

MDA and Semantic Web Technologies for Assessment Systems

Sonja Radenković, Nenad Krdžavac, Vladan Devedžić

FON – School of Business and Administration,
University of Belgrade, Serbia
sonjam@fon.bg.ac.yu, nenadkr@tesla.rcub.bg.ac.yu, devedzic@etf.bg.ac.yu

Abstract. This paper discusses how Semantic Web-based knowledge representation and Model Driven Architecture can be brought together in designing an assessment system. The proposed system is based on the IMS Question and Test Interoperability standard. The central idea is to use Description Logic reasoning techniques for intelligent analysis of students' solutions of simple choice questions problems during assessment sessions with the system.

Keywords: Assessment Systems, Description Logics, Model Driven Architecture, Semantic Web

1 Introduction

In assessment systems the huge possibilities of the Semantic Web (SW) [3] for data representation and exchange are limited by the fact that the end user still needs to take care of the data.

This paper proposes a way to create a flexible, interoperable assessment system that is easy to maintain and reuse. It is based on the IMS Question and Test Interoperability (QTI) standard [5] and designed using the Model Driven Architecture (MDA) standards [8]. The core concept here is to change the system's specification rather than implementation using the Unified Modeling Language [10].

One of the main ideas the paper proposes is to use Description Logics (DL) reasoning techniques [2] for intelligent analysis of students' answers to and solutions of the problems they are working on during assessment sessions with the system. The use of a DL reasoner enables processing of simple choice questions. This is the way for applying a framework for data sharing and reuse across heterogeneous applications, which is the core of the Semantic Web. The paper is organized as follows. The next section describes the model of the QTI-based assessment system using the MDA standards. Section 3 describes reasoning with QTI models. The last section shows the conclusions and indicates directions for future work.

2 Modeling the QTI-based Assessment System Using MDA Standards

The main reason for applying the MDA standards [8] in the development of assessment systems is to make a clear difference between conceptual and concrete modeling, in order to automate transfer and sharing of information and knowledge.

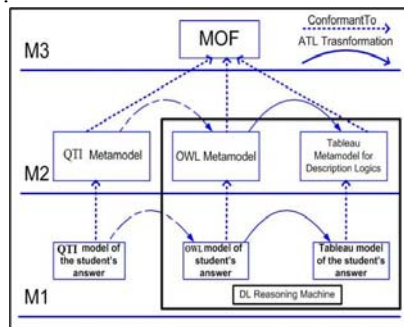


Fig. 1. Analysis of students' solutions [5]



Fig. 2. A simple choice item example

The first step in developing a QTI system using the MDA standard is to create a metamodel that captures the main concepts of QTI (Fig. 1) [9].

Having the QTI-based metamodel that is located at the M2 level of the MDA hierarchy (Fig 1), we can create models that correspond to a given metamodel. There are a lot of examples of *assessmentItems* that are proposed in the IMS QTI standard (see [7]). In order to illustrate the creation of models that correspond to a given QTI metamodel, we present as an example the Simple Choice item shown in Fig. 2.

In order to use a DL reasoning techniques in the process of analyzing students' solutions, it is necessary to perform the transformation of the QTI-based models into the equivalent QTI-OWL model (Fig. 1).

This process is automated using the Atlas Transformation Language (ATL) [1]. The result of *qti2owl.atl* transformation is the QTI metamodel as well as the QTI models in Ecore that are OWL based (Fig. 4).

3 Reasoning with QTI Models

In this section, we focus on the intelligent analysis of the semantics of students' solutions to the problems they solve during assessment (students' solutions, for short). For the explanation of our idea, we present two very easy examples of using a DL reasoner in analyzing a student's answer, using a simple choice item (Fig. 3 and 4).

3.1 Examples of Application of DL Reasoning in Intelligent Analysis of Student's Solutions

Among a few possible answers, a student may choose one or more. Using DL terms, the items correspond to TBox (Rbox) (Table 2), where questions are presented as a concept using a description logic language.

If the student choice is one among a few answers, the reasoner can check if this answer is satisfiable w.r.t the knowledge base (OWL model). QTI standard proposes possibility of saving answers in "ReponseDeclaration" [IMS QTI, 2006] (Fig. 3).

Table 2. DL expressions correspond to the OWL model (Fig. 5)

DL expressions	Formula number
$(\text{ChoiceB} \sqcap \text{ChoiceC}) \sqsubseteq \text{ChoiceA}$	(1)
$\text{ChoiceA} \sqsubseteq (\neg \text{ChoiceB} \sqcup \neg \text{ChoiceC})$	(2)

Table 2 presents how an OWL model (Fig. 4) can be described as Tbox axioms (1), i.e. only question, without answer. It means that the OWL model (Fig. 4) is satisfiable if and only if (iff) formula (1) is satisfiable. It means that formula (1) is satisfiable iff formula (2) is so. In this case it is union of three concepts.

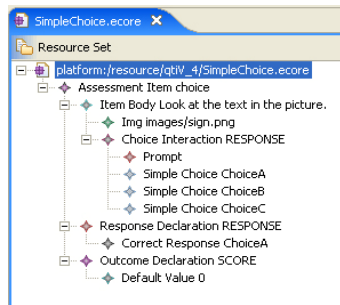


Fig. 3. Simple choice QTI model

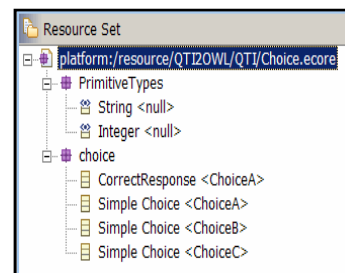


Fig. 4. OWL model transformed from QTI model

As a solution, the reasoner generates tableau models (conforms to tableau metamodel [6]) for both cases. The reasoning process is done during the model transformations, and it is the main difference between our reasoner and the existing ones, like FACT [4]. It means that the tableau model saves implicit knowledge of the reasoning process, i.e. using tableau models we may find out useful information about student's semantic mistakes. We can use the reasoner to test how some pedagogical strategies can help in student learning.

Example of Unsatisfiable Student's Answers. Suppose that a student submitted "ChoiceA" as an answer with the question presented on Fig. 4. The reasoner takes this answer as OWL class (model), and calculates (un)satisfiability of the class with

respect to the OWL model (Fig 4) and generates a tableau model as presented on Fig. 6. For explanation reasoning mechanisms on OWL models, we used DL notation.

$\text{ChoiceA} \sqsubseteq \text{ChoiceA} \sqcup (\neg \text{ChoiceB} \sqcup \neg \text{ChoiceC})$	(2)
$(\text{ChoiceA} \sqcap \neg \text{ChoiceA}) \sqcap (\text{ChoiceB} \sqcap \text{ChoiceC})$	(3)

Checking subsumption can be reduced to the satisfiability of concepts [4]. Constraint system as a beginning point in reasoning processes can be presented as a finite set of classes (models), as follows:

$L(x) = \{ \text{ChoiceA} \sqcap \neg \text{ChoiceA} \sqcap (\text{ChoiceB} \sqcap \text{ChoiceC}) \}$	(4)
--	-----

Individual “x” (Fig. 6) is an instance of all subconcepts in this set. Using reasoning rules (in this case it is an intersection rule for ALC logic) [4], the constraint system, described by formula (4), can be extended in new ones:

$L(x) = \{ \text{ChoiceA}, \neg \text{ChoiceA} \} \cup L(x)$	(5)
--	-----

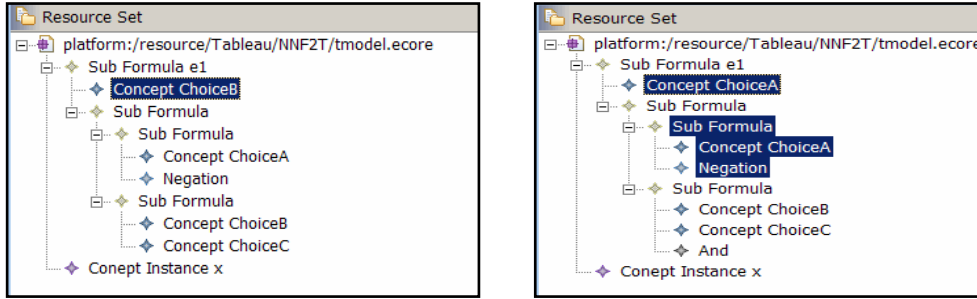


Fig. 5. Tableau model of SAT answer **Fig. 6.** Tableau model of unSAT student’s answer

The constraint system consists of clash and formula (3) is not satisfiable, which implies that the question subsumes an answer and the student gives a true answer. Unsatisfiable points of the beginning model are indicated by blue color in Fig. 6.

Example of Satisfiable Student’s Answers. Suppose that the student submitted a wrong answer. It means that the student submitted “**ChoiceB**” or “**ChoiceC**”. The reasoning process goes as follows: The question (similar as in the true case) is whether the answer is subsumed by the question. This can be described by formula (6).

$(\text{ChoiceB} \sqcup \text{ChoiceC}) \sqsubseteq \text{ChoiceA} \sqcup (\neg \text{ChoiceB} \sqcup \neg \text{ChoiceC})$	(6)
---	-----

The subsumption relation can be reduced to concept satisfiability, as follows:

$(\text{ChoiceB} \sqcup \text{ChoiceC}) \sqcap \neg \text{ChoiceA} \sqcap (\text{ChoiceB} \sqcap \text{ChoiceC})$	(7)
---	-----

To fulfill this question, the answer must be unsatisfiable of formula (7). The reasoning process starts, as usual, with the constraint system (Formula 8):

$$L(x) = \{ (\text{ChoiceB} \sqcup \text{ChoiceC}) \sqcap \neg\text{ChoiceA} \sqcap (\text{ChoiceB} \sqcap \text{ChoiceC}) \} \quad (8)$$

Applying intersection and union reasoning rules on formula (8), we will have:

$$L(x) = \{ (\text{ChoiceB}, \neg\text{ChoiceA}, \text{ChoiceB}, \text{ChoiceC}) \} \quad (9)$$

It is easy to check that there is no clash. It implies that the initial OWL model (Fig. 5) is satisfiable, and it implies that the question does not subsume the answer. Individual “x” has the same meaning as in the previous example.

4 Conclusions and Future Work

The main idea proposed in this paper is how the Semantic Web based knowledge representation and Model Driven Architecture can be brought together in designing assessment systems. The paper also describes how to use DL reasoning techniques for intelligent analysis of students’ solutions. Analysis of the semantics of student’s answers is the key for providing the response processing of open-ended (simple choice) questions in the IMS QTI standard. Our future work will be focused on integration of the IMS QTI standard and other logic-based reasoning techniques.

References

1. Atlas Transformation Language - User Manual, Ver. 0.7, ATLAS group, Nant. [Online]. Available: [http://www.eclipse.org/gmt/atl/doc/ATL_User_Manual\[v0.7\].pdf](http://www.eclipse.org/gmt/atl/doc/ATL_User_Manual[v0.7].pdf).
2. F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P. Patel-Schneider: The Description Logic Handbook-Theory, Implementation and Application, Cambridge University Press (2003)
3. Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web, Scientific American, May. Vol. 284, No. 5, pp. 34.-43. (2001)
4. I. Horrocks: Optimising Tableau Decision Procedures for Description Logics, PhD Thesis, University of Manchester (1997)
5. IMS Global Learning Consortium, Inc. (IMS), Official Site, Online: <http://www.imsglobal.org>
6. Krdzavac, N., Devedzic, V.: A Tableau Metamodel for Description Logics, In Proceedings of 13th Workshop on Automated Reasoning (ARW 2006), pp. 7-9, Bristol, UK (2006)
7. IMS Question and Test Interoperability Implementation Guide, Version 2.1 Public draft, Online: http://www.imsglobal.org/question/qti_v2p1pd/imsqti_implv2p1pd.html
8. John D. Poole: Model-Driven Architecture: Vision, Standards and Emerging Technologies, Workshop on Metamodeling and Adaptive Object Models ECOOP (2001)
9. Radenković, S., Krdzavac, N., Devedžić V., A QTI Metamodel, In Proc. of Intl Multiconference on Computer Science and Information Technology, ISSN 1896-7094, 2007, pp. 1123 – 1132
10. UML 2.0 Infrastructure Overview, <http://www.omg.org/issues/UMLSpec.pdf>