

# Exploiting Ontology Matching to Support Reuse in PURO-started Ontology Development

Marek Dudáš, Ondřej Zamazal, and Vojtěch Svátek

Department of Information and Knowledge Engineering,  
University of Economics, W. Churchill Sq.4, 130 67 Prague 3, Czech Republic,  
{marek.dudas|ondrej.zamazal|svatek}@vse.cz

## 1 Introduction

We have recently proposed an innovative method of ontological engineering [1]: easing OWL ontology development by first creating a model in a less constrained language called PURO, allowing the engineer to focus on defining what is to be described by the ontology in an example real-world situation. The model is then automatically transformed to OWL, while allowing the user to choose the OWL encoding style.<sup>1</sup> The result is then finalized in a common tool like Protégé. We present an experimental implementation demonstrating how the above-described proposal can be enhanced by exploiting ontology matching, allowing to *reuse entities* from existing ontologies (as best practice in ontology design) or searching for a *combination of existing ontologies* that could cover the desired domain. We used ontology matching to search for relevant entities in existing ontologies for the given PURO model, visualize them and enable their easy reuse in our PURO-started ontology development approach. The approach is similar to vocabulary reuse tools like LOVER [4], however, the coupling with PURO is novel.

## 2 Design and Implementation

To find out whether there are OWL entities in existing ontologies that could cover some part of the PURO model, we simply take the OWL fragment generated from the model, match it to as many ontologies as possible, and present the results in a user-friendly way. To increase the chance of finding a match, the matching is run for four different encoding style OWL variants generated from the PURO model. We match the OWL fragment to Linked Open Vocabularies (LOV).<sup>2</sup> The matching is implemented as a RESTful service and executed in two steps. First, to speed up the process, we select candidate ontologies from LOV using vocabulary term search available from the LOV API, considering all terms from the OWL fragment. Second, the ontology candidates are matched to the OWL fragment using the state-of-the-art ontology matching tool LogMap [3]. We use a cached LOV snapshot to gain speed and reliability.

---

<sup>1</sup> E.g., whether a relationship will be represented by a property or class membership.

<sup>2</sup> <http://lov.okfn.org/>

The PURO-to-OWL transformation is made via a series of SPARQL update queries. To allow that, the PURO model is first serialized into RDF. The queries use RDF annotations to keep track of which PURO entities have been transformed to which OWL entities. This allows translating the OWL-to-OWL correspondences produced by the matching service to PURO-to-OWL mappings and visualize the latter in the original PURO model. Namely, entities (represented by nodes in the PURO model node-link visualization) with available mappings to OWL are highlighted by lines encircling them and labeled with the ontology IRI where the entity has been found. This way the user can see which parts of the PURO model are potentially covered by which existing ontologies. The visualization can therefore be useful on its own, simply suggesting relevant ontologies that can cover a given situation modeled in PURO.<sup>3</sup>

A list of the mappings found for a PURO entity is displayed when the PURO entity node is selected. By selecting an OWL entity from the list, the mapping is stored in the RDF PURO serialization and visualized in the PURO model as a separate node of different color. When the PURO-to-OWL transformation is run, the mapped entities are used by the SPARQL queries for transformation of corresponding PURO entities, instead of creating new OWL entities.

### 3 Results so Far and Future Work

The proof-of-concept implementation, available online as part of the web-based tool OBOWLMorph,<sup>4</sup> suggests that the basic idea is valid. We have yet to evaluate whether matching to several OWL encoding style variants brings any improvement compared to using just a default one. Compared to manual visualization of local ontology coverage [2], automated visualization is less complete and imprecise, but could serve to bootstrap the manual approach.

*The research is supported by UEP IGA F4/28/2016. Ondřej Zamazal is supported by CSF 14-14076P.*

### References

1. Dudáš, M., Hanzal, T., Svátek, V., Zamazal, O.: OBOWLMorph: Starting ontology development from PURO background models. In: International Experiences and Directions Workshop on OWL. pp. 14–20. Springer (2015)
2. Dudáš, M., Hanzal, T., Svátek, V.: What can the ontology describe? visualizing local coverage in PURO modeler. In: VISUAL@EKAW (2014)
3. Jiménez-Ruiz, E., Grau, B.C.: Logmap: Logic-based and scalable ontology matching. In: International Semantic Web Conference. pp. 273–288. Springer (2011)
4. Schaible, J., Gottron, T., Scheglmann, S., Scherp, A.: LOVER: support for modeling data using linked open vocabularies. In: Proceedings of the Joint EDBT/ICDT 2013 Workshops. pp. 89–92. ACM (2013)

---

<sup>3</sup> We presented such a use case earlier [2], but only as an aid for the user allowing to highlight parts of the PURO model by hand. Now the same is done automatically.

<sup>4</sup> <http://goo.gl/IjTw1X>, please use login and password “om2016”.