

Making ontology documentation with LODE

Silvio Peroni¹, David Shotton², and Fabio Vitali¹

¹ Department of Computer Science, University of Bologna (Italy)

² Department of Zoology, University of Oxford (UK)

essepuntato@cs.unibo.it, david.shotton@zoo.ox.ac.uk, fabio@cs.unibo.it

Abstract. In this demo paper we provide a brief overview of the main features of **LODE**, the *Live OWL Documentation Environment*. LODE is an online service that generates a human-readable description of any OWL ontology, taking into account both the ontological axioms and the annotations, and ordering these with the appearance and functionality of a W3C Recommendations web page, namely as an HTML page with embedded links for ease of browsing and navigation.

Keywords: OWL ontologies, Web tool, ontology documentation

1 Introduction

Understanding the aim and extent of an ontology is one of the most important task of the Semantic Web. To this end, users usually start by consulting its human-readable documentation. A large number of ontologies, especially those used in the Linked Data world, have good comprehensive Web pages describing their theoretical backgrounds and the features of their developed entities. However, problems arise when we look at under-developed models, since natural language documentation is usually only published once an ontology has become stable. This approach is justifiable: writing proper documentation costs effort, and re-writing it every time the developing ontology is modified is not practical. For this reason, tools for the automatic online generation of HTML documentation from ontologies – e.g. Parrot [5], the Ontology Browser³, Neologism [1] – are critically important. Not only do they ease the task of creating effective ontology documentation, but they also enable this to be done earlier in the creation life-cycle.

In this paper we introduce the main features of *LODE*, the *Live OWL Documentation Environment*⁴ [3], a tool we developed to address the issue of the automatic production of ontology documentation. LODE is an online service that takes any OWL ontology or, more generally, an RDF vocabulary, and generates a human-readable HTML page designed for browsing and navigation by means of embedded links.

³ OWLDoc-based Ontology Browser: <http://owl.cs.manchester.ac.uk/browser/>.

⁴ LODE, the Live OWL Documentation Environment: <http://www.essepuntato.it/lode>.

The rest of the paper is structured as follows. In Section 2 we present LODE, highlighting its main characteristics and features, and in Section 3, we conclude the paper by sketching out the future developments of our work.

2 The Live Documentation Environment

LODE, the *Live OWL Documentation Environment* [3], is an XSLT-powered online service that automatically generates a human-readable description of an OWL ontology (or, more generally, an RDF vocabulary), taking into account both ontological axioms and annotations, and presents these with the appearance and functionality of a W3C Recommendations document.

LODE automatically extracts classes, object properties, data properties, named individuals, annotation properties, meta-modelling (punning), general axioms, SWRL rules and namespace declarations from any OWL or OWL 2 ontology, and renders them as ordered lists, together with their textual and graphic definitions, in a single human-readable HTML page designed for easy browsing and navigation by means of embedded links. LODE can be invoked with a number of optional parameters so as to limit or extend the final documentation produced. For instance, it is possible to take into account all the entities in the ontology closure and/or the inferred axioms. The following pseudo-URL describes how to call LODE:

`http://www.essepuntato.it/lode/optional-parameters/ontology-url`

Fig. 1 illustrates the alternative ways to build the URL to call LODE and the related modules used⁵. The main features of LODE are introduced in the following paragraphs.

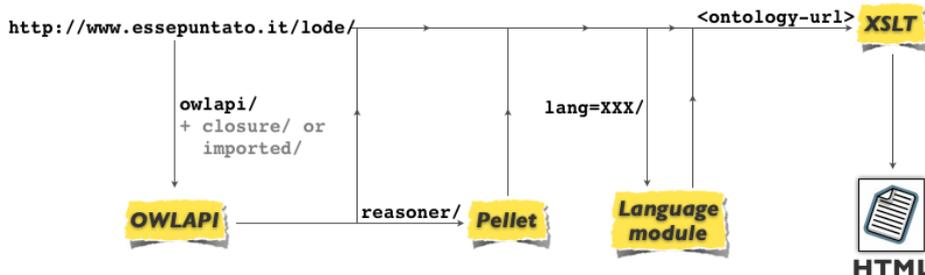


Fig. 1. All the possible ways, according to specific needs, for making a request to LODE.

Single page of documentation. The entire documentation is displayed in a single HTML page that starts with the title of the document (*dc:title* or

⁵ Note that routine use of the optional parameter “owlapi” is strongly recommended.

rdfs:label) and then lists the metadata of the ontology, such as its IRI, authors (*dc:creator* annotations), date of publication (*dc:date*), and imported ontologies (*owl:imports* axioms). A brief abstract (*rdfs:comment*) and an automatically-built Table of Content follow. When *dc:description* assertions are specified for the ontology, the section “Introduction” is created as the first item after the Table of Content. Then, LODE renders all the classes, object, data and annotation properties, named individuals, general axioms and SWRL rules, and concludes the document with a list of the namespaces defined in the ontology.

Displays information about imported ontology. All the annotations and axioms defined in the imported ontology are fully rendered in the documentation when one of the parameters “imported” or “closure” is specified.

Displays of axioms inferred by a reasoner. All the axioms inferable by Pellet [4] are fully rendered in the documentation when the parameter “reasoner” is specified. However, this option has not yet been fully tested.

Axioms descriptions fully displayed. All the axioms and restrictions are rendered within each entity definition through the Manchester Syntax [2], as shown in Fig. 2a⁶.

Cool URIs to call the service. Any LODE URL built as shown in Fig. 1 is compliant with Tim Berners-Lee’s “Cool URI” definition⁷. This enables LODE to be used in conjunction with content-negotiation mechanisms, so as to produce HTML representations of ontologies and their entities automatically when accessing ontologies from a web browser.

Images can be included in the documentation. The annotation property *dc:description* can be used to refer to graphic files in addition to specifying simple text. In this case, LODE is able to render these graphic files as part of the documentation itself, as shown in Fig. 2a.

Permits choice of language in which to display the documentation. The text of the documentation can be rendered in a specific language when the parameter “lang” is specified. The selected language will be used as preferred language instead of English (the default) to render the documentation, as shown in Fig. 2b. Of course, this presupposes that appropriate language annotations are present in the ontology.

3 Conclusions

In this demo paper we briefly overview the main features of **LODE**, the *Live OWL Documentation Environment*, our tool that automatically produces HTML documentation of OWL ontologies. We recently performed two different user evaluations so as to assess the usability of LODE when users use it to deal with ontology understanding and navigation tasks. Preliminary results, presented in [3], show that LODE is perceived as a usable tool that favourably compares

⁶ The screenshots shown in Figure 2 of this paper refers to the documentation produced by LODE for the Argument Model Ontology (<http://www.essepuntato.it/2011/02/argumentmodel>).

⁷ Cool URIs don’t change: <http://www.w3.org/Provider/Style/URI>.

(a)

(b)

Fig. 2. Screenshots of the documentation produced by LODE. (a) The way the class “argument” is rendered by LODE: its IRI, a brief abstract (*rdfs:comment* assertion), its axioms and a graphic representation (*dc:description* assertion pointing to a PNG file) of the class itself. (b) The Argument Model Ontology rendered in English (parameter “lang=en”) and Italian (parameter “lang=it”).

with similar applications such as Parrot [5] and the OWLDoc-based Ontology Browser³. In the future, we plan to widen the evaluation of the outcomes of the aforementioned user testing sessions, and to extend LODE features to include suggestions highlighted by users, such as adding a tree-browsing capability and a faceted free-text search box, thus significantly enhancing its usability for single ontologies in the one area in which LODE is presently lacking, search functions.

References

1. Basca, C., Corlosquet, S., Cyganiak, R., Fernández, S., Schandl, T. (2008). Neologism: Easy Vocabulary Publishing. In Proceedings of the 4th Workshop on Scripting for the Semantic Web. <http://ceur-ws.org/Vol-368/paper10.pdf> (last visited July 30, 2012).
2. Horridge, M., Patel-Schneider, P. (2009). OWL 2 Web Ontology Language: Manchester Syntax. W3C Working Group Note, 27 October 2009. World Wide Web Consortium. <http://www.w3.org/TR/owl2-manchester-syntax/> (last visited July 30, 2012).

3. Peroni, S., Shotton, D., Vitali, F. (2012). The Live OWL Documentation Environment: a tool for the automatic generation of ontology documentation. To appear in Proceedings of the 18th International Conference on Knowledge Engineering and Knowledge Management (EKAW 2012). <http://palindrom.es/phd/wp-content/uploads/2012/05/lode-ekaw2012.pdf> (last visited July 30, 2012).
4. Sirin, E., Parsia, B., Grau, B.C., Kalyanpur, A., Katz, Y. (2007). Pellet: A practical OWL-DL reasoner. In *Journal of Web Semantics*, 5 (2): 51-53. DOI: 10.1016/j.websem.2007.03.004.
5. Tejo-Alonso, C., Berrueta, D., Polo, L., Fernandez, S. (2011). Metadata for Web Ontologies and Rules: Current Practices and Perspectives. In Proceedings of the 5th International Conference on Metadata and Semantic Research (MTSR 2011). DOI: 10.1007/978-3-642-24731-6_6.