Computer Languages, Systems & Structures 54 (2018) 183–198

Contents lists available at ScienceDirect

Computer Languages, Systems & Structures

journal homepage: www.elsevier.com/locate/cl

Attributed and n-ary relations in OWL for knowledge modeling

Asmaa Chebba[a,1,*], Thouraya Bouabana-Tebibel[a], Stuart H. Rubin[b]

[a] Laboratoire de la Communication dans les Systèmes Informatiques, Ecole nationale Supérieure d'Informatique, BP 68M, 16309, Oued-Smar, Alger, Algérie.

[b] Space and Naval Warfare Systems Center Pacific, San Diego, CA 92152-5001, USA

ARTICLE INFO

Article history:
Received 27 December 2017
Revised 12 May 2018
Accepted 12 June 2018
Available online 22 June 2018

Keywords:
Knowledge modeling
Attributed relation
N-ary relation
Ontology
OWL
Protégé
Randomization
Reuse

ABSTRACT

Knowledge modeling is the basis for the whole process of conceptual design. We aim, in this paper, at representing knowledge through an efficient model that enables efficient reuse. The knowledge model must be both expressive, so as not to omit any pertinent knowledge, and computable to allow for reasoning and learning. The ontology model, along with its well-known description language OWL (Web Ontology Language), is commonly used for this purpose. However, despite the strong expressiveness of OWL, this ontology language still shows limitations with regard to some modeling needs. We aim at promoting the representation of such aspects. We especially deal with the lack of an adequate representation for attributed relations and n-ary relations. The attributed relation, which is a relation that possesses its own attributes, is introduced into OWL, while preserving the basic semantics. The n-ary relation is also added to OWL by introducing a supplementary representational layer that is defined with a sound semantics, inspired from the DLR logic.

1. Introduction

Conceptual design is a crucial task that influences the quality and the overall characteristics of the product. It represents the highest cost stage in terms of effort and time. Its improvement is necessarily reflected in the whole development process. This can be achieved through knowledge modeling, which is domain specific and allows for deducing best design practices, correcting the designs, and enhancing their quality.

An appropriate representation of the target domain's specific knowledge allows for efficiently assisting designers. An efficient reuse of knowledge, which concerns the previous design experiences or the amplified knowledge using randomization, enhances the quality of design by sparing time and effort by way of reuse. Reuse of existing validated conceptual schemas reduces the cost and time of development, since less effort is devoted to the processes of acquisition, specification, and validation. Randomization defines the basis sets for any process supporting reuse and thus the dependent sets.

* Corresponding author.
E-mail addresses: a_chebba@esi.dz (A. Chebba), t_tebibel@esi.dz (T. Bouabana-Tebibel), stuart.rubin@navy.mil (S.H. Rubin).

[1] Laboratoire de la Communication dans les Systèmes Informatiques, Ecole nationale Supérieure d'Informatique, BP 68M, 16309, Oued-Smar, Alger, Algérie. http://www.esi.dz.

https://doi.org/10.1016/j.cl.2018.06.001
1477-8424/

1.1. Involved paradigms: knowledge and reuse

The representational model used for the purpose of reusing conceptual design knowledge is expected to faithfully express the knowledge [1]. When formally represented, knowledge can be inferred, learned, shared, and reused [2]. The better the representation, the more likely these tasks can be achieved – and, in an optimal fashion. Enabling knowledge reuse and knowledge amplification (i.e., the transformative creation of symmetric knowledge) based on randomization [3–6] within the representational model is a major aim of our work.

Reuse in design is an emerging approach for enhancing productivity, maintainability, and quality [2]. Resorting to previous experiences, rather than relying on predefined rules and methods, is often the approach to solving design problems – and justifies the use of case-based reasoning (CBR). The CBR system helps designers, within a specific domain of interest, to solve new design problems by reusing the prior design experiences – cases of the CBR system. Based on the new problem description, the system identifies similar previously solved problems and retrieves their corresponding solutions for possible adaptation. Then, the resulting solution is proposed to the user who may revise it in the light of the current problem situation; he may completely or partly reuse its knowledge.

Along with reuse of previously stored design knowledge, we aim at generalizing/amplifying this knowledge using randomization. Implicit knowledge can be present in the knowledge base through dependencies and similarities between conceptual designs. By transmutation using analogy among the segments of information [3] – resulting in a methodology for the induction of knowledge based on randomization, new knowledge can be induced, and then reused itself.

Our investigations led us to the ontological model as pertinent support for knowledge representation. Ontologies play a key role in many application areas such as intelligent information reuse and integration [7–10] and knowledge-based systems [11]. They serve to share a common understanding of information among people or software agents, about a specific domain, where ontologies are described with a well-defined syntax and semantics [12]. The ontology allows for an advanced formal representation that is both expressive and computable. It supports sharing and reusability and is very popular, which promotes its constant evolution. Ontologies allow for reasoning about represented knowledge, which is semantically advantageous as there is a consensus on the concept's semantics. They also help to separate the domain knowledge from the operational knowledge for more efficient reasoning. Within the ontology, we organize our knowledge model into two levels of abstraction, which allows for a better management of the knowledge saved in the CBR case base. The first level describes general knowledge about the considered domain of interest for the conceptual design, whilst the second level represents the different design experiences. It is intended by the use of the ontology A faithful representation of the knowledge about the specific domain of expertise is intended through the use of the ontology. For that, a direct mapping between the CBR case constructs and the ontology language constructs used to represent this knowledge is applied. In addition, the ontology allows for the representation of dependencies/similarities between knowledge. Based on these dependencies (e.g. equivalence), knowledge derivation is hence possible through randomization and transformation.

The ontology is described through an ontology description language. The richest and most used language currently, proposed by the W3C working group, is OWL [13,14].

1.2. Motivation

Despite the well-known expressiveness of OWL, we noted through our numerous modeling tasks that some specific representation needs are not properly supported on the knowledge model. We address, through this work, the limitations regarding the relationship construct. The first deficit is the absence of a construct for the modeling of a relation incorporating its own attributes. We call such a relation an *attributed relation*. When a relation can be described by means of attributes, the latter should be attached to the relation, since they provide specific information about it, thus enhancing its semantics. The use of attributed relations, as part of the ontology, to describe the domain of knowledge is thus of great interest. Our second interest lies in the n-ary relation. OWL is known to support only binary relations between classes, which is a significant limitation. Representing relations with an arbitrary arity is very common and useful in modeling. In practice, both attributed relation and n-ary relation are usually modeled using reification. A *reified relation*, as defined in [15,16], is a relation transformed into a class.

Alternative modeling schemas to the attributed relation may be provided by the current OWL constructs. The relation is reified using the class construct, thus enabling the representation of the relation attributes. The latter are associated to the class of reification. Next, the ObjectPropertyChain construct can be used to specify, as a single relation, the relations added to link the new class to the original relation-ends. In the same way, reified relations are also used to overcome the OWL limitation regarding the n-ary relation modeling. The n-ary relation is codified as a class along with some extra work required to apply this reification.

Modeling using reification can be impractical in many cases. First, this solution is not intuitive and acts on a level of abstraction different from that of relations, which involves additional constraints when reasoning on the ontology for knowledge reuse, amplification, and extraction. For example, in attributed relations modeling, it is often better for information extraction to model the attributes as arguments of the relation, instead of burying them in new structures. Second, such a workaround modeling remains a heavy and/or improper solution since it calls for extra classes, axioms, and individuals