The Role of Foundational Ontologies for Domain Ontology Engineering: An Industrial Case Study in the Domain of Oil and Gas Exploration and Production

Giancarlo Guizzardi, NEMO Group, Federal University of Espírito Santo, Brazil
Fernanda Baião, NP2Tec/Federal University of the State of Rio de Janeiro (UNIRIO), Brazil
Mauro Lopes, NP2Tec/Federal University of the State of Rio de Janeiro (UNIRIO), Brazil
Ricardo Falbo, NEMO Group, Federal University of Espírito Santo, Brazil

ABSTRACT

Ontologies are commonly used in computer science either as a reference model to support semantic interoperability, or as an artifact that should be efficiently represented to support tractable automated reasoning. This duality poses a tradeoff between expressivity and computational tractability that should be addressed in different phases of an ontology engineering process. The inadequate choice of a modeling language, disregarding the goal of each ontology engineering phase, can lead to serious problems in the deployment of the resulting model. This article discusses these issues by making use of an industrial case study in the domain of Oil and Gas. The authors make the differences between two different representations in this domain explicit, and highlight a number of concepts and ideas that were implicit in an original OWL-DL model and that became explicit by applying the methodological directives underlying an ontologically well-founded modeling language.

Keywords: Conceptual Modeling, Oil and Gas Domain, Ontologies, Ontology Languages, OWL-DL Model

INTRODUCTION

Since the word ontology was mentioned in a computer related discipline for the first time (Mealy, 1967), ontologies have been applied in a multitude of areas in computer science. The first noticeable growth of interest in the subject in mid 1990’s was motivated by the need to create principled representations of domain knowledge in the knowledge sharing and reuse community in Artificial Intelligence (AI). Nonetheless, an explosion of works related to the subject only happened in the past decade,
highly motivated by the growing interest on the Semantic Web, and by the key role played by ontologies in that initiative.

There are two common trends in the traditional use of the term ontology in computer science: (i) firstly, ontologies are typically regarded as an explicit representation of a shared conceptualization, i.e., a concrete artifact representing a model of consensus within a community and a universe of discourse. Moreover, in this sense of a reference model, an ontology is primarily aimed at supporting semantic interoperability in its various forms (e.g., model integration, service interoperability, knowledge harmonization, and taxonomy alignment); (ii) secondly, the discussion regarding representation mechanisms for the construction of domain ontologies is, typically, centered on computational issues, not truly ontological ones.

An important aspect to be highlighted is the incongruence between these two trends. In order for an ontology to be able to adequately serve as a reference model, it should be constructed using an approach that explicitly takes foundational concepts into account; this is, however, typically neglected for the sake of computational complexity.

The use of foundational concepts that take truly ontological issues seriously is becoming more and more accepted in the ontological engineering literature, i.e., in order to represent a complex domain, one should rely on engineering tools such as design patterns, computational environments, modeling languages and methodologies that are based on well-founded ontological theories in the philosophical sense (e.g., Burek, 2006; Fielding, 2004). Especially in a domain with complex concepts, relations and constraints, and with potentially serious risks which could be caused by interoperability problems, a supporting ontology engineering approach should be able to: (a) allow the conceptual modelers and domain experts to be explicit regarding their ontological commitments, which in turn enables them to expose subtle distinctions between models to be integrated and to minimize the chances of running into a False Agreement Problem (Guarino, 1998); (b) support the user in justifying their modeling choices and providing a sound design rationale for choosing how the elements in the universe of discourse should be modeled in terms of language elements.

This marks a contrast to practically all languages used in the tradition of knowledge representation and conceptual information modeling, in general, and in the semantic web, in particular (e.g., RDF, OWL, F-Logic, UML, EER). Although these languages provide the modeler with mechanisms for building conceptual structures (e.g., taxonomies or partonomies), they offer no support neither for helping the modeler on choosing a particular structure to model elements of the subject domain nor for justifying the choice of a particular structure over another. Finally, once a particular structure is represented, the ontological commitments which are made remain, in the best case, tacit in the modelers’ mind; in the worst case, even the modelers and domain experts remain oblivious to these commitments.

An example of an ontologically well-founded modeling language is the version of UML 2.0 proposed in (Guizzardi, 2005) and, thereafter, dubbed OntoUML. This language has its real-world semantics defined in terms of a number of ontological theories, such as theory of parts, of wholes, types and instantiation, identity, dependencies, unity, etc. However, in order to be as explicit as possible regarding all the underlying subtleties of these theories (e.g., modal issues, different modes of predication, higher-order predication), this language strives for having its formal semantics defined in a logical system as expressively as possible. Now, as well understood in the field of knowledge representation, there is a clear tradeoff between logical expressivity and computational efficiency (Levesque & Brachman, 1987). In particular, any language which attempts at maximizing the explicit characterization of the aforementioned ontological issues risks sacrificing reasoning efficiency and computational tractability. In contrast, common knowledge representation and deductive database languages (e.g., some instances of Description Logics) have been spe-
Related Content

Classification-Based Optimization of Dynamic Dataflow Programs
www.irma-international.org/chapter/classification-based-optimization-of-dynamic-dataflow-programs/108448/

Eliciting Security Requirements for an Information System using Asset Flows and Processor Deployment
www.irma-international.org/article/eliciting-security-requirements-for-an-information-system-using-asset-flows-and-processor-deployment/83634/

Web Service Evaluation Using Probabilistic Models
www.irma-international.org/chapter/web-service-evaluation-using-probabilistic-models/188255/

Software Metrics, Information and Entropy
www.irma-international.org/chapter/software-metrics-information-entropy/28114/

Social Business: A Way to Promote Organizational Transformation
www.irma-international.org/article/social-business/142516/