An Ontology-based Method and Tool for Cross-Domain Requirements Visualization

Nirav Ajmeri, Kumar Vidhani, Manoj Bhat, Smita Ghaisas
Tata Research Development and Design Centre
A Division of Tata Consultancy Services
54 – B, Hadapsar Industrial Estate, Pune – 411013, India
e-mail: {nirav.ajmeri, kumar.vidhani, manoj.bhat, smita.ghaisas}@tcs.com

Abstract - **The complexity associated with understanding the cross-domain scope of a requirement has always been a challenge. Requirement Analysts use their experience in determining the functional scope boundaries of requirements. However, chances of missing out key concepts in domains peripheral to the domain of interest are quite high. Ontologies are increasingly becoming the standard way of representing shared understanding of a domain and their use in understanding and visualizing the cross-domain scope of requirements can be a step towards improving the completeness of requirements. We present a method - based on ontology mapping technique, and an assisting tool that would help Requirement Analysts visualize - how requirements span across multiple domains.**

Keywords- cross domain scope of requirements; ontology for requirements; visualization ; multiple ontologies

I. INTRODUCTION

Requirement Analysts and/or Subject Matter Experts (SME) responsible for elicitation of requirements are aware of intricacies of their respective domains of expertise. But typically in any project, requirements span across multiple domains. A lack of understanding of interaction between domains can result in incomplete requirements and wrong estimation and schedule slippage thereof.

Ontologies representing domain knowledge [1, 5] can be used to manage [8, 9], improve completeness and formally represent requirements specification [10]. Different ontologies are created by ontology engineers based on individual perspectives of their respective domains. This can result in assigning different meanings to the same concepts in different domain ontologies.

The contribution of this paper is twofold. First, we present a method to identify interfaces between ontologies of different domains. Second, we propose criteria to derive conceptual model of requirements with the help of identified interfaces. The derived model visually represents multi-domain interaction in the context of a requirement and thus contributes to the understanding of a requirement by improving its completeness.

II. METHOD FOR IDENTIFYING INTERFACES

In this section we present a method for identifying interfaces between the ***central domain*** (the domain of interest) and the ***peripheral domain*** (domain peripheral to the domain of interest). Our method extends existing ontology mapping techniques [2, 3, 11-15], that focus on mapping ontologies of same domain, to identify interfaces between ontologies of different domains.

To detect ***interfaces*** or ***interface concepts***, we compute *Semantic Similarity*, measured as a weighted mean of *Syntactic similarity*, *Sense similarity* and *Context similarity* between two concepts of different ontologies. (1) *Syntactic similarity* is based on the notion that if the strings of two concepts are syntactically the same, they are most likely to convey same meaning. It is computed as the Jaro-Winkler[7] distance between two strings. (2) *Sense similarity* takes into account the fact that each word has multiple usage sense. We use wordnet synsets [4] to compute sense similarity as a ratio of *'similar usage senses'* of concepts over *'all usage senses'*. (3) The association of individual concepts with their neighborhood concepts establishes the context of a concept. Taking this into account, *Context similarity* of two concepts is computed as a mean of sense similarity of their respective parent and child concepts.

Deriving a conceptual model requires understanding of not only the interface concepts but also the concepts associated through some relation(s) with the interface concept. We refer to the associated concepts as *complementary concepts*.

III. METHOD FOR DERIVING CONCEPTUAL MODEL FROM MULTIPLE ONTOLOGIES

To derive the conceptual model of requirement from multiple ontologies we need to identify (1) if the identified interface concepts represent functionality, (2) the domain of user story. We use OpenNLP [16] for concept extraction and POS tagging. The interface concept is said to be representing functionality if it is present in the list of functional user story and is part of a verb phrase or prefixed by a verb phrase. The domain of a user story is the domain to which the maximum number of extracted concepts [16] of the user story maps.

Based on the domain of the user story, the conceptual model of a user story is defined as per the following criteria -

Criterion 1. If the user story executes in peripheral domain, there is a need to register its successful completion to central domain. Here, Requirement Analyst should decide which complementary concept from peripheral domain registers this successful completion. We define this concept as the *acknowledgement concept*. Conceptual model in this case would include interface concepts, complementary concepts from central domain ontology, acknowledgement concept from peripheral domain ontologies, and all constraints restricting these concepts.

Criterion 2. If the user story executes in central domain, conceptual model would include interface concepts, complementary concepts from central domain and peripheral domain ontologies, and all constraints restricting these concepts.

IV. Expiremental Results And Conclusion

Based on the method described in Section II and III, we developed an ontology-based tool. Ontologies for Insurance, Banking and Healthcare domain are created using semi-automated method discussed in our earlier work [6]. Table I lists some of the user stories of Insurance domain which are found to be interacting with peripheral domains (Banking and Healthcare), with corresponding interface concepts and complementary concepts.

TABLE I. Insurance domain user stories interacting with Banking and Health-care domains (For Semantic Similarity: 0.5)

User Story	Description	User Story Domain	Interface and Complementary Concepts#
US_1	As Insurance Company, we want policyholders to make payment of premium online through internet banking.	Banking	Payment($C_{interface}$), Premium, Risk, Transaction(C_{ack})
US_2	As a Claim Scrutinizer, I should be able to view insured's medical history in claims scrutiny screen.	Insurance	Insured-Patient($C_{interface}$), Policy, Personal Details, Contact Details, Medical Report, Prescription, Disease, Healthcare provider
US_3	As Claims Manager, I want system to find out symptoms which are pointers to terminal diseases in health records furnished by customer.	Insurance	Symptom($C_{interface}$), Disease, Insurance, Insurance policy, Syndrome, Organ

$C_{interface}$ – *interface concept*, C_{ack} – *acknowledgement concept*

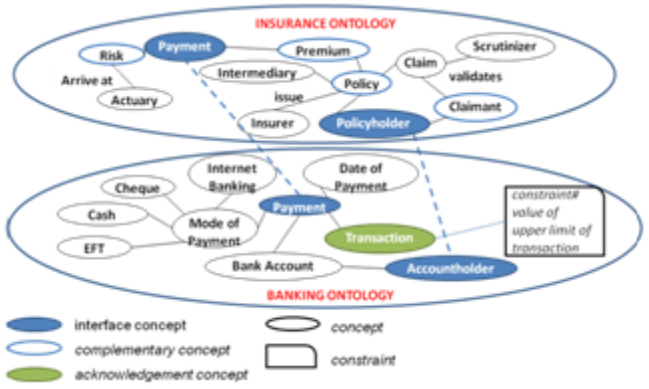

Figure 1. Conceptual model of US_1

The derived conceptual model of US_1, as shown in Figure 1, suggests that- to automate US_1, apart from concepts from Insurance domain, we also need to the consider acknowledgement concept *Transaction* (with associated constraints) from Banking domain.

Our method tested on sample user stories brings out that the approach can help in explicitly visualizing the cross-domain scope of requirements and improve their completeness at the stage of specification itself. However, we are aware that the completeness of ontologies is a precursor to this method. This is a first step towards solving a practical problem often encountered by Requirement Analysts and as such open to discussions and criticisms.

References

[1] Gruber, Thomas R., "A translation approach to portable ontology specifications". Knowledge Acquisition 5 (2), June 1993, pp. 199–220.

[2] Kalfoglou Y., Schorlemmer M. "Ontology mapping: the state of art", The Knowledge Engineering Review, Volume 18 Issue 1, January 2003.

[3] Choi N., Song Y., Han H. "A Survey on Ontology mapping", ACM SIGMOD Record Volume 35 Issue 3, September 2006.

[4] Princeton University, "About WordNet.", Princeton University, 2010.

[5] Chandrasekaran, B., Josephson, J., Benjamins, V. "What Are Ontologies, and Why Do We Need Them?", IEEE Intelligent Systems 14, 1999, pp. 20-26.

[6] N. Ajmeri, R. Sejpal, S. Ghaisas, "A Semantic and Collaborative platform for Agile Requirements Evolution", Third Workshop on Managing Requirements Knowledge, MaRK10, September 2010, pp. 32-40.

[7] Winkler, W. E. , "String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage". Section on Survey Research Methods, American Statistical Association, 1990, pp. 354-359.

[8] Mayank, V., Kositsyna, N., Austin, M., "Requirements Engineering and the Semantic Web", Part II. Representation, Management, and Validation of Requirements and System-Level Architectures. Technical Report. TR 2004-14, University of Maryland, 2004.

[9] Decker, B., Rech, J., Ras, E., Klein, B., Hoecht, C., "Selforganized Reuse of Software Engineering Knowledge supported by Semantic Wikis", Workshop on Semantic Web Enabled Software Engineering (SWESE), November 2005.

[10] Wouters, B., Deridder, D., Van Paesschen, E., "The Use of Ontologies as a Backbone for Use Case Management", European Conference on Object-Oriented Programming (ECOOP 2000), Workshop: Objects and Classifications, a natural convergence, 2000.

[11] Shaviko P., Euzenat J., "A Survey of Schema-based Matching Approaches", Journal on Data Semantics IV, 2005, pp. 146-171.

[12] Rubin D.L., Shah N.H., "Biomedical ontologies: a functional perspective", Briefings in Bioinformatics, 9(1), 2008, pp.75-90.

[13] Rodriguez M.A., Egenhofer M.J., "Determining Semantic Similarity Among Entitiy Classes from Different Ontologies", IEEE Transactions on Knowledge and Data Engineering, Vol 15, 2003, pp. 442-456.

[14] A. Maedche, S. Staab, "Semi-automatic engineering of ontologies from texts", 12th International Conference on Software Engineering and Knowledge Engineering (SEKE 2000), Chicago, IL, USA, July 2000, pp. 231-239.

[15] Madhavan, J.; Bernstein, P. A.; Rahm, E., "Generic Schema Matching with Cupid", 27th Int. Conf. on Very Large Databases (VLDB), 2001.

[16] Baldridge J, "The OpenNLP project", 2005.