AN ONTOLOGY CHANGE MANAGEMENT SYSTEM
An Experiment on a Health Care Case Study

Soumaya Slimani, Karim Baïna, Salah Baïna
ENSIAS, Mohammed V-Souissi University, Rabat, Morocco

Martin Henkel, Erik Perjons
Department of Computer and Systems Sciences, Stockholm University, Stockholm, Sweden

Keywords: Ontology Evolution, Semantic Service, Multi-agent System, Eye Specialist Ontology, Primary Health Care Ontology.

Abstract: Numerous ontologies have been developed for life science domains. These ontologies are continuously changing. Thus, it is becoming profitable to study and to manage these ontologies change in order to keep all dependent ontologies and their related mappings consistent. The aim of this paper is to propose an agent based approach enabling not only ontology and ontology mapping evolution analysis but also to manage their changes. An experiment in health care illustrates the benefits of our approach. We apply our algorithm, and implementation prototype *p²OEManager* to eye specialist ontology (ESO) and primary health care ontology (PCO), and particularly, we use our ontology agent model, and prototype to manage some significant changes in the ESO ontology.

1 INTRODUCTION

Ontologies become increasingly important in life sciences. In electronic health care, the greater problem is the heterogeneity of information systems. Semantic interoperability of these heterogeneous systems can be achieved through an agreement between the underlying ontologies (e.g. RDF, OWL, etc.). In the context of web services, several standards were developed to describe web services semantics (e.g. WSDL-S, WSMO, OWL-S, etc.). Due to the rapid development of life science research, ontologies evolve continuously, i.e., they are frequently changing to incorporate new domain knowledge into them. However, these changes, may impact the correctness of future communication using these ontologies because other services are not aware of these changes. Hence, ontology mappings (which allow services to interpret correctly (translate) exchanged data) should be corrected. Research in ontology evolution and change management deal particularly with the versioning and evolution of the same ontology, and do not process the evolution of different ontologies, interrelated by mappings and describing different services. In this paper we propose an agent-based algorithm and prototype managing the ontologies evolution life cycle, as well as the evolution of ontology-related mappings. Also, in a comprehensive evaluation, we apply our algorithm to eye specialist ontology (ESO) and primary health care ontology (PCO), and, particularly, we use our ontology agent model and p2OEManager prototype to manage some significant changes in the ESO ontology.

2 *P²OEMANAGER* DESIGN AND IMPLEMENTATION OVERVIEW

Fig. 1. shows the general architecture of *p2OEManager* (which stands for peer to peer Ontology Event Manager) upon 3 main components: Ontology Manager, Ontology Mapping Manager, and Ontology Agent Manager built on an open Service Layer. Interactions between instances *p2OEManager* peers (i.e. ontology change event

Slimani S., Baïna K., Baïna S., Henkel M. and Perjons E..
AN ONTOLOGY CHANGE MANAGEMENT SYSTEM - An Experiment on a Health Care Case Study.
DOI: 10.5220/0003105504490452
In *Proceedings of the International Conference on Knowledge Engineering and Ontology Development* (KEOD-2010), pages 449-452
ISBN: 978-989-8425-29-4

messages) are handled through an ontology change communication channel. In fact, *p2OEManager* is neutral regarding the service infrastructures. Business messages between these services are supported outside the scope of *p2OEManager* within service marshalling/unmarshalling and security, addressing, reliable messaging, routing, and transport standard protocols.

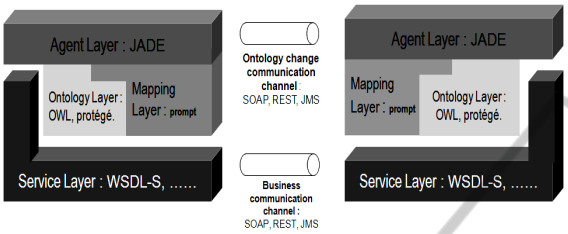

Figure 1: *p2OEManager* Architecture.

As shown in Fig.1. , Service layer, which is outside *p2OEManager*, represents services that are described by service ontologies. These service ontologies are defined by human designers using ontology editors, and these service ontologies are then monitored synchronously by *p2OEManager*. Service dependencies are represented by service ontology mappings defined by human designers too using ontology mapping generators. Services communicate and collaborate with each other within service infrastructure based on ontology mappings evolving in *p2OEManager*.

Our proposal is based on a combination of ontologies and agents. We associate an agent to each ontology. This agent is responsible of change management and propagation of these changes to other dependent ontologies. Table 1 summarizes the four cases and the agent actions for each case.

For more formalisation details of our ontology agent model, and algorithms implemented within our *p²OEManager* Architecture, please refer to our paper (Slimani and al., 2010).

Table 1: Dependent agent actions.

		Syntactic Search	
		True	False
Semantic Search	True	-Updates the related mapping	-Updates the mapping. -Annotates the corresponding object by the added object.
	False	-Reformulates the change definition. -Negotiates the change definition.	-Adds the new object to the ontology. -Updates the related mapping.

3 EVALUATION SCENARIO: THE S:TERIKS HEALTH CARE

In order to illustrate the approach presented in this paper a health-care case from the REMS project. The main objective of the REMS project was to develop a set of e-services that could be used to create, manage and transfer health care referrals between S:t Erik's eye hospital specialist clinic and primary care units. Having a set of e-services available from different health care providers would enable healthcare systems to be interconnected in order to share information. To achieve this integration, it is crucial for that the services share the same set of concepts. Fig. 2 shows the value model (Henkel and al., 2007) which depicts the main flow of resources between the patient, the primary health care and the eye specialist clinic. There are important aspects that need to be considered when designing e-services for the information exchange. In the case of Swedish health services these should follow standards on the international and national levels. However, even given these standards it is still plenty of room for interpretation of the concepts. Furthermore, there is also a need to specialize the concepts/models in order to cover specific details of the Eye health care. Thus it is likely that two organizations that follow the standardized concepts/models will end up with two different models that need to be kept synchronized if they are to be able to exchange information.

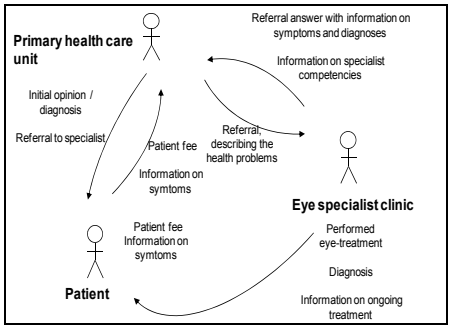

Figure 2: Actors and resource exchanges in the S:tEriks health care case.

For this example, we focus on the interconnection of the systems at the primary care units (PCU) and the systems at the eye specialist clinics (ESC). Fig. 3 shows the basic ontology that is used on the PCU systems. Fig. 4 shows the ontology that is used at the ESC. These ontologies will be the basis for the services that need to interconnect when exchanging referral information. We have developed the

Primary health care ontology (PCO) and the Eye specialist ontology (ESO), using Protégé 3.4 (http://protege.stanford.edu).

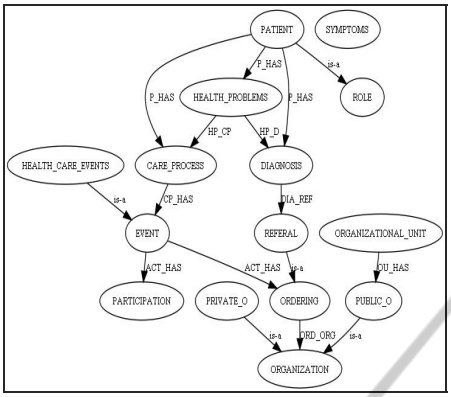

Figure 3: Primary health care ontology (PCO).

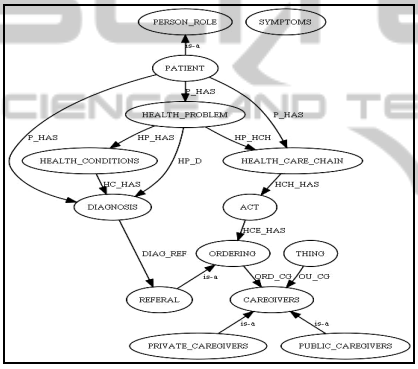

Figure 4: Eye specialist ontology (ESO).

Common to both ontologies are the concepts of PATIENT, ROLE, HEALTH PROBLEMS, SYMPTOMS and REFERAL. We have generated a mapping using Prompt which generate an OWL file for the mapping[1] (http://protege.stanford.edu/plugins/prompt/prompt.html). For the sake of describing our approach we provide in the next section four examples of changes

4 ALGORITHM APPLICATION: PROCESS OF CHANGES

As input of the process of using the Ontology Agent Model, we have the Primary health care ontology (PCO), the Eye specialist ontology (ESO), and the mapping between PCO and ESO (MEP). The process of using the Ontology Agent Model can be presented in 3 steps: (1) We have paramerezed the Ontology Agent Model by integreting PCO, ESO, and MEP URIs. We have obtained two Ontology Agent: one for the Primary health care ontology (PCO) and another for the Eye specialist ontology (ESO), (2) run the PCO agent and the ESO agent, (3) Process changes.

The application of the algorithm in this scenario begins on the Initiator side (ESO agent). First, changes are listed as follow:

```
  changeSet = <
  C1=<ADD :Class :EYE REFERAL:
subclassOf REFERAL>,
  C2=<ADD:Class:HEALTH CARE ACTIVITY:sub
classOf ACT>
  C3=<ADD:Class:PARTICIPATION:subclassOf
Thing>
  C4=<ADD:Class:ORGANIZATIONAL UNIT
:subclassOf Thing>
  C5=<ADD:ObjectProperty:ORG_CG:<Domain:
ORANIZATIONAL_UNIT, Rang:CARE_GIVERS>>
  >
```

Then changes are classified according to their relationship with the mapping. All changes are sent to PCO agent which process for each change the correspondents actions as follow:

For C1. The syntactic and semantic search return false. So, PCO update the mapping by adding the following correspondences: *mapped* (ESO. C_1, PCO.C_1)

For C2. The syntactic search return false, but semantic search will return true, because HEALTH_CARE_ACTIVITY can be mapped to the existing concept of HEALTH_CARE_EVENTS in the PCO. Thus the PCO agent (1) Annotate the concept HEALTH_CARE_EVENTS concept by adding Rdfs:seeAlso HEALTH_CARE_ACTIVITY annotation, (2) Update mapping by adding the following correspondence :
mapped(HEALTH_CARE_EVENTS,HEALTH_CARE_ACTIVITY)

For C3. The syntactic and the semantic search return true. So, the PCO agent updates the mapping by adding the following correspondence:
mapped(ESO.PARTICIPATION, PCO.PARTICIPATION)

For C4 and C5. The concept ORGANIZATIONAL_UNIT is added with an association to CARE_GIVERS in the ESO. ORGANIZATIONAL_UNIT matches syntactically to ORGANIZATIONAL_UNIT in the PCO, but this match is semantically incorrect since only PUBLIC_O can have ORGANIZATIONAL_UNIT in the PCO. So, the PCO agent calculates adefinition for ORGANIZATIONAL_UNIT as follow:

```
<<Class:PCO. ORGANIZATIONAL UNIT:
   subclass of Thing>
<ObjectProperty: PCO.OU HAS :
   Domain: PCO.RGANIZATIONAL UNIT,
   Rang :PCO.PUBLIC_O >>
```

PCO Agent sends this definition to ESO agent, which calculates a new definition:

```
<<Class:PCO.ORGANIZATIONAL UNIT:
   subclass of Thing>
<ObjectProperty:PCO.OU HAS:
   Domain: PCO.RGANIZATIONAL_UNIT,
   Rang:PCO.PUBLIC_O>
<Class:ESO.ORGANIZATIONAL_UNIT:
   subclass of Thing>
<ObjectProperty: ESO.OU HAS :
   Domain:ESO.ORGANIZATIONAL_UNIT,
   Rang:ESO.CARE_GIVERS>>
```

The following correspondence introduces a conflict in the definition of CARE_GIVERS:
mapped (ESO.CARE_GIVERS, PCO.PUBLIC_O)
Indeed, as we have already in the mapping that ESO.PUBLIC_O corresponds to PCO.PUBLIC_O and ESO.PUBLIC_O is a subclass of CARE_GIVERS, this can be a source of errors. So, the agent sends an alert message to the user. User can modify the mapping and the ontology or validate any of the definitions contained in the negotiation exchange. In this example, our algorithm allows services in the Eye Specialist Clinic and in the Primary Care Provider to be aware of evolution in other services. Agents take the necessary decisions to maintain a reliable exchange of data between these entities.

5 RELATED WORKS

The ontology evolution and change management has been addressed by many researches. (Klein and al., 2001). (Klein, 2004), investigated the versioning of ontologies, (Plessers and al., 2007) define Change Definition Language (CDL), (Djedidi and al., 2009) a patterns-based ontology evolution approach , (Zablith, 2009) a Framework for ontology evolution and (Hartung and al., 2008). But we propose an evolution model for multi-ontology system when ontologies are different and not only for instance. When ontologies are considered as an ontology instances, the source ontology has the semantics of the dependent ontologies (because they are instances of the source ontology). So, ontology evolution will require applying the same changes on the dependent ontologies. Note also that in the case of different ontologies, mapping between ontologies must be managed in parallel.

6 CONCLUSIONS AND FUTURE WORK

Interconnecting services is a complex task. A part of the complexity comes from the need to have a common, shared view of the information. The approach proposed in this paper is suitable for ontology evolution management in distributed and heterogeneous environments. The aim is to provide a flexible way to partially automate the process of ontology evolution management. The approach consists of software agents that represent each of the services involved, and their respective ontologies and related mappings. To illustrate our approach we applied it to a health care case. The case study highlighted some of the main benefits of the approach. The approach could be extended and improved in several ways. First, there is a need to further analyze the implications that changes have to the logic of the running software services. Furthermore, there is a need to extend the ontology agent model to include different types of changes in our algorithms (removal, etc...).

REFERENCES

Slimani,S.,Baïna,S.,Baïna,K., "Agent-based Architecture for Service Ontology evolution Management", SEKE'10, San Francisco Bay, USA 2010.

Henkel, M., Johannesson, P., Perjons, E., Zdravkovi, J., "Value and Goal Driven Design of E-Services", The IEEE International Conference on e-Business Engineering (ICEBE'07), Hong Kong, China, October 24-26.

Klein, M., Fensel, 2001. D. Ontology versioning on the Semantic Web. In: Proc. Int. Semantic Web Working Symposium (SWWS).

Klein, M., 2004. Change Management for Distributed Ontologies. PhD thesis, Vrije Universiteit Amsterdam.

Plessers, P., Troyer, O., and Casteleyn, S., 2007. Understanding ontology evolution : A change detection approach, Web Semantics : Science, Services and Agents, on the World Wide Web 5 39-49.

Djedidi, R., Aufaure, MA., 2010. ONTO-EVOAL an Ontology Evolution Approach Guided by Pattern Modeling and Quality Evaluation. FoIKS.

Zablith, F. 2009. Evolva : A Comprehensive Approach to Ontology Evolution". In Proceedings of the 6th European Semantic Web Conference PhD Symposium LNCS 5554. eds. L. Aroyo et al., Springer-Verlag, Berlin, Heidelberg, pages 944-948, ESWC.

Hartung, M., Kirsten, T., Rahm, E., 2008. Analyzing the evolution of life science ontologies and mappings. In: Proc. of DILS.