among them can be effectively registered and managed. This platform will greatly enhance the discovery of services annotated with multi-ontology in S2R2.

The paper is organized as follows. Section II gives a short interpretation of what is MFI-3. Section III presents how to realize semantic annotation to service based on Multi-Ontologies in S2R2. In Section IV illustrate why and how to design the OMP. The related work is discussed in Section V, followed by the conclusion and future work in Section VI.

What is MFI-3

In order to effectively manage multi-ontology, we take ISO SC32 19763-3: MFI-3 as theoretical basis in this paper, which is mainly used for providing a common metamodel framework for ontology registration so that ontology definition from every metamodel could be unified [3, 4].

As shown in Fig.2, MFI-3 provides the solutions by defining "Reference Ontology" (RO), "Local Ontology" (LO). Both RO and LO are composed of the Ontology Component (OC: ROC or LOC), and Ontology Component is composed of the Ontology Atomic Construct (OAC: ROAC or LOAC). Fig. 2 also shows the ROC or ROAC defined in RO can be reused by the other ontologies, while that defined by LO can only for its own definition purpose. Generally speaking, RO is used to represent common and global ontology in that domain, which is constructed and maintained by experienced and authoritative domain experts. So RO is comparatively stable in that domain it belongs to. Different from RO, LO is often constructed and evolved from RO for particular purpose. And more important, the LOs evolved from the same RO can interoperate with each other according to the evolution rules between them, which will be discussed below. That is to say, this mechanism provides a solid foundation for semantic interoperability among multi-ontology.

Besides RO and LO, the evolution information from RO to LO is a core part in MFI-3. The ontology evolution rules can help record the detailed information when LOs evolve from RO, which ensure that the difference of ontologies will not hamper the interoperability between RO and LOs. There always exist three basic evolution rules: SameAs Rule, and Enhancement Rule and Contraction Rule. Theoretically, the three evolution rules can cover all the possible modifications on RO and LO because any complex evolution can be decomposed, viewed as a sequence consisting of these three rules.

Now the first version of MFI-3 has become the international standard since 2008 [3]. This version only includes the SameAs Rule at present (See Fig 2), and Enhancement Rule and Contraction Rule will be included in the second version of MFI-3 in the future. Therefore the evolution rules from RO

to LO described in following parts of the paper focus on the SameAs Rule. Now we give the brief definition of SameAs Rule: It can help set up the equivalent mapping between instances of Ontology, Ontology Component or Ontology Atomic Construct. If the semantic conflict occurs, we need to inform the changes to the petitioner for a better understanding of the actual needs. When SameAs Rule is adopted, there is no change in connotation of corresponding concept.

In order that the LOs evolved from the same RO can understand each other, the evolution information should be recorded when SameAs Rule is adopted. In this paper, according to MFI-3, the evolution information is kept as a form which shows that the concept XXX in LO is same as YYY in RO. That is to say, according to the SameAs Rule, we create the equivalent mapping relationship between XXX in LO and YYY in RO, which help them to understand each other.

Multi-Ontology Based Annotation in S2R2

As we all know, Domain Ontology (DO) is always constructed and standardized by domain experts who use common approved concepts and relationship among them, but sometimes even in the same domain, different domain users can conceptualize the same real world entities in different views which will lead to multiple domain ontologies. We regard the DO defined by authoritative experts as Domain Reference Ontology (DRO) and the DO defined by users as Domain Local Ontology (DLO).

Fig 3 shows DRO and DLO can both be adopted to add semantic information to Service. That is to say, we realize multi-ontology based annotation to service. This method extends the scope of semantic annotation in the domain, which will enhance the flexibility during the process of service publication and discovery.

Fig.3. Multi-ontology based annotation to service

However, DRO and DLO, which can both be annotated to Service, are not isolated but interrelated to each other. Based on MFI-3, the evolution information between DRO and DLO should be registered and recorded in S2R2. As mentioned in section II, although there exist three evolution rules between DRO and DLO, we are ready to adopt SameAs Rule because this rule is the part of the first version of MFI-3 which has already become International Standard. Next we will give an example to illustrate how DRO and DLOs interrelate to each other.

Our solution is not isolated with other classic matching algorithms, but takes them as the basis. That is to say, after applying the matching algorithms, we make a further step to calculate SameAs relationship between DRO and DLO. Therefore, it is necessary to introduce the classic matching algorithms first.

General speaking, matching algorithm is divided into two stages namely **Syntactic Matching** and **Functional Matching**.

$$MD(S_A, S_R) = \frac{w1 * SyntacticS\ im(S_A, S_R) + w2 * Functional\ Sim(S_A, S_R)}{w1 + w2} \in [0..1]$$

Where w1 and w2 are defined by users. S_A and S_R represent advertised Service and requested Service on, respectively.

$$SyntacticSim(S_A, S_R)$$

$$= \begin{cases} \dfrac{w3 * NameMatch(S_A, S_R) + w4 * DescrMatch(S_A, S_R)}{w3 + w4} \in [0..1] & \begin{aligned} &Descr[S_A] \neq \phi \ or \\ &Descr[S_R] \neq \phi \end{aligned} \\ \\ NameMatch(S_A, S_R) & \begin{aligned} &Descr[S_A] = \phi \ and \\ &Descr[S_R] = \phi \end{aligned} \end{cases}$$

Syntactic Matching. relies on name and description of S_A and S_R, which is calculated as the weighted average of the Name similarity (NameMatch (S_A, S_R)) and the Description Similarity (DescrMatch(S_A, S_R)). **Functional Matching** calculated as the average Match Score of the operations of S_A and S_R, including operation similarity, IO similarity, property similarity and etc. In this process, ontology concept matching is involved, which tries to find the requested service that is the most similar (the highest degree match).

In the paper, we are not prepared to show the complete algorithms in detail, because they are comparatively popular and widely used. So we focus on how to use MFI-3 to manage multi-ontology effectively. Next section will discuss the design process of ontology management platform.

Ontology Management in S2R2

Design of OMP. As mentioned above, DRO, DLO and the evolution information among them play the key role in S2R2 which will assure the interoperability among the services. In order to effectively manage these ontologies and evolution information, the Ontology Management Platform (OMP) based on MFI-3 is designed. First, we register DRO, including *ontology name*, *ontology document (*.owl file)*, *ontology descriptions*, *domain which the ontology belongs to* and *ontology type* (DRO or DLO). Whatever DRO or DLO, they are both registered by the pattern of "Ontology–Ontology Component–Ontology Atomic Construct" defined in MFI-3 shown in Fig 4.
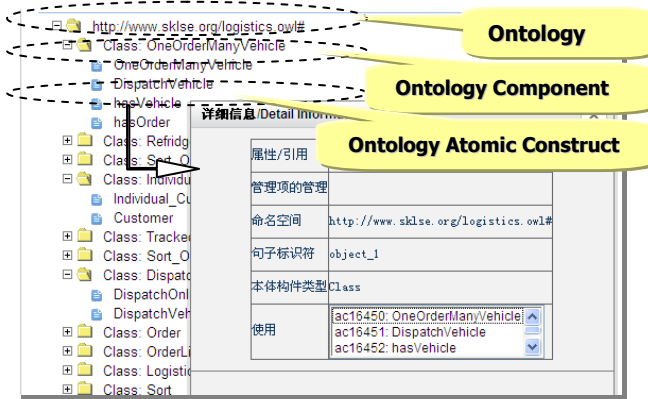

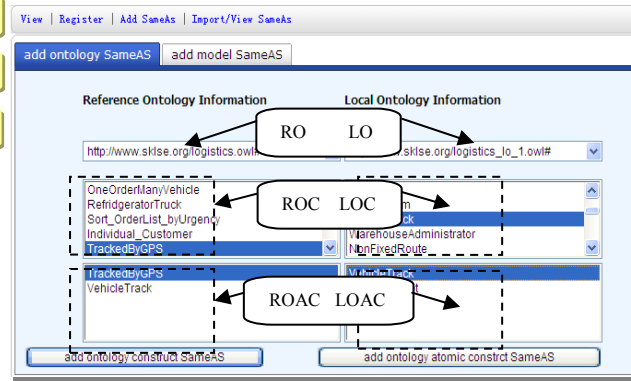Fig.4. Overview of ontology structure based on MFI-3

Fig.5. *SameAs* relationship definition between DRO/DLO

Once the DRO is successfully registered in OMP, we can generate DLO manually according to DRO and then register it in OMP. The concepts in DRO can be partly **reused** by DLO, and more important some new concepts which do not exist in DRO can also be **added** to DLO. At the same time, the *SameAs* relationship between the DRO and DLO (especially for new concepts in DLO) is established according to MFI-3.

Fig 5 illustrates how to define the *SameAs* relationship between DRO and DLO. In the left pane of Reference Ontology Information, when you choose the ontology in the dropdown list DRO, the *Ontology Component* of DRO will be listed in the ListBox automatically. And then when an *Ontology Component* is selected, *the Ontology Atomic Construct* will also be listed. In the right pane of Local Ontology information, there exist the same operations as the left pane. After operations above, we can define the *SameAs* relationship between the *Ontology Component* or *Ontology Atomic Construct* in DRO and that in DLO.

In this section, the Ontology Management Platform (OMP) based on MFI-3 is designed in which we have discussed how to register the DRO/DLO and define *SameAs* relationship between the DRO and DLO in detail. According to DRO, DLO and *SameAs* relationship between them, the multi-ontology based annotation in S2R2 can be achieved, which can help improve the semantic interoperability of services based on these ontologies.

As a whole, Fig 6 illustrates the framework of S2R2, including three components at present: **Ontology Management Platform (OMP)**, **Service Registration System** and **Semantic Query System**. From the framework, we can easily draw a conclusion that OMP plays a very important role in S2R2. On one hand, it provides a mechanism to semantically annotate services when they are registered in **Service Registration System**, on the other hand it supports semantic discovery of services in **Semantic Query System**.
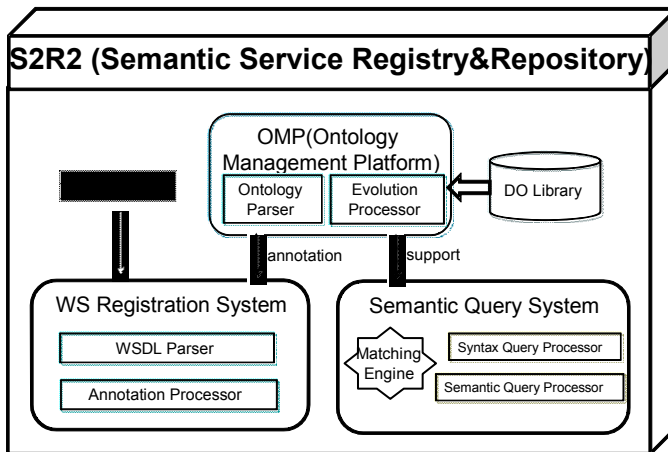
Fig.6. Framework of S2R2
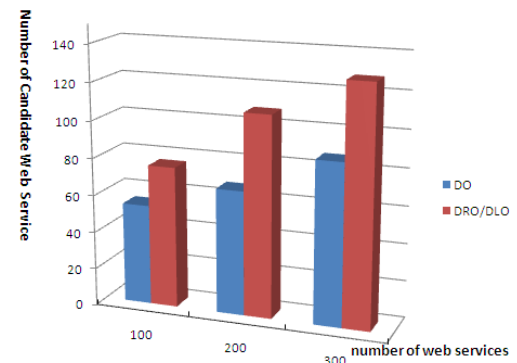
Fig.7. Relationship of the number of candidate service and the number of service in S2R2

Evaluation. To evaluate our approach of ontology management, we take recall ratio, the number of returned candidate services and the total number of services as the evaluation metrics.

Because of the absence of the relevant standard platform and test data sets, web services are randomly generated as a test case in this paper. The service itself does not perform a meaningful action, but these services have a mapping with multi-ontology. From the evaluating point of view, there is no real difference from the true Web services.

Experimental settings are:

(1) Using Protege to create domain ontologies, including class, subclass and hierarchical relationship. The number of concept of ontology is about 50~100;

(2) 100~300 web services are registered in each S2R2, and the functional description of the services such as input and output as well as the atomic service parameters are all mapped to the concept of domain ontology;

 (3) The number of input and output parameters of services is about 1~4, and each service is provided by a composition of atomic services;

(4) The performance is measured on a workstation of 2GB RAM, 2.6 GHz CPU with Microsoft Windows 2003

Fig.7 illustrates that the number of the returned candidate services annotated with DRO/DLO is increased by 40% compared to the number of the returned services annotated with single DO, which means the ratio of recall has been increased greatly, because our proposal not only makes full use of the logic reasoning algorithm in single ontology just as the traditional methods has done for many years, but also focus on the *SameAs* relationship among multi-ontology based on MFI-3 to achieve the high performance.

S2R2 Online. A test version of multi-ontology management and annotation in S2R2 had been released on website http://www.s2r2.org. Users can publish and discovery services through multi-ontology method on this platform.

Related work

During the recent years, many efforts have been done to ease the Web Services annotation and discovery process. M. Klusch etc. [14-16] design the constraints as a plug-in into a simple matchmaker version for the desired service. Other work on services discovery includes LARKS (Language for Advertisement and Request for Knowledge Sharing) [8, 9], a project based on a collaboration between Toshiba and Carnegie Mellon University [10, 11], a Matchmaker from TU-Berlin [12], and systems by Li and Horrocks [13], Paolucci [7] etc. However, these services discovery systems only support matching services using the same or single ontology which both provider and requester share. This assumption implies that if different ontologies are used, matching cannot be carried out. As mentioned, this is a major limitation. That is to say it might produce a semantic gap between providers and requesters due to the use of different ontologies to describe services and requests, respectively .

There exist some studies in multi-ontology environment. In [17], the system is extended with a method for allowing providers and requesters to use different ontologies. This method only takes into account the two provided ontologies (provider's and requester's ones). Our method enables to add evolution information to the different ontologies, which can simplify the process of service discovery. In [18], another proposal for matchmaking based on OWL-S is presented. In this system, when providers and requesters use different ontologies, the mappings have to be provided by the requester. All these approaches could get rid of their assumption of using the same ontologies for both requester and provider (or having the mappings) by using the ontologies integrated with the information of the extracted senses. Although the technologies of ontology mapping and merging might be an important step towards semantic interoperability, the building of mappings and mediators between ontologies is a costly process to be done manually and difficultly.

Our approach of multi-ontology (DRO/DLO) annotation based MFI-3 is adopted to reduce the complexity of ontology management. More important, we design the OMP to effectively manage DRO, DLO and the evolution information between them which will lead to the interoperability of services in S2R2 based on these ontologies. That is to say, our approach will extend the scope of semantic annotation in the Domain and greatly enhance the flexibility during the process of service publication and discovery.

Summary

In this paper, we make full use of MFI-3 to enhance the services discovery in multi-ontology environment. Different from the traditional Service Registry, in S2R2 providers and requesters can use different ontologies (DRO/DLO) in the same domain to annotate and discovery services. More important, in order to effectively manage the DRO, DLO and the evolution information between them, we design the Ontology Management Platform to help realize semantic annotation and discovery of services based on multi-ontology. The evaluation result illustrates that the performance of discovery is increased greatly by use of our approach.

In future, we plan to expand the Ontology Evolution Rules which make the evolution information to be recorded comprehensively and clearly, and verification criterions should be attached to each kind of rules to avoid semantic contradiction when ontology evolving.

Acknowledgment

References

[1] World Wide Web Consortium (W3C). WSDL Specification Version 1.1. [EB/OL]. http://www.w3.org/TR/2001/NOTE-wsdl-20010315

[2] K. Verma and A Sheth, "Semantically Annotating a Web Service," IEEE Internet Computing 11 (2),March–April 2007, pp. 83–85

[3] ISO/IEC 19763-3:2007. Keqing He. http://www.iso.org/iso/iso_ catalogue/catalogue_tc/ catalogue_detail.htm?csnumber=38637, 2008

[4] Chong Wang, Keqing He, Yangfan He. MFI4Onto: Towards Ontology Registration on the Semantic Web [A]. In Proceedings of 6th International Conference on Computer and Information Technology, CIT 2006, Seoul, 2006, pp.40-46

[5] D. Fensel and C. Bussler, "The Web Service Modeling Framework WSMF", Electronic Commerce: Research and Applications, 1 (2002) 113-137

[6] J. Cardoso, C. Bussler, "Semantic Web Services and Processes: Semantic Composition and Quality of Service", On the Move to Meaningful Internet Computing and Ubiquitous Computer, Irvine CA, October 2002

[7] M. Paolucci, T. Kawamura, T. Payne, and K. Sycara, "Semantic matching of Web Services capabilities", in Proc. of the 1st International Semantic Web Conference(ISWC), 2002

[8] Sycara, K.; J. Lu, M. K.; Widoff, S., Dynamic service matchmaking among agents in open information environments, ACM SIGMOD Record (Special Issue on Semantic Interoperability in Global Information Systems),, (1999) 28, No.1, 47

[9] Sycara, K.; Widoff, S.; M. Klusch, J. L., (2002), LARKS: Dynamic Matchmaking Among Heterogeneous Software Agents in Cyberspace, Autonomous Agents and Multi- Agent Systems, Vol.5, 173

[10] Kawamura, T.; Blasio, J. D.; Hasegawa, T., et al., (2003), Preliminary Report of Public Experiment of Semantic Service Matchmaker with UDDI Business Register, 1st International Conference on Service Oriented Computing (ICSOC 2003), Trento, Italy

[11] Kawamura, T.; Blasio, J. D.; Hasegawa, T., et al., (2004), Public Deployment of Semantic Service Matchmaker with UDDI Business Registry, 3rd International Semantic Web Conference (ISWC 2004), LNCS 3298

[12] Jaeger, M. C.; Tang, S.; Liebetruth, C., The TUB OWL-S Matcher. Available at: http://ivs.tuberlin.de/Projekte/owlsmatcher/index.html

[13] Li, L.; Horrocks, I., (2003), A software framework for matchmaking based on semantic web technology, 12th International World Wide Web Conference, Budapest, Hungary, ACM Press

[14] M. Klusch, B. Fries, and K. Sycara, "Automated Semantic Web Service discovery with OWLS-MX," in Proc. of the 5th Intl. Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2006), Hakodate, Japan. ACM, May 2006, pp. 915–922.

[15] M. Klusch and F. Kaufer, "WSMO-MX: A hybrid Semantic Web Service matchmaker," Web Intelli. and Agent Sys., vol. 7, no. 1, pp.23–42, 2009.

[16] M. Klusch, P. Kapahnke, and I. Zinnikus, "SAWSDL-MX2: A machine-learning approach for integrating Semantic Web Service matchmaking variants," in Proc. of 7th Intl. Conference On Web Services (ICWS 2009), Los Angeles, CA, USA. IEEE, July 2009, pp. 335–342.

[17] S. A. Oundhakar, K. Verma, K. Sivashanmugam, A. P. Sheth, and J. A. Miller, "Discovery of Web Services in a multi-ontology and federated registry environment," Int. J. Web Service Research., vol. 2, no. 3, pp. 1–32, 2005.

[18] J. Pathak, N. Koul, D. Caragea, and V. G. Honavar, "A framework for Semantic Web Services discovery," in Proc. of the 7th Annual ACM Intl. Workshop on Web Information and Data Management (WIDM 2005), Bremen, Germany. ACM, November 2005, pp. 45–50.