A pragmatic approach to support concept-based educational information systems communication

A Pragmatic Approach to Support Concept-based Educational Information Systems Communication

Darina Dicheva
Winston-Salem State University
Department of Computer Science
601 Martin Luther King, Jr. Drive, Winston Salem, NC 27110
dichevad@wssu.edu

Lora Aroyo
Eindhoven University of Technology
Faculty of Mathematics and Computer Science
P.O.Box 315, 5600 MD Eindhoven, The Netherlands
l.m.aroyo@tue.nl

Abstract

Significant efforts are currently focused on defining powerful frameworks and architectures to support interoperability and integration of Web-based Educational Information Systems (EIS). We approach this integration problem from a rather practical perspective and propose a pragmatic framework for supporting communication between existing concept-based EIS aimed at utilizing systems' resources. The framework allows two independent systems to share and interchange information solely through ontology-based communication without sharing data stores. As a basis of the framework, we define a communication ontology and propose an interaction protocol, CB-EIS IP, built over a SAAJ-enabled SOAP transport layer.

1 Introduction

Web-based learning support systems, including Educational Information Systems (EIS) that are aimed at providing resources and services for various educational goals and tasks attract a growing interest. Representatives of such systems are adaptive textbooks constructed with AHA! [De Bra et al, 2003], InterBook [Brusilovsky et al, 1998] and NetCoach [Weber et al, 2001], or adaptive courses prepared within ELM-ART [Brusilovsky et al, 1996], PAT Online [Ritter, 1997], AIMS [Aroyo et al 2001], etc. Most of these specialized educational systems and content providers support only a single task/function within the educational process. In order to support a richer set of educational functions and increase their effectiveness, such systems need to interoperate, collaborate and exchange content or re-use functionality. Consequently, considerable efforts are currently focused on defining powerful frameworks and architectures to tackle issues of integration and interoperability of such systems. These frameworks prove useful for developing future effective large-scale web-based educational systems. In this paper we try to approach the integration problem of present systems from a rather practical perspective and propose a pragmatic framework for supporting communication between existing concept-based EIS aimed at utilizing systems' resources.

The main goal of web-based EIS is to provide the learners with immediate, on-line access to a broad range of structured information. They also support more efficient task performance by offering learners a domain-related help in the context of their work. There are a number of concept-based EIS already developed [Brusilovsky et al, 1998; Weber et al, 2001; Brusilovsky et al, 1996; Aroyo et al 2001; Dolog et al, 2004; Dicheva et al, 2004b] which typically include:

- concept-based (ontology-driven) subject domain,
- repository of learning resources,
- course (learning task) presentation,
- adaptation & personalization.

The fundamental feature of these systems is the *subject domain conceptualization*. It supports not only efficient implementation of their required functionality but also standardization: the concept structure can be built to represent a domain ontology that provides a broadly agreed vocabulary for domain knowledge representation. If the attached learning resources have also a standards-based representation as opposed to a system-specific internal representation, this will insure that the application's content is reusable, interchangeable, and interoperable. Good examples of such systems are AIMS (Adaptive Information Management System) [Aroyo et al, 2001] and TM4L (Topic Maps for Learning) [Dicheva et al 2004b], which we use as examples in our discussion. Though quite similar, these systems can be seen as complementary in the way they support learning tasks. While AIMS includes course representation and sequencing, TM4L is a digital library, which does not include direct course representation.

Integration and interoperability are very important for the EIS systems. If interoperable, two systems can benefit of additional functionality (supplied by the other system) and especially of sharing resources and common components, e.g. user models. In our example of AIMS and TM4L, TM4L can use AIMS course sequencing model, and resource metadata, while AIMS can use TM4L external and internal resources, domain and resource merging capability, text search, and external search. Our approach to the concept-based EIS integration problem is rather practical and based on sharing information between systems solely through communication without sharing data stores (e.g. providing data from one system on a request from another without allowing a general access to the private data store of the first system). The main questions related to the implementation of such communication concern the level of granularity of communicated information, the syntax and semantics of communication messages, and possible modes of use of user models (communicated or shared by systems). We have tried to answer these questions at two levels – a general one and a pragmatic one, which provided guidelines to the design of two corresponding frameworks for supporting communication between concept-based EIS. While the general framework fits well in the ambitious effort to define conceptually the shared and interoperable Educational Semantic Web by providing a powerful service-oriented architecture to support efficient communication between component-based EIS, the pragmatic one presents an efficient currently realistic solution by providing a *constrained* architecture for supporting shareability and exchangeability of existing systems' resources. Our implementation efforts as well as the focus of this paper are directed to the constrained architecture, since we believe that it will help to fill a gap between the current situation and the promises of the Educational Semantic Web of the future.

The paper is organized as follows. After a brief description of the general framework for supporting interoperability of various concept-based EIS in Section 2 (for details see [Dicheva et al, 2004a]), we propose a pragmatic approach for implementing the communication between two existing concept-based EIS (Section 3) by defining a communication ontology (Section 4) and proposing an interaction protocol, CB-EIS IP, built over a SAAJ-enabled SOAP transport layer (Section 5). We conclude with a short discussion.

2 General Architecture for Component-based EIS Interoperability

The proposed in [Dicheva et al, 2004a] general architecture for supporting component-based EIS include (see Figure 1):

- Stand-alone, component-based *independent EIS* using their private subject domain ontologies.
- *Information brokerage bureau* where all applications are registered.
- *Services* to support systems communication, e.g. for ontology mapping.
- *Communication bridges* between the systems supporting standardized transport mechanisms and a common interaction protocol.

The main purpose of the architecture is to support sharing and exchanging information between EIS initially designed to be standalone. This is achieved through communication between the systems (or their components) via services included in the framework to facilitate systems' communication. The services, including ontology-related services, are intended to support different specific aspects of the communication.

A communication is an interaction between two software systems (agents) guided by an interaction protocol. The communication between the systems requires not only standardized transport mechanisms and communication languages, but also *common content languages* and *semantics*. We have chosen XML as an 'information' content language to represent the content embedded in the messages in our architecture, as opposed to the commonly used 'logic' languages for representing the content, embedded in ACL (Agent Communication Languages) messages, such as KIF (Knowledge Interchange Format) [KIF, 1998], SL (Semantic Language proposed by FIPA) [FIPA-SL, 2004], and Prolog.

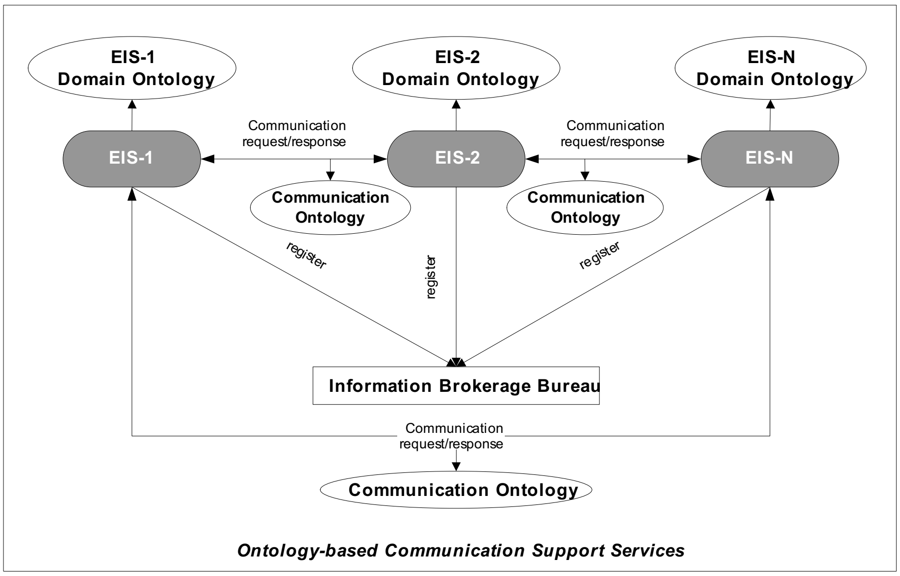

Figure 1. General architecture for component-based EIS interoperability.

As to the communication semantics, in order for the applications to understand each other we propose using a communication ontology that defines the vocabulary of terms used in the messages at both message and content layers (see Section 4). To interpret the requests and answers standardized domain ontologies, User Model ontologies, as well as upper-level ontologies such as, WordNet, etc. can be used.

Our next step is to *constrain* the proposed general architecture by considering *only two* communicating systems that "know" each other and "trust" each other. We consider that this is a common case and the goal is to find a configuration that will support such communication and allow sharing of systems' knowledge and resources.

3 Constrained Architecture for Concept-based EIS Communication

Since our present goal is to support communication between already developed concept-based EIS, each system is assumed to be a standalone application and is not required to have a particular architecture or to "adapt" to the other system in the framework. We make two important presumptions and use them as a basis for our design of a constrained architecture:

- The two systems know and are committed to communicating with each other. This implies that the systems will communicate directly and there is no need of Information brokerage bureau for registering the applications. Note that one system can communicate with more than one other system in such a direct mode.
- Concerning the services related to ontology mapping, an important presumption for our simplified framework is that the domain ontologies are created and used within *one community*, not different communities. This eases a lot the task since we may assume that in one community there exists an agreed upon understanding that favors the sharing of knowledge. Indeed, the goal of the EIS systems that we consider is to support learning in a specific course (discipline), for example a Database course. The community of users includes potential instructors (authors) and learners. Since the authors are knowledgeable in the specific subject domain, e.g. databases, it can be assumed that in defining the domain ontology they will use terms (concepts) that are accepted and agreed upon in that domain. This will remove the necessity of alignment and translation of domain ontologies. That is why ontology-related services are not included in the constrained architecture. Note that it will be useful though to include a service for merging ontologies. Currently, we delegate this task directly to the applications.

Table 1 compares the components in the general and constrained architectures.

Components	General Architecture	Constrained Architecture
Stand-alone EIS based on domain ontologies	Yes	Yes
Information Brokerage Bureau	Yes	No
Communication-supporting services	Yes	No
Communication Bridges	Yes	Yes

Table 1. Components required in both architectures

As shown in Table 1, the proposed constrained architecture includes only the two concrete communicating systems (e.g. AIMS and TM4L) and a communication bridge between them (see Fig. 2). Since in this architecture there are only two "committed" communicating systems, there is no real need of agent communication management, represented by Information brokerage Bureau and Communication-supporting Services.

We propose a common interaction protocol for the communication between Concept-Based EIS (CB-EIS IP) built over a SAAJ-enabled SOAP transport layer. As a content language we use XML, which is designed to support data exchange interoperability between applications. In the next sections we first propose a communication ontology for the pragmatic framework and then discuss implementation details of the proposed constrained architecture, including the proposed interaction protocol.

4 Communication Ontology

In order for the communicating applications to *understand* each other we need an ontology to provide a basis for sharing a precise meaning of symbols exchanged during communication. An ontology denotes a representation vocabulary of a specific domain, and more precisely the conceptualizations that the terms in the vocabulary are intended to capture [Chandrasekaran et al, 1999]. In our case we define a *communication ontology* which conceptualizes the domain of the communication between two concept-based EIS. We distinguish two parts, corresponding to both layers of an interaction between two communicating systems, the *message layer* and the *content layer*. Consequently, we propose the Communication Ontology (CO) to consist of communication content ontology and interaction protocol ontology (see Fig. 3):

- Communication content ontology (CCO) - describes the content (knowledge) that can be exchanged by the systems (corresponds to the *content layer*).
- Interaction protocol ontology (IPO) - specifies interaction communicative act types (corresponds to the *message layer*).

4.1 Content Ontology

The Communication Content Ontology defines the terms (concepts) needed to exchange messages, i.e. gives the meaning of the symbols included in the content expression.

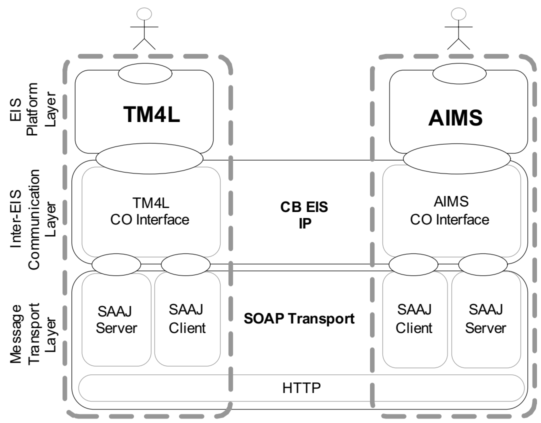

Figure 2. Pragmatic integration framework for AIMS and TM4L

When two concept-based EIS exchange data, the message content will typically include two types of terms (concepts): terms belonging to the domain ontology of the sender (the application sending the message) and terms categorizing domain term(s). The latter belongs to the general information model of concept-based EIS. For example, the sender can send a request for information of the kind "Send me all relationships in which you believe concept 'ER-model' is involved". In this message 'ER-model' is a term (concept) from the subject domain of the requesting application, while 'relationship' and 'concept' are terms belonging to the information model of concept-based EIS.

Thus in our framework, the content ontology consists of two parts: the application domain ontologies (DO) of the involved EIS and an application domain-independent ontology defining the concept-based information model of EIS (EISO). The latter includes basic terms describing the information model of concept-based EIS, such as concept, concept name, relationship type, relationship role, etc. Figure 3 presents an excerpt from this ontology. In the proposed framework, each application uses the common EISO ontology and its own domain ontology. For this reason we have depicted application domain ontologies separately from the EIS ontology in Fig. 1.

4.2 Interaction Protocol Ontology

The Interaction Protocol Ontology (IPO) defines terms related to message types, reasons, and preconditions. While the communication content ontology is generally independent of the framework's functionality, the IP ontology has to reflect its functionality (e.g. whether it supports agent communication).

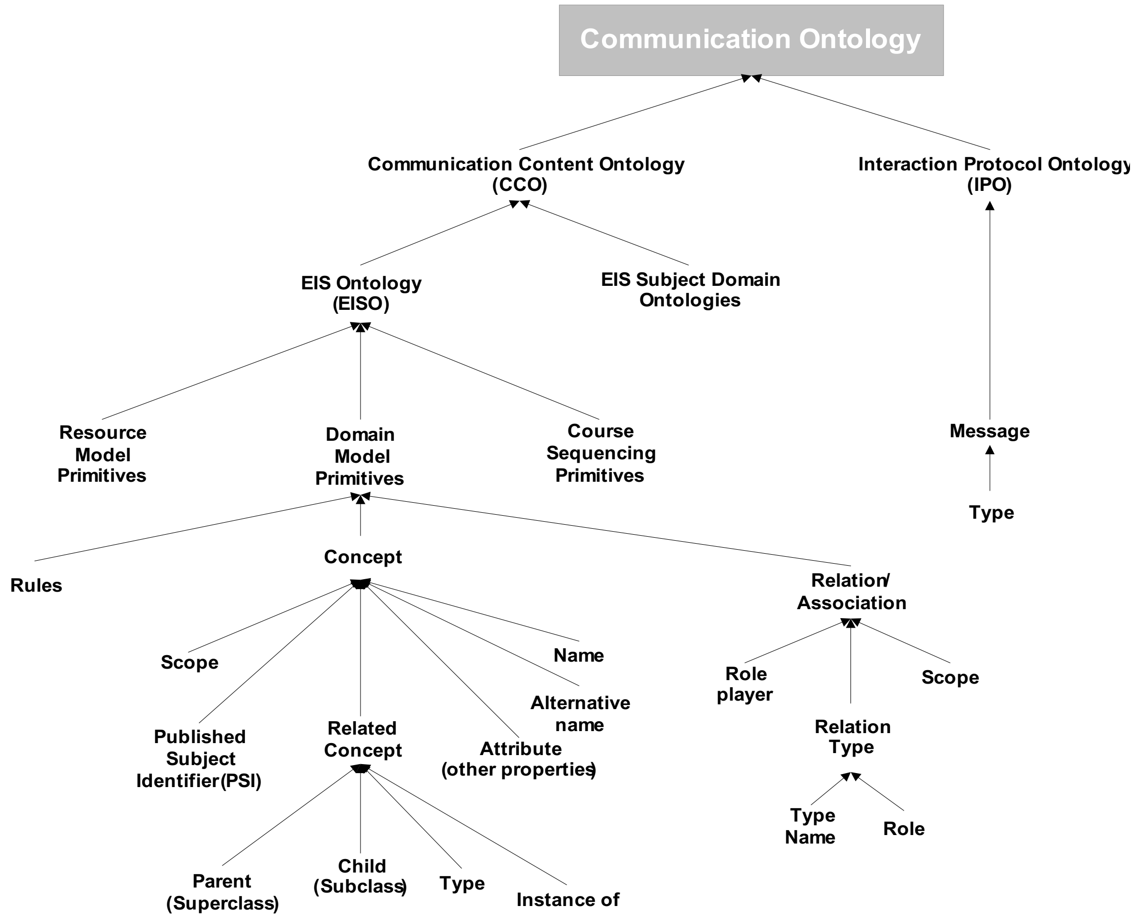

Figure 3. An excerpt from the communication ontology

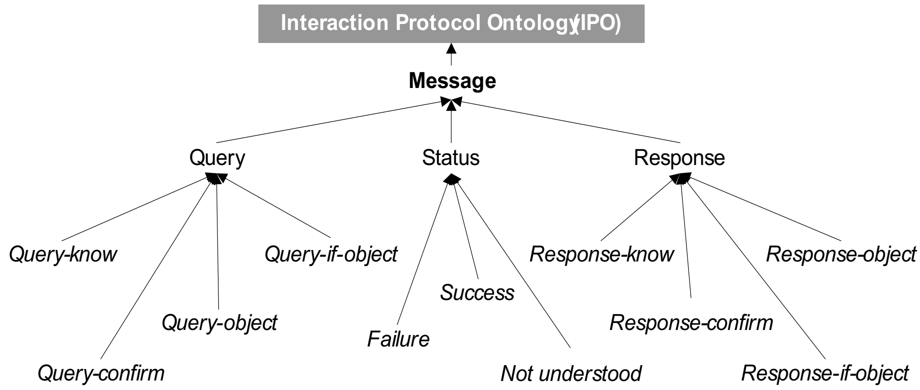

Figure 4. An excerpt from the IPO ontology

Messages represent communicative acts denoting the actions related to communication. In general, communicative acts (performatives) include (1) queries, (2) responses, (3) informational, (4) capability definition, (5) generative, and (6) networking (see KQML [Finin et al, 1994]). Since the two applications in our constrained architecture are "committed" to collaborate, the communication between them is very simple and does not require the typical variety of message types, for example, types such as agree, accept, cancel, propagate, and refuse, as well as defining message preconditions and reasons. Thus, in our case we choose the IPO ontology (Fig. 4) to include the following message types:

Status

- **Failure:** Informing that an action was attempted but the attempt failed.
- **Not understood:** message or Domain Ontology term.
- **Success:** Informing that an action was attempted and the attempt succeeded.

Query

- **Query-know:** Asking whether the receiver knows about an object corresponding to an EIS Ontology term/category (e.g. specific concept, relationship, etc).
- **Query-confirm:** Asking whether a proposition is true.
- **Query-object:** Asking for an object or all objects of specific category in the EIS Ontology (e.g. concept, relation, etc).
- **Query-if-object:** Asking for objects as in 'query-object' but in case a specified proposition is true.

Response

- **Response-know:** Informing the receiver whether or not the sender knows about the specified object.
- **Response-confirm:** Confirming to the receiver that the specified in the query proposition is true or not.
- **Response-object:** Sending to the receiver the objects specified in the request.
- **Response-if-object:** Sending to the receiver the objects specified in the request only if the specified proposition is true.

5 Communication Bridge

As a basis of the transportation mechanism in our framework we have chosen SOAP (Simple Object Access Protocol) [SOAP], which is a standard lightweight protocol for exchanging information in a decentralized, distributed environment. It complies with the WS-I Basic Profile 1.0 specifications and therefore supports interoperability across platforms, operating systems, and programming languages. It actually permits an exchange of messages in XML format between physically distributed machines. More specifically, the communication bridge is based on using the SOAP with Attachments API for Java (SAAJ). The SAAJ API, allows creating XML messages that conform to the SOAP 1.1 and WS-I Basic Profile 1.0 specifications. A SAAJ client is a standalone client. It sends point-to-point messages, i.e. a message goes from the sender directly to its destination. Messages sent using the SAAJ API are request-response messages. They are sent over a SOAP connection, which sends a message (request) and then blocks until it receives the reply (response). A SOAP message is an XML document. It always has a required SOAP part, and it may also have one or more attachment parts (that can contain any kind of content). The SOAP part must always have an envelop, which contains a SOAP body.

To realize the communication between two concept-based EIS, we propose an interaction protocol, CB-EIS IP (Concept-Based EIS Interaction Protocol), which provides the real semantics of the communication between them. Since the message content language in the framework is XML, we have defined a DTD for XML files representing the content of interaction messages that conform to this protocol.

```
<!ELEMENT message (queryMessage | responseMessage) | statusMessage>

<!ELEMENT queryMessage (query-know | query-confirm | query-object | query-if-object)>

<!ELEMENT query-know (commOntoTerm, dmOntoTerm)>
<!ELEMENT query-confirm (proposition)>
<!ELEMENT query-if-object (proposition, query-object)>
<!ELEMENT query-object (objectSpec, categorySpec)>

<!ELEMENT proposition (relation, dmOntoTerm, dmOntoTerm)>

<!ELEMENT categorySpec (commOntoTerm)>
<!ATTLIST categorySpec type (category | ALL )>

<!ELEMENT objectSpec (relOperator, commOntoTerm, dmOntoTerm)>
<!ATTLIST objectSpec type (object | ALL )>

<!ELEMENT relation (#PCDATA)>    <!-- term from DO -->
<!ELEMENT dmOntoTerm (#PCDATA)>   <!-- term from DO -->
<!ELEMENT commOntoTerm (#PCDATA)> <!-- term from EISO -->
```

Figure 5. An excerpt of the DTD definition of the CB-EIS IP

The DTD definition is based on the developed Communication Ontology (CO). An excerpt of the DTD document is given in Figure 5. This DTD allows sending messages like the following:

- A request asking whether the recipient "knows" the concept 'relational model':

```
<message>
   <queryMessage>
      <query-know>
         <commOntoTerm> concept </commOntoTerm>
         <dmOntoTerm> relational model </dmOntoTerm>
      </query-know>
   </queryMessage>
</message>
```

- A message, containing a "yes" response to the previous request:

```
<message>
   <response-know type = known/>
</message>
```

- A message, requesting the relationships in which concept 'ER-model' is involved:

```
<message>
   <query-object>
      <objectSpecification>
         <relationalOperator type = equal/>
         <commOntoTerm> concept </commOntoTerm>
         <dmOntoTerm> ER-model </dmOntoTerm>
      </objectSpecification>
      <categorySpecification type = category>
         <commOntoTerm> relationship </commOntoTerm>
      </categorySpecification>
   </query-object>
</message>
```

The CO interface modules in our architecture (see Fig. 2) are responsible for translating the messages (requests and responses) from the native language of EIS (e.g. TM4L or AIMS) into the language of the universal CB-EIS IP and vice versa. We plan to develop an API for Java (EISIPAJ), to be used by the CO interface for creating and interpreting XML files (representing the content of interaction messages) that conform to the CB-EIS IP. The CO interface is built on top of a SAAJ module and uses it to realize the CB-EIS IP with SOAP messages (the CB-EIS commands are embedded within the SOAP body).

Thus, in the proposed pragmatic framework, two independent systems can share and interchange information solely through ontology-based communication without sharing data stores. This removes any constraints on the systems architecture as well as the necessity of developing a 'wrapper' system, i.e. an environment that host the communicating systems. The only requirement for the systems is to be furnished with a plug-in realizing a CO interface that enables sending and receiving messages conforming to the proposed CB-EIS IP (through a SAAJ client and a SAAJ servlet).

6 Conclusion

We believe that the time for implementing large-scale educational web-service frameworks hasn't come yet. Thus our efforts are focused on increasing the use and efficiency of present, i.e., already developed or currently being developed systems, more specifically concept-based educational information systems. We propose to complement their functionality by supporting them to ask external 'known' peersystems for information, possibly involving information-providing processing.

We approach the problems related to systems integration and communication at two levels: a general level, proposing a powerful service-oriented framework to support efficient communication between component-based EIS, and a pragmatic one, illustrating an efficient proof of concept for supporting shareability and exchangeability of system resources, applicable in the context of the current educational computing advancement. We believe that the proposed constrained architecture will contribute to filling the gap between the current realistic situation and the desired future educational semantic web. As part of the framework we have defined a communication ontology consisting of communication content ontology and interaction protocol ontology and have embedded the latter within the CB-EIS IP. We have illustrated the concrete realization of the interaction protocol ontology within the constrained architecture. This way, we show how two independent systems can share and interchange information solely through ontology-based communication without sharing data stores.

The proposed framework for supporting communication between applications will eliminate in many cases the need for exporting the entire application domain model or other application model to another application. Thus, this will be an alternative to interchanging and merging domain models. The advantage is in eliminating duplication of stored information, which is unlikely to be often used. In addition, if an application has a specific concept-based application model with no corresponding model in the other system, import will not work and the proposed communication is the only way for the second system to use information from the first one. This will also solve problems related to shareability and reusability for already developed applications that don't use standards-based information but rather their own internal representations.

Acknowledgements

We would like to thank Christian Tsolov at the Univerity of Twente, The Netherlands and the paper reviewers for many valuable comments and suggestions on how to improve the paper.

References

[Aroyo et al, 2001] L. Aroyo and D. Dicheva. AIMS: Learning and Teaching Support for WWW-based Education. *International Journal for Continuing Engineering Education and Life-long Learning*, 11(1/2): 152-164.

[Brusilovsky et al., 1998] P. Brusilovsky, J. Eklund and E. Schwarz. Web-based education for all: A tool for developing adaptive courseware. Computer Networks and ISDN Systems. In *Proceedings of the 7th International WWW Conference*, pages 291-300. Springer.

[Brusilovsky et al., 1996] P. Brusilovsky, E. Schwartz, G. Weber. ELM-ART: An Intelligent Tutoring System on the World Wide Web. In *Proceedings of the 3rd International Conference on Intelligent Tutoring Systems*, pages 261-269, Springer.

[Chandrasekaran et al., 1999] B. Chandrasekaran, J. Josephson and R. Benjamins. What are Ontologies and Why Do We Need Them? J. *IEEE Intelligent Systems*, January-February, pages 20-26, 1999.

[De Bra et al., 2003] P. De Bra, A. Aerts, B. Berden, B. de Lange, B. Rousseau, T. Santic, D. Smits, and N. Stash. AHA! The Adaptive Hypermedia Architecture. In *Proceedings of the 14th ACM Conference on Hypertext and Hypermedia*, pages 81-85, 2003. ACM Press.

[Dicheva et al., 2004a] D. Dicheva and L. Aroyo. General Architecture Supporting Component-based EIS Interoperability. In *Proceedings of the Workshop on Semantic Web Technologies for E-Learning*, ITS'04, September 2004, Maceió, Brazil, 21-28.

[Dicheva et al., 2004b] D. Dicheva, C. Dichev, Y. Sun and S. Nao. Authoring Topic Maps-based Digital Course Libraries. In *Proceedings of the Workshop on Applications of Semantic Web Technologies for Adaptive Educational Hypermedia,* AH 2004, August 2004, Eindhoven, The Netherlands, pages 331-337.

[Dolog et al., 2004] P. Dolog, N. Henze, W. Nejdl and M. Sintek. Personalization in Distributed eLearning Environments. In *Proceedings of the 13th International WWW Conference.*

[Finin et al., 1994] T. Finin, R. Fritzson, D. McKay and R. McEntire. KQML as an agent communication language. *In Proceedings of the 3rd International Conference on Information and Knowledge Management*, pages 456-463.

[FIPA-SL, 2004]. FIPA SL Content Language Specification. Available at:

http://www.fipa.org/specs/fipa00008/SC00008I.html

[KIF, 1998]. Knowledge Interchange Format. Available at:

http://logic.stanford.edu/kif/dpans.html

[Ritter, 1997] S. Ritter. PAT Online: A Model-Tracing Tutor on the World-Wide Web. In *Proceedings of the Workshop on Intelligent Educational Systems on the 6th WWW conference*, pages 11-17.

[SOAP, 2004]. Available at: http://www.w3.org/TR/soap/

[Weber et al., 2001] G. Weber, H. C. Kuhl and S. Weibelzahl. Developing adaptive Internet-based courses with the authoring system NetCoach. In *Proceedings of the 3rd International Workshop on Adaptive Hypermedia*.