MUTU: An Analysis Tool for Maintaining a System of Hierarchically Linked Ontologies

Sini Pessala, Katri Seppälä, Osma Suominen, Matias Frosterus, Jouni Tuominen, and Eero Hyvönen

Semantic Computing Research Group (SeCo)
Aalto University School of Science, Dept. of Media Technology, and
University of Helsinki, Dept. of Computer Science
http://www.seco.tkk.fi/, firstname.lastname@aalto.fi

Abstract. We consider ontology evolution in a system of light-weight Linked Data ontologies, aligned with each other to form a larger ontology system. When one ontology changes, the human editor must keep track of the actual changes and of the modifications needed in the related ontologies in order to keep the system consistent. This paper presents an analysis tool MUTU, by which such changes and their potential effects on other ontologies can be found. Such an analysis is useful for the ontology editors for understanding the differences between ontology versions, and for updating linked ontologies when changes occurred in other components of an ontology system.

1 Facilitating changes across ontologies

1.1 Position statement

Ontologies are often linked into larger ontology systems systems of a general upper ontology complemented by domain-specific ontologies. This enables that the experts of various domains can create their field-specific ontologies that co-operate with the other ontologies.

When updating the upper ontology, the changes require modifications in the domain ontologies. Since the domain ontology developers are not specialized in the upper ontology, the changes to it should be conveyed to the domain ontology developers in a readable form.

1.2 Introduction

Ontologies define concepts and relations between them in a machine-processable form [5]. The backbone of ontologies is typically based on *rdfs:subClass* relation, that can also be used in Linked Data for aligning ontologies hierarchically into larger systems. In such systems, more domain-specific ontologies extend the concepts of more general (upper) ontologies. This idea has been implemented in the Finnish Collaborative Holistic Ontology (KOKO)[1]. KOKO has the Finnish General Upper Ontology YSO [1] as its

[1] http://www.seco.tkk.fi/ontologies/koko/

upper ontology, aligned with 15 more specific ontologies in domains such as cultural heritage, agriculture and forestry, sea faring, photography, and public governance.

Domain ontologies are typically separately developed by different expert groups, since the process requires field-specific knowledge often found in separate communities and organizations. In KOKO, all component ontologies are based on existing keyword thesauri developed by such distributed, independent expert groups. However, the ontologies have mutual implicit relations with each other via shared concepts and relations. The added value of KOKO is to explicate such alignment relations with the upper ontology YSO, linking the ontologies into a coherent global hierarchical system.

This paper concerns the problems of ontology evolution in such a system of Linked Data ontologies. When an ontology has been edited, possibly by several persons in a team, the ontology editor needs tools for seeing the actual changes of the ontology with regard to the earlier versions and examining the possible consequences of the changes to other ontologies. We built an analysis tool MUTU for the ontology editor to address such changes. The idea in MUTU is to provide the domain ontology developer with an analysis of changes with regard to a prior version of her ontology, and support her in performing the required modifications, when changes have occurred in other related ontologies, in particular the upper ontology gluing the component ontologies together.

2 Tool Description: MUTU

2.1 Adressing the Changes of the Upper Ontology with MUTU

MUTU has been developed as an answer to a real demand in collaboration with ontology developers. MUTU lists the changes occurred at the update of the upper ontology as an HTML file, and marks the modified concepts of the updated upper ontology grouping them to different categories, based on the type of change. The groupings help the domain ontology developer to identify which concepts still need to be checked. After the developer has performed the required modifications to the domain ontology, the groupings can be removed. MUTU supports lightweight RDFS, SKOS[2] and OWL ontologies. MUTU is adapted for a new upper and domain ontology pair by creating a configuration file containing a list of the ontology namespaces and property URIs.

MUTU is based on the assumption that the URI of a concept is the same across different ontology versions, thus concepts are identifiable by their URIs. This means that when a label of the concept changes, no new concept is created.

Interesting Change Categories of the Upper Ontology MUTU divides the changes in an upper ontology into eight categories, which were determined in collaboration with ontology developers. A change can be either an addition or a removal of a concept or an addition, a removal or a replacement of a property value of a concept. It is assumed that the schema of the ontologies remains unchanged simplifying the process compared with a more general approach as in, e.g., [3]. However, all of the changes are not relevant to the developer of the domain ontology, since the domain ontology extends only some

[2] http://www.w3.org/2004/02/skos/

parts of the upper ontology. Thus, we define that a change in the upper ontology is interesting, if it will most likely cause update procedures to the domain ontology. This means that different subgroups of the upper ontology changes are interesting depending on the domain ontology and the connection between it and the upper ontology.

Generally, the changes nearest to the extended parts are interesting, but the domain ontology developer needs to examine all of the changes. By dividing the interesting and rest of the changes into different groups, the domain ontology developer could only check the interesting changes. The uninteresting changes are also available for acquiring a broader picture of the changes of the upper ontology.

The domain ontology is connected to the upper ontologies via RDF triples where the subject belongs to the upper ontology and the object to the domain ontology or vice versa. The concepts of the upper ontology participating in the described triples are called connecting concepts. Respectively, connecting concepts that are connected via an equivalency property are called equivalent-connecting concepts. Next, we introduce the change categories and define the interesting changes in these categories.

1. **Added and removed concepts:** Added and removed concepts are concepts that do not exist in both of the upper ontology versions. An added or removed concept is interesting when it is an ancestor of a connecting concept, since it should be checked that the reasoning of the hierarchy is valid for the connecting concepts of the domain ontology. Additionally, some added concepts might already exist in the domain ontology, so these concept pairs should be marked equal.
2. **Hierarchy and partonomy changes:** Hierarchy and partonomy changes are in the values of the hierarchy building properties, e.g., *rdfs:subClassOf* or *skos:broader*. The changes of this category are interesting if the concept is an ancestor of a connecting concept with a continuous chain of that property.
3. **Label changes:** Label changes are changes in the value of label-related properties, such as *skos:prefLabel* or *skos:altLabel*. The label changes are interesting, if the preferred label of an equivalent-connecting concept has been replaced, since then the preferred label of the domain ontology concept might need to be updated.
4. **Associativity and equivalency changes:** Associativity and equivalency changes are the changes of the properties that change additional connections between concepts. Examples of these properties are *skos:related* and *skos:exactMatch*. These changes are interesting if they occur in a connecting concept.
5. **Other changes:** Other changes contain the changes of the properties that were not mentioned above. They are interesting if they occur in a connecting concept.

Additional features In addition to the change categories listed above, the ontology developers requested additional features to aid their work. These features were reporting on multiple concepts with the same label and blocking of uninteresting changes in properties and concepts.

The label-matching feature compares all of the labels of the updated upper ontology with the labels of the domain ontology, detecting the unwanted situations of having the multiple occurrences of the same concept without any relation between them, e.g., *owl:sameAs*. If some labels match and there is no marked equivalency, then the domain ontology developer should see if the concepts should be marked equivalent. Additionally, the labels are lemmatized, since depending on the ontology, some labels might be in plural and the others in singular.

Blocking an uninteresting property means that the changes of that property are ignored, and blocking an uninteresting concept means that changes to the subconcepts of the concept are ignored. This prevents flooding the change listing with the modifications of irrelevant properties, whose change would not cause any need for updating the domain ontology. As an example, label changes in languages that are not used in the domain ontology are not relevant.

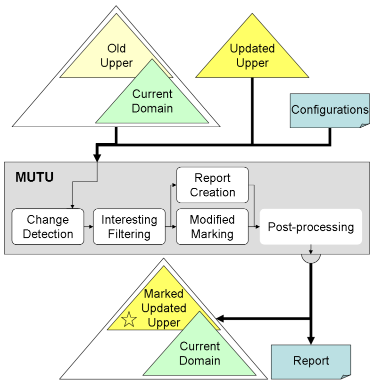

Fig. 1. Overview of the process of MUTU

MUTU Process The usage of MUTU is shown in Figure 1. The domain ontology developer inputs the current upper ontology, the old upper ontology, the domain ontology and a configuration file. MUTU detects the changes between the different versions of the upper ontology and detects the connecting concepts. Then MUTU separates the changes to interesting and the rest and prints this list to a report in HTML format. Finally, MUTU marks the modified concepts and outputs the HTML listing and marked upper ontology for the domain ontology developer.

The HTML listing contains the changes sorted by different modification types. By expanding one modification type, one can choose to expand either the interesting changes or the rest modifications and then see a list of the concepts with these changes and browse the changed property values. The marked upper ontology can be used in the ontology editor for keeping on track of which concepts are still unchecked. In the marked upper ontology, the changed concepts are marked as subclasses for structuring concepts representing different modification types. The changes are marked in the ontology so that the subconcepts of one modification type are the same as in the HTML listing.

2.2 Case Study: YSO and LIITO

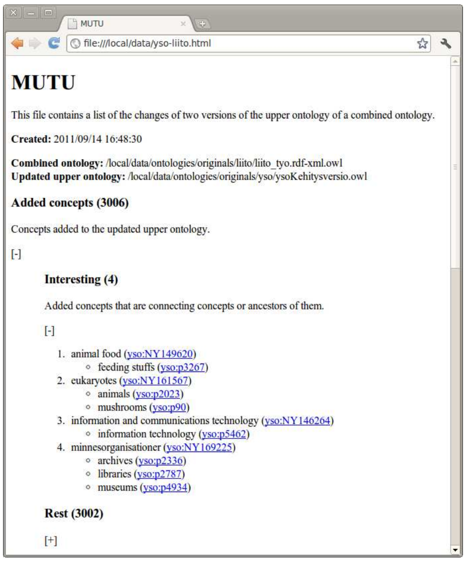

Fig. 2. The aFdded, interesting concepts of the HTML change list. The bulletin-listed concepts are the nearest connecting concept children of the added concept.

In our case study, we used the Finnish General Upper Ontology YSO [1] as the upper ontology and the Business Economy Ontology LIITO[3] as the domain ontology. Both ontologies are in active development and there is a new release of YSO quarterly. LIITO contains almost 3200 concepts, the old version of YSO contains 20700 concepts and the updated YSO contains 23600 concepts.

We have not yet conducted a formal evaluation of MUTU, but the first results of using MUTU with YSO and LIITO are promising. The outputs, the change list and marked updated upper ontology, were considered useful among the domain ontology developers. It was easy for the developers to see what changes had been made. A screenshot of the interesting added concepts are shown in figure 2.

When examining the change lists, some interesting findings were made: somewhat over 10% of the hierarchy changes occurred near to the connecting concepts. Most of these changes were of hierarchically close concepts. Another interesting finding was that since the upper ontology has not been updated before, over 500 concepts added to YSO that had labels similar to those in LIITO. This means that over 15% of the concepts of LIITO exist in the updated upper ontology without any marked connections to the concepts in the YSO. Nevertheless, this does not mean that all of these concepts

[3] http://onki.fi/en/browser/overview/liito

are duplicates, since for example “The Museum of Modern Arts” could mean a building in the upper ontology and an organisation in the domain ontology.

LIITO is developed using the Protégé[4] ontology editor, and for aiding the development process, the source file of LIITO contains a copy of YSO. The YSO copy does not automatically update when the original YSO updates, thus the old YSO version is replaced with a copy of the updated YSO version in the post-processing phase of MUTU.

LIITO and YSO contain labels in Finnish, Swedish, and English, so the label changes are divided into subgroups according to the languages to ease browsing.

3 Discussion and Related Work

Contributions This paper rises the issue of updating systems of separately-developed ontologies. To address this, we presented a tool for detecting the changes in an upper ontology and listing the relevant changes for the developer of the domain ontology. This supports the developer when reflecting the changes of the upper ontology into the domain ontology.

Related Work Changes and evolution of a single ontology is a widely researched area, see for example [4][7]. However, the situation where the changes in the upper ontology need to be reflected in the domain ontology has not been researched as extensively. Maedche & al. [2] discuss the evolution of distributed ontologies depending on other ontologies. They use change logging and when the domain ontology requests the modifications of the upper ontology, the changes of the upper ontology are merged to the change log of the domain ontology. Their main goal was to ensure the consistency of the ontologies, where as we are concerned of conveying the changes to the domain ontology developer.

Future work This paper described only the preliminary results of the idea and testing of MUTU, and its web interface is still under development. We intend to put the system to use with the domain ontologies of YSO. The web interface of the MUTU will be integrated into the ONKI ontology service[5] [8]. After the integration is complete, we will evaluate the application in collaboration with the actual domain ontology developers. The evaluation will be that the domain ontology developers use MUTU in their normal updating tasks and then analyze their use experiences with MUTU and give proposals for improvements.

In addition, we will enhance the change categories and interesting changes with the feedback and group single changes to more human-understandable composite changes similarly to the work of Stojanovic et al. [6]

[4] http://protege.stanford.edu/
[5] http://onki.fi/

Acknowledgements This work is part of the National Semantic Web Ontology project in Finland[6] FinnONTO (2003–2012), funded currently by the National Technology and Innovation Agency (Tekes) and a consortium of 35 public organizations and companies.

References

1. E. Hyvönen, K. Viljanen, J. Tuominen, and K. Seppälä. Building a national semantic web ontology and ontology service infrastructure—the FinnONTO approach. In *Proceedings of the European Semantic Web Conference (ESWC 2008)*, 2008.
2. A. Maedche, B. Motik, and L. Stojanovic. Managing multiple and distributed ontologies on the Semantic Web. *The International Journal on Very Large Data Bases (The VLDB Journal)*, 12(4):286–302, 2003.
3. A. Maedche, B. Motik, L. Stojanovic, and N. Stojanovic. User-driven Ontology Evolution Management. *Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web*, pages 133–140, 2002.
4. P. Plessers, O. De Troyer, and S. Casteleyn. Understanding ontology evolution: A change detection approach. *Web Semantics: Science, Services and Agents on the World Wide Web*, 5(1):39–49, 2007.
5. Staab S. and Studer R., editors. *Handbook on Ontologies (2nd Edition)*. Springer–Verlag, 2009.
6. L. Stojanovic, N. Stojanovic, and S. Handschuh. Evolution of the metadata in the ontology-based knowledge management systems. In *German Workshop on Experience Management*, volume 2002, pages 65–77, 2002.
7. M. Tury and M. Bieliková. An approach to detection ontology changes. *Workshop proceedings of the sixth international conference on Web engineering (ICWE 2006)*, 2006.
8. K. Viljanen, Jouni T., and E. Hyvönen. Ontology Libraries for Production Use : The Finnish Ontology Library Service ONKI. In *Proceedings of the 6th European Semantic Web Conference (ESWC 2009)*, 2009.

[6] http://www.seco.tkk.fi/projects/finnonto/