Modeling the Content of Adaptive Web-Based System Using an Ontology

Mária Bieliková and Michal Moravčík
Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies, Slovak University of Technology
Ilkovičova 3, 842 16 Bratislava, Slovakia
bielik@fiit.stuba.sk, michal.moravcik@mail.t-com.sk

Abstract

Adaptive web-based applications deal with an adaptation of information presentation (the content or navigation) employing goals and other characteristics of a user or context of the presentation. By increasing the number of existing adaptive applications the need for more effective creating and sharing the content among adaptive systems rises. In this paper we present a method aimed at creating the content represented by ontology and exporting it into existing adaptive application. The ontology includes a domain model and domain dependent part of a user model. We evaluated the method using the domain ontology of teaching programming by examples. We claim that the content and structure of the adaptive application represented by the ontology is suitable for sharing and reusing when defining new applications.

1. Introduction

Assisting a user in finding relevant information by navigating in large information space through a web-based application is a key requirement today. One approach to achieve this is adapting information presentation or navigation to the user or context of the presentation. Nowadays, new adaptive web-based applications are being still developed. With respect to the increasing usage of adaptive web-based applications the need for effective authoring and content delivering becomes increasingly important. However, the content of current adaptive applications is generally not designed for cross-system usage. We can improve situation by providing means for developing reusable models of adaptive applications together with tools for creating instances of the models.

The base idea of modeling is to deliver general, simple, consistent, portable, reusable representation of the modeled subsystem. These characteristics are in the goals of several existing reference models of the adaptive hypermedia. By enabling a transformation of the content among adaptive applications existing authoring systems can be reused as authoring tools not only for adaptive applications they were developed for. This principle is known as a metaphor "*authoring once, delivering many*" [16].

Our aim is to support a reuse the content (domain) model of adaptive web-based applications employing an ontology as the knowledge structuring approach used extensively for the Semantic Web applications. The goals and possibilities of the Semantic Web designate the advantage of using the ontology as a domain model of adaptive applications. Several approaches expressing particular models of adaptive applications by means of the Semantic Web technologies exist (e.g., [1], [12], [15]). Nevertheless, the lack of effective reuse of adaptive applications models is still in course.

We present a method for modeling content of adaptive applications using ontology, exporting the content into intermediate format and importing it into the adaptive application. For evaluation of the method we selected the CAF format (Common Adaptation Format, [8]) and adaptive web-based application AHA! [9]. We present core ontology of the content used throughout the method. We also evaluate the method using the domain of teaching programming.

2. Adaptive application content modeling

Content-oriented adaptive applications modeling can be seen from two points of view: modeling the architecture of an application (models of generic web-based applications can be also considered) or modeling the content presented by the application (represented using a domain model). Architecture of adaptive web-based systems can be expressed using methods for modeling generic web-based systems (e.g., OOHDM [17], WebML [5], Hera [12]) even though the specification of adaptive behavior is not accordingly handled by all of these methods. Common architecture of adap-

tive hypermedia systems reflects reference models as AHAM [19], Munich reference model [13] or LAOS [6], which define layers of the adaptive application by separating data models (e.g., domain, user, context, goals) from an adaptation engine. Reference models are used as a base for new adaptive applications.

Not all layers of adaptive application model are adequate to be shared and/or transferred between systems. The common for the applications is the content with definitions of its usage in a user model (expressed in domain-dependent part of the user model).

2.1. Content representation

Modeling the content of a content-oriented web-based system and its efficient representation is as well important as modeling the system itself. For content modeling it is important to analyze to what extent is a particular representation flexible for different domains together with the possibility of reasoning directed to decisions on properties of the information space (e.g., consistency). We do not consider proprietary formats as they almost totally prevent the sharing and reuse of the domain model.

Existing approaches to representing the content of a web-based system include mainly approaches using a relational database or an XML based language. XML offers powerful enough expressiveness. The performance of this solution is limited by the performance of the used file system (it is effective for domain models with few instances and rich structure of concept's characteristics). Reusability and sharing is better than with the database approach, thanks to the platform independence of XML. Using XML has also the advantage that it can be used directly in the Web environment. However, XML as a meta-language defines only general syntax without formally defined semantics, which leads to difficulties when reasoning is required. Moreover, everyone can invent his own names for tags; somebody stores attributes as tags; somebody uses the attributes of tags defined by XML syntax.

Both above mentioned approaches offer only a way of describing characteristics of domain model concepts and do not offer any added value from the content modeling perspective. Ontology-based approach offers a way of moving content modeling from the low-level describing of domain concept characteristics to a higher-level with additional possibilities (reasoning).

According to the most cited definition of ontology in the Semantic Web community, ontology is an explicit specification of the conceptualization of a domain [11]. The term ontology includes a whole range of models that show varied semantic richness. In this paper we consider representing the ontology by RDF/OWL formalisms (Resource Description Framework; Web Ontology Language). An approach based on RDF and its extension OWL takes the previously mentioned XML representation (syntax) and eliminates its disadvantage by defining a vocabulary for describing properties and classes. OWL serves as a common language for automated reasoning about the content for the vision of the Semantic Web.

The advantages leading to using ontologies for content modeling come from the fundamentals of this formalism. Ontologies provide a common understanding of the domain to facilitate reuse and harmonization of different terminologies. They support reasoning, which is considered as an important contribution of the ontology-based models. Although several systems use ontology as a base for models representation [15], usually specialized ontology for particular case is developed only. We use the ontology as a universal format for representation of models and define a method for their reusability.

2.2. Content transformation

Creating a schema of the domain and instances of it is serious bottleneck of content-oriented web-based systems. Using ontology for representation of the domain increases the probability that domain concepts together with their characteristics will be shared among a range of applications of the same domain (especially on the Web, where most ontologies are currently represented using OWL).

Sharing the content can be realized by using commonly accepted model for the content of adaptive applications in particular domain, or converting content among adaptive applications. A commonly accepted domain model is ideal solution. Since we agree that building common vocabularies is important and useful (we remark the role of standards), considering a large distributed information space (e.g., the Web) we need to make a compromise between enabling diversity and looking for mappings between various models. The idea of commonly accepted domain ontology is simply impossible to reach in such diverse and distributed environment as the Web.

On the other hand, designing converters for each pair of applications is rather ineffective and demanding approach. A compromise lays in defining an intermediate format of the information content for adaptive applications. In this case it is sufficient to convert the content from the intermediate format to the adaptive application format and vice versa. Standards can also help in this process [4].

One of the first attempts to use information content in several applications was a conversion from the Interbook to AHA! [10]. In this case the Interbook system serves as authoring tool and the AHA! system for

adaptive presentation. Another approach described in [8] defines a conversion from the MOT system (My Online Teacher, [6]) to adaptive applications of AHA! and WHRULE systems. MOT is used as an authoring tool, where it is possible to define the content of adaptive application and adaptation strategy that specifies personalization of educational content according changing user characteristics.

The conversion from MOT to AHA! uses intermediate format CAF (Common Adaptation Format) that defines a hierarchy of concepts and corresponding attributes of concepts using an XML.

3. Development of sharable content of adaptive application

Ontology is suitable mean for representing and maintaining data related to an application domain. Figure 1 depicts our proposal of domain model schema as it is modeled using the ontology. Considering adaptive web-based applications we define also domain dependent part of the user model that is automatically generated from the domain model.

The domain model defines concepts and their content (for the content of concepts we use term *concept attributes*), domain attributes of concepts and typed *relations* to other concepts. Domain attributes represent information important for adapting but they are not necessarily viewable in the presentation.

The user model defines for each concept a *User View* and a set of *User View Attributes*. User view makes a connection among specific user, concept and attributes, which frame together the overlay user model defined in both AHAM and LAOS reference models.

Boxes in Figure 1 represent classes in the ontology and every connection (except those deriving the attributes) stands for a relation in the ontology. Structure specification of the adaptive application content is done by defining classes and their relations in the ontology. The content of adaptive application itself is represented by instances of classes defined in the ontology with connections specified by relations.

3.1. The method

We present a method for modeling the content of adaptive application that uses *core ontology* designed for adaptive application content modeling. It defines steps essential for creating a new domain ontology, which is compliant with the core ontology, describes an export of the ontology into an intermediate format, which can be used for importing into existing adaptive systems. We assume also a situation where domain model already exists and existing ontology is used as pure domain model of the adaptive application.

The method defines following steps for authoring adaptive application domain model represented by ontology:

1. Specify the concept classes and the concept class hierarchy.
2. Define attributes for the concept classes
   - data attributes,
   - domain attributes,
   - user attributes.
3. Specify and type relations between concepts of specified classes.
4. Create concept instances as the class instances of the ontology.
5. Deliver the ontology content into existing adaptive system.

Performing steps 1-3 leads to a structure of adaptive application domain ontology. Step 4 is devoted to fill-

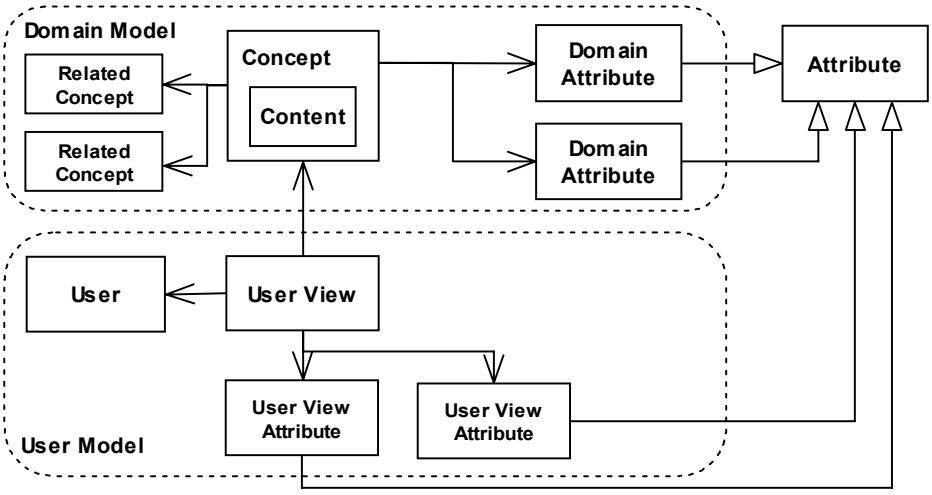

Figure 1. Schema of adaptive application content model.

ing the information content into the ontology. Finally, step 5 results in creating the intermediate format for adaptive application content and importing the content into particular adaptive application. The method is described in detail in [14]. It does not aims at creation of domain-independent pare of the user model as it is not related to the information content.

3.2. Core ontology of content model

The core ontology is depicted in Figure 2. It defines basic terms of adaptive application content and relations between them. Relations between concepts are typed as *linked*, *parent*, and *child*. We assume bidirectional relations between concepts which are supported and simple to maintain in OWL.

We specify three types of concepts on the highest level (classes derived directly from the *Concept* class):

- *DefinedConcept:* this class is the base class for all user defined concept classes. By deriving classes from the *DefinedConcept* class we differentiate user defined classes from standard classes from the core ontology;
- *RelationalConcept:* it is the base class for all relational classes. It indicates that the concept is in a relation with other concept. Type of the relation is defined by source and target relational concept classes, which are derived from the *RelationalConcept* class;
- *ApplicationConcept:* it stands for the root concept class, which is an entry point to the adaptive application. The *ApplicationConcept* class is also derived from the *ParentConcept* class, which defines its relation of type parent-child with some other concepts (application content).

The *View* class represents the view in terms of the *User View* example. It is defined as generic to enable adding new types of views into the core ontology.

4. Evaluation of proposed method

In our experiments we focused on adaptive web-based system AHA! [9] and authoring tool for adaptive applications MOT [8]. The AHA! system is partially compliant with the AHAM model and the MOT system is based on the LAOS model. Both provide authoring tools for creating adaptive applications. Level of authoring is superior in MOT, which provides simple and powerful definition of domain concept maps, lesson model (defined in the LAOS model) and adaptation strategies (LAG programs, [7]). On the other side, AHA! provides superior adaptive techniques for defining adaptive presentation. This resulted to a proposal to use MOT as an authoring system and AHA! as a delivery system supported by transforming the content between these two systems (MOT2AHA, [8]). Transformation is either straightforward, or consists of two steps where the intermediate format (CAF, Common Adaptation Format) is used.

We use the core ontology for authoring and the CAF format together with existing converting tools for delivering the content into the AHA! application.

Evaluation was realized in three stages:

1. Definition of prototype ontology in domain of teaching programming by examples.
2. Export prototype ontology into the adaptive application.
3. Developing software support for authoring the adaptive application content ontology.

In the first stage we developed the ontology describ-

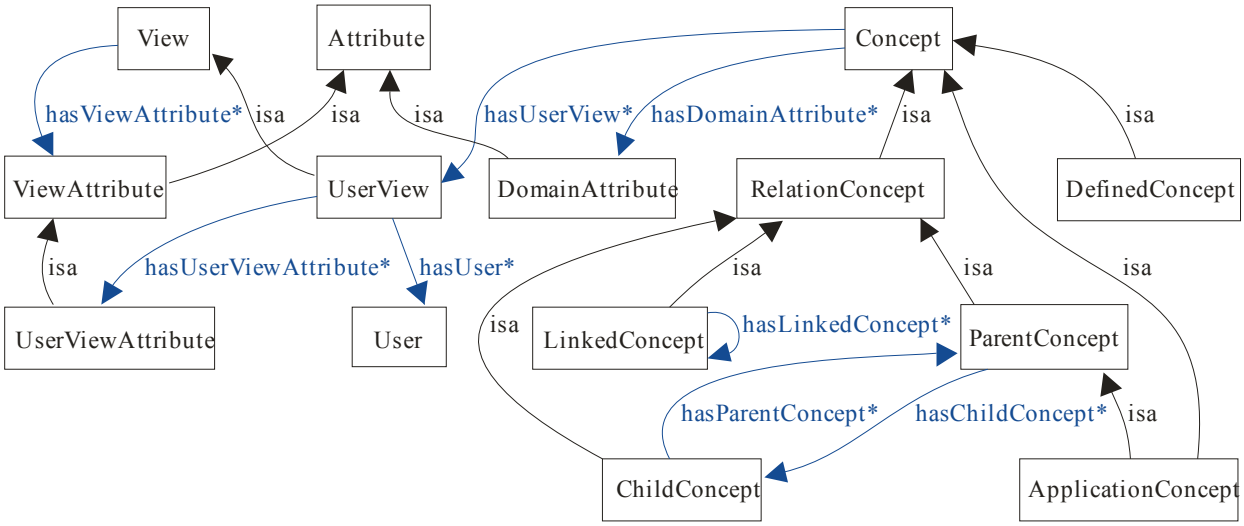

Figure 2. Core ontology of adaptive application content model.

ing a programming course using program exercises for languages Lisp and Prolog (based on the adaptive web-based system ALEA used for teaching programming in Functional and Logic programming course at the Slovak University of Technology in Bratislava [2] and the manual transformation of a part of the ALEA content into the AHA! application [2]).

Structure of the domain model is shown in Figure 3. For simplicity we omit here inverse relations between the concepts. *ProgrammingExercises* represents the root concept of the application, which can include a set of program schemata (*TemplateConcept*) and a ser of exercises (*ExerciseConcept*). Program schemata can include concepts describing the program schema usage (*TemplateUsage*) and concepts containing exercises (*ExerciseConcept*). Each exercise contains one or more concepts defining the exercise (*ExerciseDefinition*) and its solution (*ExerciseSolution*). The *hasSubTemplate* relation enables to build a hierarchy of the template concept instances.

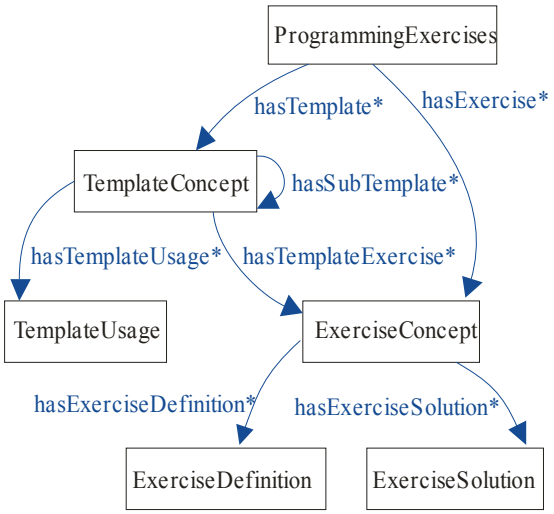

Figure 3. Programming course domain model.

After defining a structure of programming course, we filled the ontology up with instances of programming templates and exercises to enable further evaluation.

During the second phase of evaluating we found a problem of unwanted misplacing concept attributes in generated presentation. To solve this issue we designed a metamodel of ontology, which defines concept classes and their attributes and relations. By editing the metamodel and applying it during exporting to CAF format concept attributes and relations are resorted as specified in metamodel.

Due to limitations of CAF format we proposed an extended format called CAFE (CAF Extended), which simplifies CAF format and introduces possibility of complete and consistent definition of concept attributes and relations. The lesson model from the CAF format is no longer needed, while the concept hierarchy can be derived from concept relations.

Second and third stages of the evaluation are supported by software tools developed in order to make proposed sequence of steps defined by the method for modeling the content of adaptive application practicable. We developed tools for editing, importing and exporting the ontology defining the content of adaptive application. Figure 4 visualizes original process of transforming content between MOT and AHA! (top part of the figure). Grayed elements represent proposed process of defining and exporting adaptive application content using ontology.

Currently the whole process of developing the content represented by ontology and transforming it for specific adaptive application is supported by using the CAF format. The import of the content represented by CAF format to AHA! is realized using CAF2AHA! tool developed at Eindhoven University.

5. Conclusion and future work

In this paper we described a method for adaptive application content modeling using ontology that allows the content reuse between applications. It is based on designed core ontology that is open and can be used for integrating other aspects of adaptive behavior and other layers of reference models. To provide generating fully editable AHA! applications implementing conversion from CAFE format to the AHA! models is necessary. Adaptive applications benefit also from

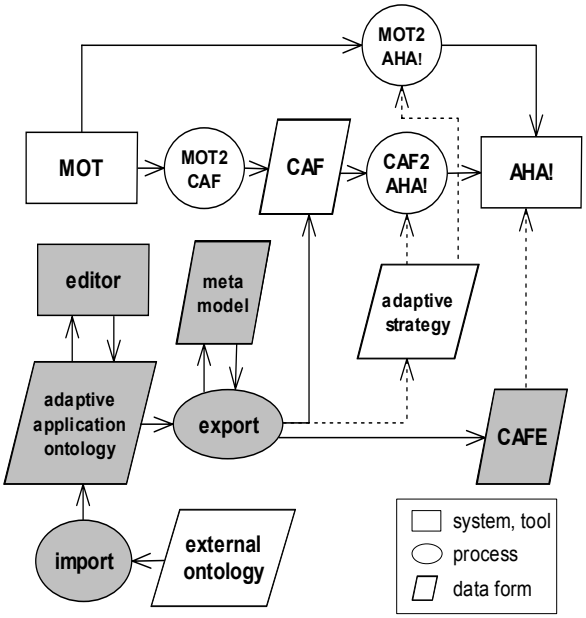

Figure 4. Content transformation possibilities.

generated domain-dependent part of user model.

Our long term goal is a use of ontology as knowledge representation in adaptive web-based applications. It allows building both closed and open corpus systems using the same processes. Knowledge representation using ontology reveals possibilities of reasoning and thus supports defining adaptive behavior. Moreover, automatic means for sequences of content generation (e.g., using knowledge on user behavior [18]) or checking its consistency could be employed.

Our work was aimed at delivering adaptive application content into existing adaptive systems. We have demonstrated authoring of adaptive application content using the ontology and its delivering into existing system in terms of "authoring once, delivering many", which is promising real utilization of value-added possibilities for modeling adaptive applications with means of the Semantic Web. Described approach of modeling adaptive applications content employing the ontology is a step to support the Semantic Web technologies in adaptive web-based systems.

Acknowledgements

This work was partially supported by the Scientific Grant Agency of Slovak Republic, grant VG1/3102/ 06 and by the Cultural and Educational Grant Agency of the Slovak Republic, grant KEGA 3/2069/04.

The authors would like to thank members of PeWe group (`www.fiit.stuba.sk/research/pewe`) for fruitful discussions and feedback on work presented in this paper.

References

[1] Andrejko, A., Barla, M., Bieliková, M., Tvarožek, M.: Ontology-based user modeling for web-based information systems. In *Proc. of Int. Conf. on Information Systems Development, ISD'06*, Budapest, Hungaria, 2006.

[2] Bieliková, M., Kuruc, J., Andrejko, A.: Learning programming with adaptive web-based hypermedia system AHA! In *Proc. of Int. Conf. on Emerging e-Learning Technologies and Applications, ICETA'05*, F. Jakab et al., editors, Košice, Slovakia, 2005, pp. 251-256.

[3] Bieliková, M.: An adaptive web-based system for learning programming. *Int. J. Continuing Engineering Education and Life-Long Learning*, Inderscience, Vol. 16, Nos. 1/2, 2006, 122-136.

[4] Bureš M.,Jelínek, I.: Reusable Adaptive Hypermedia E-learning Content Using AICC. In *Proc. of the IADIS Int. Conf. WWW/Internet'05*, Lisboa, Spain, IADIS Press, 2005, vol. I., pp. 376-378.

[5] Ceri, S., Fraternali, P., Matera, M.: Conceptual modeling of data-intensive web applications. In *IEEE Internet Computing*. July/August 2002, pp. 20-30.

[6] Cristea, A.I., De Mooij, A.: LAOS: Layered WWW AHS authoring model and their corresponding algebraic operators. In *Proc. of 12th Int. World Wide Web Conference, WWW'03*, Budapest, Hungary, ACM, 2003.

[7] Cristea, A.I., Verschoor, M.: The LAG grammar for authoring adaptive web. In *Proc. of Int. Conf. on Information Technology: Coding and Computing, ITCC'04*, IEEE Computer Society Press, 2004, pp.382-386.

[8] Cristea, A.I., Smits, D., De Bra, P.: Writing MOT, reading AHA! – converting between an authoring and a delivery system for adaptive educational hypermedia. In *Proc. of 3rd Int. Workshop on Authoring of Adaptive and Adaptable Educational Hypermedia, A3EH'05 at AIED'05*, Amsterdam, The Netherlands, 2005.

[9] De Bra, P., et al.: AHA! – The adaptive hypermedia architecture. In *Proc. of the ACM Hypertext Conf.*, Nottingham, UK, 2003, pp. 81-84.

[10] De Bra, P., Santic, T., Brusilovsky P.: AHA! meets Interbook, and more. In *Proc. of the AACE ELearn'03 Conf.*, Phoenix, Arizona, November 2003, pp. 57-64.

[11] Gruber, T.R.: Towards principles for the design of ontologies used for knowledge sharing. In N. Guarino and R. Poli, editors, *Formal Ontology in Conceptual Analysis and Knowledge Representation*, Kluwer, 1993.

[12] Houben, G. J., Barna, P., Frasincar, F.: HERA: development of semantic web information systems. In *Proc. of Int. Conf. on Web Engineering, ICWE'03*, LNCS 2722, Springer, 2003, pp. 529-538.

[13] Koch, N., Wirsing, M.: The Munich reference model for adaptive hypermedia applications. In *Proc. of Int. Conf. on Adaptive Hypermedia and Adaptive Web-based Systems, AH'02*, P. De Bra, P. Brusilovsky, R. Conejo, editors, LNCS 2347, Springer, 2002, pp.213-222.

[14] Moravčík, M.: Modeling of adaptive web-based systems. Slovak University of Technology in Bratislava, Diploma thesis, M. Bieliková, supervisor, 2006.

[15] Seefelder, P.A., Schwabe, D.: A semantic meta-model for adaptive hypermedia systems. In *Proc. of 3rd Int. Conf. on Adaptive Hypermedia and Adaptive Web-Based Systems, AH'04*, Eindhoven, The Netherlands, LNCS 3137, Springer, 2004, pp. 360-365.

[16] Stewart, C., Cristea A.I., Brailsford, T.: Authoring once, delivering many: creating reusable adaptive courseware. In *Proc. of 4th IASTED Int. Conf. on Web-Based Education*, *WBE'05,* Grindelwald, Switzerland, 2005.

[17] Schwabe, D., Rossi, G.: An object-oriented approach to web-based application design. In *Theory and Practice of Object Systems*, Special issue on the Internet, Vol. 4, No. 4, 1998, pp. 207-225.

[18] Velart, Z., Šaloun, P.: User behavior patterns in the course of programming in C++. In *Proc. of the Int. Workshop on Adaptivity, personalization & the Semantic Web*, Odense, Denmark, 2006, pp. 41-44.

[19] Wu, H., Houben, G.J., De Bra, P.: AHAM: A reference model to support adaptive hypermedia authoring. In *Proc. of the Conf. on Information Science*, Antwerp, 1998, pp. 51-76.