A framework to solve the ontology translation problem

Oscar Corcho

Facultad de Informática. Universidad Politécnica de Madrid
Campus de Montegancedo, s/n. 28660 Boadilla del Monte. Madrid. Spain
ocorcho@fi.upm.es

Abstract. Ontologies are developed with different tools and languages. Reusing an ontology usually requires transforming it from its original format to a target format. However, many problems usually arise in these transformations, related to the compatibility among tools/languages. We propose an ontology reengineering methodology (with its technological support) as a solution to the ontology translation problem. . . .

1 Introduction

Nowadays, different tools exist for developing ontologies: OILEd, OntoEdit, Ontolingua, OntoSaurus, Protégé2000, WebODE, WebOnto, etc. Each tool has their own knowledge model, and usually allows exporting/importing ontologies in their own textual representation.

Several languages are also used for the implementation of ontologies, such as Ontolingua, LOOM, OCML, FLogic, XOL, SHOE, RDF(S), OIL, DAML+OIL, etc. Apart from their lexical and syntactical differences, there are also more significant ones due to the knowledge representation (KR) formalism in which they are based (frames, semantic nets, description logic, etc.) and the semantics of their representation primitives and constructs, which fully determine both their expressiveness and reasoning capabilities.

The ontology translation problem appears when we decide to reuse an ontology (or part of an ontology) using a tool or language that is different from those ones in which the ontology is available. If we force each ontology developer, individually, to commit to the task of translating and incorporating the necessary ontologies to their systems, they will need both a lot of effort and a lot of time to achieve their objectives. Therefore, ontology reuse in different contexts will be highly boosted as long as we provide automatic ontology translation services among those languages and/or tools.

2 Characterisation of the ontology translation problem

The first reference to this problem was presented by Gruber in [2]. He proposed, as a solution, to follow a set of *ontological commitments* when an ontology was created. From all these ontological commitments, the ”minimal encoding bias” deserves special attention: Gruber proposed to conceptualise ontologies in the knowledge level, instead of doing it in the symbolic level, and to implement them using automatic translators. However, this criterion has not been commonly followed in ontology development, forcing ontology developers to translate existing ontologies manually or create ad-hoc translators between languages or knowledge models, which is a time consuming task.

Translation problems can be classified as follows:

Lexical problems. They appear when the terms used for language identifiers, texts and constructs (names of components, sizes of their textual descriptions, etc.) follow different conventions in the different languages and/or tools. For instance, concept National Park in Ontolingua is usually written as *National-Park*, while in FLogic hyphens are not allowed inside identifiers (hence, it is written as *NationalPark*).

Syntax problems. Different languages/tools use different grammars to represent their components. Some languages/tools also allow defining the same component in different ways. When performing translations, both situations must be taken into account.

Expressiveness problems. These problems are caused because different languages/tools are based on different KR paradigms. First, not all the languages allow expressing the same knowledge: we must analyse what components can be translated directly from a language to another one, what components can be expressed using other components from the target language, what components cannot be expressed in the target language, and what components can be expressed, although with losses of expressiveness.

Reasoning problems. The existence or not of an inference engine for a language, and the characteristics of this inference engine, usually bias the implementation of an ontology.

3 The framework: WebODE and OntoDialect

We propose to solve the translation problem in the context of a methodology for ontological reengineering. We distinguish three main phases: *reverse engineering* (we transform automatically an ontology that has been coded in a language to a knowledge model that is independent from the implementation); *reestructuration* (performed in the knowledge level, in accordance to the future uses of the ontology in an application); and *implementation* (we transform automatically the reestructured ontology into the target language).

This methodology is technologically supported by the WebODE ontology-engineering workbench [1]. Reestructuration is currently performed manually, with the WebODE ontology editor. Translators from WebODE to ontology languages/tools and vice versa can be created with the OntoDialect system, integrated in the workbench.

References

1. Arpírez JC, Corcho O, Fernández-López M, Gómez-Pérez A. WebODE: a scalable workbench for ontological engineering. First International Conference on Knowledge Capture (KCAP2001). Victoria. Canada. October, 2001.
2. Gruber R. A translation approach to portable ontology specification Knowledge Acquisition. #5: 199-220. 1993