iMERGE: Interactive Ontology Merging

Zoulfa El Jerroudi and Jürgen Ziegler

University Duisburg-Essen, 47057 Duisburg, Germany

Abstract. In this paper we present novel visual analytics techniques which help the user in the process of interactive ontology mapping and merging. A major contribution will be the strong integration and coupling of interactive visualizations with the merging process enabling the user to follow why concepts are merged and at which position in the ontology they are merged. For this purpose, adapted ontology similarity measures and new techniques for representing ontologies will be required to enable responsive, real time visualization and exploration of the comparing and merging results.

1 Visual Ontology Mapping

The areas of ontology mapping and ontology merging have largely relied on automatic and semi-automatic methods in the past (FOAM [1], PROMPT [2], OLA [3] and FCA-Merge [4]), where user control and interaction is limited and results are typically only presented to the user at the end of some complex computational process. The effectiveness of these approaches can be increased in many application domains, if these approaches use the users knowledge and expertise in the comparing and merging process and support the explorative and iterative activities that are essential for the user's sensemaking process. Ontology merging is still largely a human-mediated process. The user could not trust in automatic results, where he does not know where and for which reason concepts are merged.

In this paper we present an approach, which helps to enable users to explore the ontology and to compare results in an intuitive and efficient manner. We aim to support the analytic comparing and merging process providing tightly linked and integrated techniques and views for visualizing and exploring the raw ontologies and derived merging results. For this purpose we develop a prototype editor iMerge. The introduced editor iMerge provides different views for this purpose(see Fig. 1).

The *SmartTree-View* [5] extends the conventional tree widget with a number of mechanisms facilitating ontology exploration and development. In addition to the hierarchical structure shown (typically the class hierarchy), non-hierarchical relations are shown dynamically upon selecting a node. SmartTree introduces *Condense+Expand* and *Prune+Grow*, two new interaction techniques allowing to hide and expose parts of the tree.

The *Matrix-View* [6] is suited for comparing two ontologies and determining where most of the mappings between ontologies occur. In the *Matrix-View* the

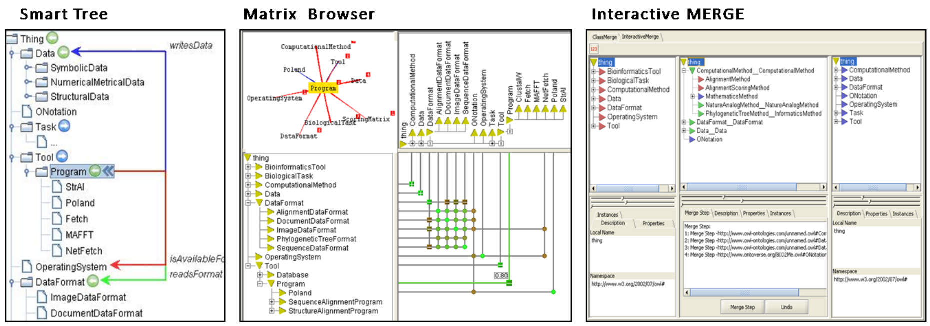


Fig. 1: Different Views of the ontology and the results of comparison

ontologies to be compared are confronted on the both axes of the matrix. High agreement in the ontologies are signified with green symbols at junctures, parts which are different are signalised with red symbols. A plus symbol in the matrix indicates that there are similar concepts hidden in the substructure. For the comparing process different algorithms can be selected by the user. Based on the results of comparison the ontologies can be merged.

The *InteractiveMERGE-View* supports merging of two ontology step by step and with leverage the users knowledge and expertise. For supporting this task, both ontologies are highlighted in different colors, so the user can register from which parts the changes comes from. The differences are shown first in the original ontology to the user and after that the consequences of the merging step are shown visually in the target ontology. The domain expert can *accept, change* or even *reject* this step. The alternative views ease the ontology designers to comprehend the consequences of their work.

Similarity measures for ontological structures have been widely researched, e.g. in [1], [2], [3] and [4]. A survey of ontology mapping is given in Falconer [7] and Choi [8]. The mentioned approaches do not give attention to the interactive aspects of ontology mapping and merging (except PROMPT [2]). Falconer et. al. [9] have defined requirements which should support the user in the cognitive tasks for ontology mapping and merging. The proposed iMerge editor considers the following important requirements:

- Support ontology exploration and manual creation of mappings
- Provide a visual representation of the source and target ontology
- Provide a method for the user to accept/reject a suggested mapping
- Provide access to full definitions of ontology terms
- Show the context of a term when a user is inspecting a suggested mapping
- Provide interactive access to source and target ontologies
- Support interactive navigation and allow the user to accept/reject mappings
- Provide progress feedback on the overall mapping process

Only the automatic verification of the supposed mapping is not done by the proposed editor. The iMerge proposes some mappings but does not consider possible conflicts which may occur if the concepts are merged.

2 Components of the iMerge-Editor

The structure of the proposed editor is divided in the units shown in Fig. 2.
Visualisation: In the first step of the visual ontology exploration the user needs

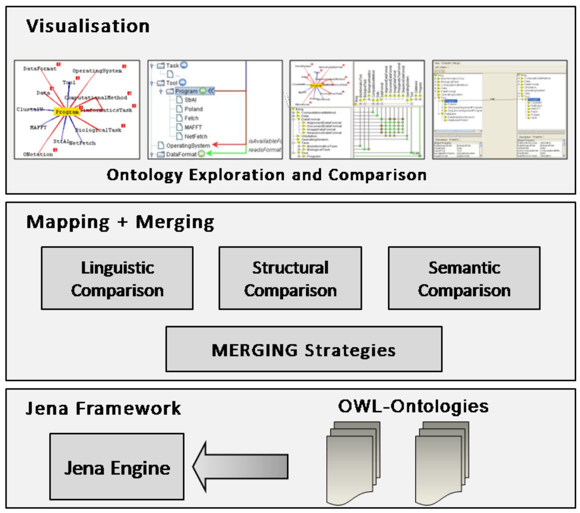


Fig. 2: Units of the iMerge editor

to get an overview of the ontologies, which have to be merged. Here, the user needs different access points (SmartTree, Graph-Viz) to the ontology, because he sets his exploration objectives in the most time during the interaction with and navigation in the ontology. The different views should be coupled in a way, that even if changing the view, the actual focus remains clear. After the visual exploration, the focus switches to the proposed mappings.

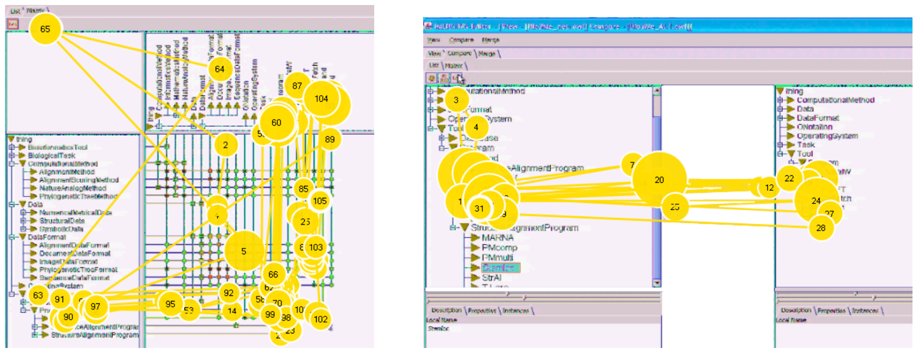
Mapping + Merging: This component provides methods for identifying mappings between the source and target ontology, which try to approximate the understanding of what the users consider to be a good match. For this purpose, our approach combines the results of several independently executed match algorithms.
The **linguistic approach** exploits text-based properties of the ontologies, such as name and description. With the method *EditDistance* [10] string similarity is computed from the number of edit operations (insertions, deletions, substitutions of single characters) necessary to transform one string into another one. As

an alternative the method *N-Gram* [11] can be used. Here, strings are compared according to their set of n-grams, i.e., sequences of n characters.
The **structural approach** exploits relationships between concepts that appear together in a structure. Usually, concepts and their relations are represented in a graph so that different kinds of structural related elements can be identified for matching. To estimate the similarity between two concepts, we can compare different kinds of their neighbor elements, such as the *parents, children*, or the *leaves* subsumed by them.
The **semantic approach** estimates the similarity between concepts based on their terminological relationships, such as synonymy, hypernymy, hyponymy. This approach requires the use of auxiliary sources, such as documents or annotations, in which the semantic relationship is captured. This method takes as input two ontologies and a set of documents which are linked with the concepts. We assume that, if documents annotated with concept a (of O_1) are similar to the documents annotated with concept b (of O_2), then the concepts a and b are similar.
Merging Strategy: This component develops step by step a new ontology G based on the preceding comparison, where the user can follow why the concepts are merged. First, concepts without a mapping pair are copied in the resulting ontology G. Concepts with a mapping candidate should be merged. For merging two concepts the user has to specify a threshold. Similar sub-concepts and properties with a similarity value higher than the threshold are merged recursively. The color indicates if a concept comes from O_1 or O_2.

3 Discussion and Further Work

Within this work informal usability tests are conducted, which give first hints for the correctness of the assumptions*(Users could not trust in automatically generated merging results, because they could not follow why concepts are merged).* Furthermore, the visual exploration of the mapping pairs has been compared with the eye tracking system Tobii T60 both in the *Matrix View* and the *List View* (see Fig. 3). The results of the eye tracking analysis confirm the assumption, that both tasks *get an overview* and *comprehend details* need different views. In the *List View* the fixations are concentrated only on the two concepts that are compared and the gaze motion goes between these both concepts. In the *Matrix View*, not only the comparing concepts are regarded, but also nearly concepts are considered. The defined matrix leads to consider concepts in the neighborhood. It is also visible, that in the *Matrix View* the number of fixations in the same time is higher, but the duration is smaller. In this view the user tries to get only a fast overview about the ontology.

Based on these results further work goes in coupling different views with the merging process more directly, e.g. conflicts during the merging process can be shown visually. Furthermore, we should consider the development and changes in the ontology over time especially for different ontology versions.



(a) Matrix View (b) List View

Fig. 3: Gaze Motion: *Matrix View* and *List View*

References

1. Ehrig, M.: Ontology Alignment: Bridging the Semantic Gap (Semantic Web and Beyond). Springer-Verlag New York, Inc., Secaucus, NJ, USA (2006)
2. Noy, N.F., Musen, M.A.: The prompt suite: interactive tools for ontology merging and mapping. Int. J. Hum.-Comput. Stud. **59**(6) (2003) 983–1024
3. Euzenat, J., Valtchev, P.: Similarity-based ontology alignment in owl-lite. In de Mántaras, R.L., Saitta, L., eds.: Proceedings of the 16th European Conference on Artificial Intelligence (ECAI-04), IOS Press (2004) 333–337
4. Stumme, G., Maedche, A.: FCA-MERGE: Bottom-Up Merging of Ontologies. In: Proc. of the 17^{th} International Joint Conference on Artificial Intelligence (IJCAI). (2001) 225–234
5. Hüsken, P., Ziegler, J.: Degree-of-interest visualization for ontology exploration. In Baranauskas, M.C.C., Palanque, P.A., Abascal, J., Barbosa, S.D.J., eds.: INTERACT (1). Volume 4662 of Lecture Notes in Computer Science., Springer (2007) 116–119
6. Ziegler, J., Kunz, C., Botsch, V., Schneeberger, J.: Visualizing and exploring large networked information spaces with matrix browser. In: : Information Visualisation, 2002. Proceedings. Sixth International Conference on. (2002) 361–366
7. Falconer, S., Noy, N., Storey, M.A.: Ontology mapping - a user survey. In Shvaiko, P., Euzenat, J., Giunchiglia, F., He, B., eds.: Proceedings of the Workshop on Ontology Matching (OM2007) at ISWC/ASWC2007, Busan, South Korea. (November 2007)
8. Choi, N., Song, I.Y., Han, H.: A survey on ontology mapping. SIGMOD Rec. **35**(3) (2006) 34–41
9. Falconer, S.M., Storey, M.A.D.: A cognitive support framework for ontology mapping. In Aberer, K., Choi, K.S., Noy, N.F., et al., D.A., eds.: ISWC/ASWC. Volume 4825 of Lecture Notes in Computer Science., Springer (2007) 114–127
10. Levenshtein, V.I.: Binary codes capable of correcting deletions, insertions, and reversals. Technical Report 8 (1966)
11. Damashek, M.: Gauging similarity with n-grams: Language-independent categorization of text. Science **267**(5199) (February 1995) 843–848