An Approach for Ontology-based Elicitation of User Models to Enable Personalization on the Semantic Web

Ronald Denaux
Dept. of Computer Science
Eindhoven University of Technology, The Netherlands
r.o.denaux@student.tue.nl

Lora Aroyo
Dept. of Computer Science
Eindhoven University of Technology, The Netherlands
l.m.aroyo@tue.nl

Vania Dimitrova
School of Computing
University of Leeds
LS2 9NA, Leeds - UK
vania@comp.leeds.ac.uk

ABSTRACT

A novel framework for eliciting a user's conceptualization based on an ontology-driven dialog is presented here. It has been integrated in an RDF/OWL-based architecture of an adaptive learning content management system. The implemented framework is illustrated with an application scenario to deal with the cold start problem and to enable tailoring the system's behavior to the needs of each individual user.

Categories and Subject Descriptors

I.2 [**Artificial Intelligence**]: Knowledge Representation Formalisms and Methods; K.3.1 [**Computers And Education**]: Computer Uses In Education—*Computer-assisted instruction, Distance Learning*

General Terms: Human Factors, Design

Keywords: Personalization on the Semantic Web, User Modeling, Adaptive Content Management, Application of Semantic Web Technologies

1. INTERACTIVE ONTOLOGY-BASED USER MODELING

New approaches for open-world user modeling able to elicit extended models of users and to deal with the dynamics of a user's conceptualization are required to effectively personalize the Semantic Web. We present an ontology-based approach based on interaction between a software agent and a user, which results in the elicitation of a user's conceptual model aligned with a domain ontology. Our ultimate goal is to construct and maintain an **enhanced user model** that integrates different user perspectives, such as knowledge, personal preferences, interests, browsing patterns, cognitive and physical state. We illustrate this in an integrated environment for personalized learning content management, called OntoAIMS and the OWL-OLM [4] tool for open user modeling.

WWW 2005, May 10–14, 2005, Chiba, Japan.
ACM 1-59593-051-5/05/0005.

We use an example to illustrates the application of OWL-OLM to deal with the **cold start** problem concerning the lack of information about a user who logs on for a first time. OWL-OLM is employed to elicit a user model via a dialog (conducted in a graphical way, see Figure 1) guided by the Dialog Agent utilizing an existing domain ontology [2] and to exploring aspects of the User's conceptualization of the topic at hand. The agent opens the *dialog episode* and tells what the *goal* is. It then starts a *probing dialog game* to elicit the part of the user's conceptualization related to the current task topics, by using dialog *utterances* which consist of an *OWL statement*, rendered in a graphical form, and a *dialog intention*, rendered as a sentence opener. To state his intention, the user uses a sentence opener, see the buttons group to the right in Figure 1. The utterance is added to the the dialog history, shown at the top. When a probing is finished the agent closes the game and opens a dialog game that probes for another main concept. At any point a *mismatch* could be detected, which is represented as a discrepancy between the domain ontology and the user's conceptualization. In that case the agent can suspend the probing game and start a *clarification game* to check for an alignment between the user's conceptualization and the domain ontology and for inherited properties from various links. When the agent is done it sums up the result of the dialog episode based on the extracted user model. The dialog episode is closed by the agent because it has elicited sufficient aspects from the user's conceptualization to take the adaptation decision, which in this case is related to recommending appropriate resources. The dialog can be closed at any time by the user.

2. ARCHITECTURE OF OWL-OLM

OWL-OLM uses the STyLE-OLM framework for interactive open user modeling [4], which is amended to work with a domain ontology and user model built in OWL. It follows a general dialog framework that is domain independent. The only restriction imposed is that a URI of an OWL domain ontology has to be provided. For the current instantiation we use basic Linux ontology[1]. OWL-OLM uses Jena 2.1 [1] extensively for input and output of OWL ontologies and models, creating and changing OWL resources and resolving domain ontology queries (in the Dialog Agent). We also use the OWL generic reasoner from Jena to make inferences from the domain ontology. The architecture of OWL-OLM

[1]built in the SWALE project: `http://wwwis.win.tue.nl:8080/~swale/blo`

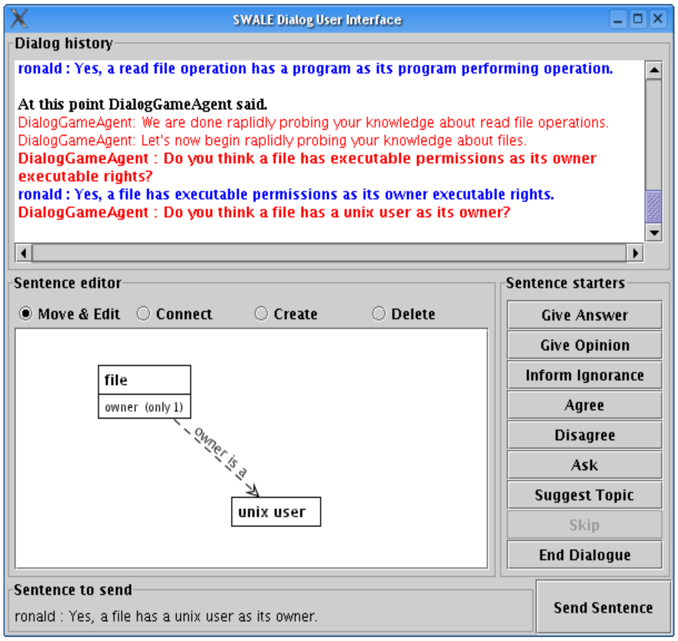

Figure 1: Example of OWL-OLM dialog interface.

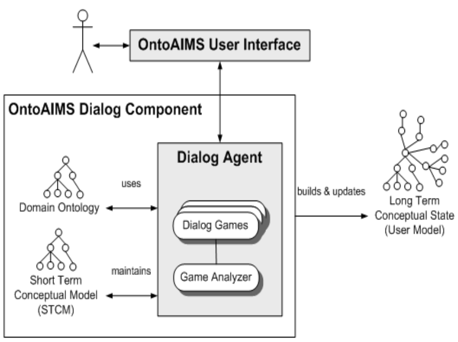

Figure 2: The architecture of OWL-OLM.

is presented in Figure 2.

The *Dialog Agent* is the main OWL-OLM component which maintains the user modeling dialog. A *Domain Ontology* built in OWL is used to maintain the dialog and to update the user's *Short-Term Conceptual State*. The latter is an OWL-based model of the user's conceptualization gathered throughout the dialog. The dialog agent maintains a *dialog episode goal*, which is divided into *sub-goals* that trigger *Dialog Games*- sequences of utterances to achieve a specific sub-goal. The agent has also a *Game Analyzer* that analyzes each user's utterance to decide the agent's response and to update the user's short term conceptual state. When a dialog episode finishes, the short-term conceptual state is used to update the user's *Long-Term Conceptual State*, referred to as *User Model*. For this, the simplified belief revision framework from is employed.

As shown in the OWL-OLM architecture, the dialog agent extracts a user's *Short Term Conceptual State* which is then used to tune the user's *Long Term Conceptual State*. The main idea of a conceptual state is that it gives a partial model of a user's conceptualization which is linked to one or more existing ontologies. A user's conceptual state is defined in OWL-OLM as a triple of URIs pointing to a *Conceptual model*, a *Domain ontology* and a *User*. The *conceptual model* is specified in OWL resembling an ontology specification, i.e. it defines classes, individuals, and properties, and uses OWL properties to define relationships. It makes links to resources defined in the domain ontology.

3. APPLICATION

The first stage of an empirical evaluation of OntoAIMS was conducted in a domain of Linux showing a strong potential of OWL-OLM to deal with the *cold start problem*. The users appreciated the help and guidance provided by OntoAIMS and felt that the OWL-OLM interaction was part of their learning process. Hence, OWL-OLM is regarded as a *key tool of the whole environment*, which is seen both as *complying with the overall goal* and *unobtrusive*. This is a great advantage of OWL-OLM over traditional, test-based methods for initializing user models in web-based learning systems. The dialog was seen as coherent by the users and they appreciated the option to ask questions. The evaluation revealed also pitfalls to be addressed in the next OWL-OLM versions. Result of the following larger user trial with the improved version of OntoAIMS are reported in [3].

This work contributes to research on personalization on the Semantic Web. It demonstrates an approach that can be used to improve the functionality of web ontology-based systems that aim at adapting to individual users conceptualization. Although this paper shows a specific application in a learning domain, the OWL-OLM user modeling approach can be applied in a variety of other domains where a system's behavior is driven by some encoded assumptions of a user's conceptualization. Possible applications include *digital libraries*, where effective help can only be provided if a user's view on the subject domain is considered, *online banking*, where a probing dialog can be used to quickly identify what conceptual models the users have of key terms, or *online catalogues*, where the search is driven by some taxonomy which may often differ from the user's perception of the domain. In all these cases, the cold start problem will be an issue, at least for some time until the system collects enough information about the user, and rapid diagnostic approaches, such as OWL-OLM, may be very useful.

Acknowledgements The work was supported by the UK-Netherlands Partnership in Science program and is part of WP1 of the PROLEARN network of excellence. Thanks to Michael Pye for implementing the GUI of OWL-OLM.

4. REFERENCES

[1] J. J. Carroll, I. Dickinson, C. Dollin, D. Reynolds, A. Seaborne, and K. Wilkinson. Jena: Implementing the semantic web recommendations. In *Proceedings of the Thirteenth International World Wide Web Conference, WWW 2004(Alternate track papers and posters)*, pages 74–83, 2004.

[2] R. Denaux, V. Dimitrova, and L. Aroyo. Interactive ontology-based user modeling for personalized learning content management. In *AH 2004: Workshop Proceedings Part II*, pages 338–347, 2004.

[3] R. Denaux, V. Dimitrova, and L. Aroyo. Integrating open user modeling and learning content management for the semantic web. In *10th International Conference on User Modeling*, 2005.

[4] V. Dimitrova. Style-olm: Interactive open learner modelling. *Int. Journal of Artificial Intelligence in Education*, 13(1):35–78, 2003.