Ontologies Change and Queries Break: Towards a Solution

Yaozhong Liang
Intelligence, Agents and Multimedia Group
School of Electronics and Computer Science
University of Southampton
Highfield, Southampton
England, United Kingdom
yl504r@ecs.soton.ac.uk

Harith Alani
Intelligence, Agents and Multimedia Group
School of Electronics and Computer Science
University of Southampton
Highfield, Southampton
England, United Kingdom
ha@ecs.soton.ac.uk

Nigel Shadbolt
Intelligence, Agents and Multimedia Group
School of Electronics and Computer Science
University of Southampton
Highfield, Southampton
England, United Kingdom
nrs@ecs.soton.ac.uk

ABSTRACT

Keeping track of ontology changes is becoming a critical issue for ontology-based applications. Updating an ontology that is in use may result in inconsistencies between the ontology and the knowledge base, dependent ontologies and applications/services. Current research concentrates on the creation of ontologies and how to manage ontology changes in terms of mapping ontology versions and keeping consistent with the instances. Very little work investigated controlling the impact on dependent applications/services; which is the aim of the system presented in this paper. The approach we propose is to make use of ontology change logs to analyse incoming RDQL queries and amend them as necessary. Revised queries can then be used to query the ontology and knowledge base as requested by the applications and services. We describe the design of our prototype system, and discuss related problems and future directions.

Categories and Subject Descriptors

H.3.1 [**Content Analysis and Indexing**]: Abstracting methods; H.3.3 [**Information Search and Retrieval**]: Query formulation; I.2.4 [**Knowledge Representation Formalisms and Methods**]: Representation languages

General Terms

Ontology Management

Keywords

Ontology Change Management, Ontology Versioning, Knowledge Management, Semantic Web

1. INTRODUCTION AND RELATED WORK

Ontologies are quickly becoming indispensable parts of the Semantic Web. The number of ontologies that are being developed and used by various applications is continuously increasing. One of the major problems with ontologies is change. Ontology changes may cause serious problems to its data instantiations (the knowledge base), the applications and services that might be dependent on the ontology, as well as any ontologies that import that changed ontology [3].

Most work so far has focused on ways to handle ontology change, such as change characterisation [3], ontology evolution [4], ontology versioning [2], and consistency maintenance [5, 6, 7]. However, not much has been done with respect to using change-tracks to eliminate or reduce any impact that ontology change can have on any dependent applications and services. It would be very costly and perhaps even unrealistic to expect all parties that could be affected by a change to coordinate any such changes [1]. Therefore, we believe that it would be very beneficial to have a system that could track such changes, relate changes to incoming queries, amend such queries accordingly, and inform the query source of those changes and actions taken.

In this paper we describe a prototype system that targets these problems. The system uses a semantic log of ontology change to amend RDQL queries sent to the ontology as necessary. Such a system could save many hours of application re-development by not only updating queries automatically and maintaining the flow of knowledge to the applications as much as possible, but also to inform the developers of such changes in the ontology that relates to their queries.

2. SYSTEM DESCRIPTION

The solution shown in Figure 1 to tackle the identified problems is described as a series of steps as follows:

1. **Capture**: The changes made between two versions of the same ontology is captured at this stage. Currently, we identify changes by comparing two versions using PromptDiff in Protégé [4].
2. **Instantiate**: The *Log Ontology* is populated with change information identified in step 1.
3. **Analyse**: Queries submitted by the applications are analysed to find out whether any of the entities within the queries could be affected by the changes stored in the Log Ontology.

Posters and Demos of the 15th International Conference on Knowledge Engineering and Knowledge Management, EKAW 2006 Podebrady, Czech Republic, 2nd-6th October, 2006

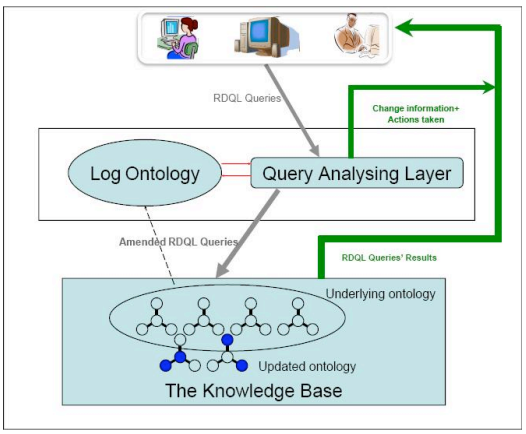

Figure 1: An overview of the Approach

4. **Update**: If entities within the queries are found to have been changed, they are replaced with their changes to form the new queries with updated entities, and then resubmitted to the queried ontology.

5. **Respond**: After the new-formed queries are submitted to the ontology for processing, the results are returned back to the application. At the same time, a summary of change/update information will also be returned back to the end-users with the query results so as to inform users of the updates.

Analyse, Update and Respond are implemented in the Middle Layer System in Figure 1. Its working process is presented in Figure 2.

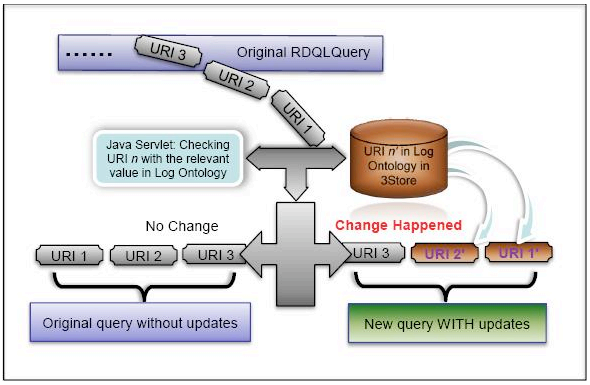

Figure 2: The working process of the Middle Layer System

3. CONCLUSIONS AND FUTURE WORK

We proposed an approach for handling ontology changes by means of using change-tracks to eliminate or reduce any impact that ontology change can have on the application queries. We developed a prototype system that analyses the incoming queries, amends the entities within the queries according to the change information stored in the Log Ontology built to store and manage change information between ontology versions, and informs the end-user of any changes and actions taken. We showed that with the extra support of the middle layer, some of the queries that are targeting parts of the ontology that have changed can be updated and processed properly.

In our next stage work, Enabling *Log Ontology* to capture a series of changes between multiple versions of the same ontology would be a necessity to assist our system to cope with more complex changes. In addition, (semi-)automatic collecting ontology change information between ontology versions would make our system usable in a large scale. Providing more machine-processable formats, such as RDF or OWL, of the query result would be beneficial for agents to understanding the change information within Semantic Web-based applications.

4. ACKNOWLEDGMENTS

This work has been supported under the Advanced Knowledge Technologies Interdisciplinary Research Collaboration (AKT IRC), which is sponsored by the UK Engineering and Physical Science Research Council under grant number GR/N15764/01. Special thanks for the excellent technical supports from my colleague David Dupplaw (dpd@ecs.soton.ac.uk).

5. REFERENCES

[1] Heflin, J. and Hendler, J. Dynamic ontologies on the web. In *Proceeding of the 17th American Association for Artificial Intelligence Conference (AAAI)*, pages 443–449, Menlo Park, CA, US, 2000. AAAI/MIT Press.

[2] Huang, Z. and Stuckenschmidt, H. Reasoning with multi-version ontologies: A temporal logic approach. In *Proceeding of the 4th International Semantic Web Conference (ISWC)*, Galway, Ireland, 2005.

[3] Klein, M. and Fensel, D. Ontology versioning on the semantic web. In *Proceeding of International Semantic Web Working Symposium (SWWS)*, Stanford University, California, U.S.A, 2001.

[4] N. K. Klein, M., and Musen, M.A. Tracking changes during ontology evolution. In *Proceeding of the 3rd International Semantic Web Conference (ISWC2004)*, Hiroshima, Japan, November 2004.

[5] Noy, N.F., and Musen, M.A. Promptdiff: A fixed-point algorithm for comparing ontology versions. In *Proceeding of the 18th National Conference of Artificial Intelligence (AAAI)*, pages 744–750, Edmonton, Alberta, Canada, 2002.

[6] K. K. Ognyanov, D., and Fensel, D. Ontology versioning and change detection on the web. In *Proceeding of 13th International Conference on Knowledge Engineering and Management*, Siguenza, Spain, 2002.

[7] H. H. H. Stuckenschmidt, H., and Sure, Y. A framework for handling inconsistency in changing ontologies. In *Proceeding of the 4th International Semantic Web Conference (ISWC)*, Galway, Ireland, 2005.