Courseware authoring for adaptive e-learning

Boyan Bontchev  
Department of Software Engineering  
Faculty of Mathematics and Informatics at SU  
Sofia, Bulgaria  
bbontchev@fmi.uni-sofia.bg

Dessislava Vassileva  
Department of Software Engineering  
Faculty of Mathematics and Informatics at SU  
Sofia, Bulgaria  
ddessy@fmi.uni-sofia.bg

Abstract— In recent years, the number of adaptive applications for e-learning content delivery increased immensely. Usually, such applications have own authoring tool or use external one in order to create learning materials. However, very few authors of educational content use such systems. The reason is that the tools are rather complicated and follow modern e-learning standards, which leads to a need for content authors to fill multiple metadata for theirs learning materials, that requires much time. In this paper we present an authoring tool, which is a part of a platform for building edutainment (education plus entertainment) services – ADOPTA (ADaptive technOlogy-enhanced eduTainment platform). This authoring tool provides inheritance mechanisms for learning object metadata descriptions, metadata for semantic ontologies, and good integration with instructor tool for creation of adaptive courseware.

Keywords: authoring tools, learning style, AHS, LOM

I. INTRODUCTION

In last fifteen years, authoring and delivery of adaptable e-learning courseware appears to be very important for design of modern learning management platforms. During that period, there have been proposed a lot of works identifying the key challenges in adaptive Web based multimedia information delivery. The chief goal of personalized and adaptive e-learning was formulated in [1] as assuring of “e-learning content, activities and collaboration, adapted to the specific needs and influenced by specific preferences and context of the student”. In order to achieve that goal, Adaptive Hypermedia Systems (AHS) possess abilities for provisioning of various forms of adaptation, such as adaptive navigation, structural adaptation, adaptive presentation and historical adaptation. Some research groups focus on adaptability to learners’ current knowledge based on the theory of knowledge spaces [2]. The use of learning objects provides an excellent opportunity for learners to apply their own meanings in various information contexts. Dynamic adaptation is used in different instructional scenarios with content package adaptation facilitated by wide usage of Web services [3]. Other researchers introduce additional level of system self adaptability based on the idea that different forms of learner model can be used to adapt content and links of hypermedia pages to given user [4]. The adaptability supposes dynamic changes in adaptation process based on modification of the content parameters according input from learner passing hypermedia resources and assessment about their understanding [5].

The paper describes authoring of e-learning courseware adopted in the scope of ADOPTA (ADaptive technOlogy-enhanced eduTainment platform) for building edutainment (education plus entertainment) services. ADOPTA will provide authoring and instructor tools and adaptation engine executing rules controlling the adaptation process toward the learner model. The authoring process is strongly separated from the instructor’s learning design and uses semantic ontologies and inheritance mechanisms for metadata descriptions of both the learning objects and ontologies. The adaptive instructional design allows construction of different paths within the storyboard suitable for different learning styles (for adaptive navigation and link annotation) and, also, tagging of learning objects within a page depending on learner performance (for adaptive content selection).

II. RELATED WORKS

In the last decade numerous e-learning authoring tools and platforms for courseware have been developed. In this chapter we will make a brief overview of some of them covering most expected of authors of e-learning content characteristics. We choose to consider following authoring tools:

- InterBook [6] – a system dedicated to serving adaptive electronic textbooks (called interbook). The InterBook transforms plain text to specially annotated HTML. It supports adaptive guidance, adaptive help and navigation. The individual user model for each user is stored. Two major kinds of information items are supported by InterBook - a book page and a domain knowledge concept – assuring a clear separation between the contents and the concepts.
- AHA! [7] – a successor of 2L690, and it has inherited several of its features. AHA! is open source general-purpose Adaptive Hypermedia Architecture based on AHAM (Adaptive Hypermedia Application Model) [8]. User Model and Adaptation Engine in it are strictly separated. The adaptive content presentation is based on link annotation and link hiding.
- BlackBoard [9] – it provides course and content management systems, collaboration tools and a number of other services. It is one of the most
popular and successful commercial e-learning systems.

- NetCoach [10] – it contains an own authoring system that allows the development of adaptive courses. Generally all material belonging to a course is organized in a tree structure and can be freely browsed by the learner. Additionally, the system offers personalization of courses by adaptive curriculum sequencing and adaptive link annotation.

- Hypertext Composer (HyCo) [11] – this authoring tool creates semantic learning objects for web-based e-learning systems. It supports the management of bibliographical references and particularly the design of adaptive learning experiences.

Considered above authoring tools generalize contemporary characteristics, which it is expected to provide one, such as:

- adaptation support – through using adaptive techniques as adaptive guiding, link hiding and annotation, adaptive presentation.
- standards compatibility – this gives possibility to assuring interoperability between different learning management systems. The most widespread standards are SCORM, IEEE LOM, IMS CP and IMS QTI.
- re-usability – for this goal, the learning content is composed of learning objects (LOs), which can be used in different courses and in other LOs.
- flexibility and extensibility.

III. A CONCEPTUAL MODEL OF AHS

The AHS model described in details in [12] follows a metadata-driven approach, explicitly separating narrative storyboard from the content and adaptation engine (AE). Fig. 1 represents the triangular structure of our model which refines the AHAM reference model [8] by dividing in three each one of the learner’s, domain, and adaptation models. This is a new hierarchical organizational model for building adaptive hypermedia learning management system (LMS).

At first level, the model is based on a precise separation between learner, content and adaptation model, while at second level each of these sub-model is divided into three others sub-models. All the sub-models should be defined as XML schemas representing the characteristics of a learner that must be modeled and used for cross-session interoperability and consistency.

The main benefit of the proposed model is in assuring strong independence of any of the building models and, at the same time, in facilitating a flexible adaptation of content delivery. It can be supported by different system architectures not limiting application of various adaptation techniques, such as adaptive content presentation, navigation support and content selection. In order to be able to describe polymorphic learner profiles, we define conceptual characters of particular domain such as characteristics of the learning style, psychology characters, etc. Each of the conceptual characters describing the learner has a weight factor Wci (zero or any integer number, or percent between 0% or 100% incl.) specifying the importance or the level of presence of that concept (character) inside the learner model as shown in fig. 2. Thus, a conceptual character having no importance or not being present receives zero weight. Note, that the learner model is not fixed to one of the existing learning styles models (such as that of Honey and Mumford, Gregoric-Mind, Dunn and Dunn, etc. [13]) and can be used for any of these style models.

![Figure 1. Structure of the triangular model](image)

A. The learner model

Unlike other approaches, in the learner model we separate goals and preferences from shown knowledge and performance, as the first sub-model (goals and preferences) is static while the second one (knowledge and performance) is rather dynamic and takes a part in the event-driven storyboard monitoring. The third sub-model is that one of learning style and takes a central point within the learner model. Depending on the style model, learner characters could be activist, theorist, pragmatist or reflector, or such as visual, auditory, or kinesthetic, or others. The learning style is detached as separate sub-model and can be used for choosing best content for a learner possessing given mixture of learning styles as far as most of learners cannot be determined only by one style (fig. 2). While the learning style can be determined in the very beginning of the learning explicitly by the learner or by appropriate pre-tests, other tests should be exercised during the e-learning process in order to assess prior or gained knowledge and performance.

![Figure 2. A sample conceptual learner style model.](image)
results of each individual learner. Learner performance is used to control adaptive content selection.

B. The domain model

The domain model is composed of content itself (granulized in learning objects (LOs) according to the SCORM standard), LO’s metadata (LOM) and semantic ontologies organizing the content (LOs). There are supported various types of LOs – not only narrative content but also any learning activity such as task, topic for writing an essay, assessment question, game, etc. The semantic ontology should be specified by the course author at the beginning, in order to form a logical taxonomy for the knowledge domain (i.e., domain ontology) during the authoring process. Thus, the content LOs are developed by the author and next are placed by the course instructor on course pages.

C. The adaptation model

The adaptation model (AM) captures the semantics of the pedagogical strategy employed by a course and describes the selection logic and delivery of learning activities/concepts. AM includes a narrative storyboard sub-model supporting course storyboard graphs. It consists of control points (CP) and work paths (WP) meaning the path from one control point to another. Within one storyboard graph, the instructor may create various WPs for different mixes of learning styles. LOs are situated on storyboard pages representing nodes within course storyboard graph. Moreover, AM should provide a schema of storyboard rules used for controlling the e-learning process. Storyboard rules determine sequencing of the course pages upon inputs from learner sub-models. The narrative metadata sub-model sets such rules for given learning style, for passing a CP (e.g., as threshold level of assessment performance at that CP) or for returning back to the previous CP.

D. The adaptation engine

Content pages delivery is controlled by the adaptation engine (AE) for choosing most appropriate both WP (by adaptive navigation) and content (by adaptive content selection, link annotation and hiding, etc.) for presenting it to a learner with particular learning model. Instead of choosing dynamically a page (i.e. node of the storyboard graph) with its content, we propose choice of best working path within the graph for specific learner with given learning style on one hand, and shown prior knowledge and performance on the other. For this purpose, we define storyboard control points (CPs) as nodes of the storyboard graph, where AE either measures learner knowledge/performance, or receives input about satisfaction level of learner’s goals and preferences. When a learner starts a new course, adaptive engine finds the best path for him/her in the course graph. The best path is that one with the highest weighed score. For a particular user, the best path is calculated by a sum of multiplications between page parameters values and weights of their correspondent learner’s characters.

IV. AUTHORING ADAPTIVE COURSEWARE

This chapter provides description of authoring of creating learning objects organized in ontologies and used within the instructor tool to create and maintain specific e-learning courseware. Before starting explanations about the authoring process, we present a brief overview of the system architecture.

A. Principal architecture of an adaptive hypermedia system based on the triangular model

The software architecture of the adaptive hypermedia system being under development is component based. Fig. 4 shows a general view of the system by representing a UML deployment diagram. There are four application clients – one of each of the actors (author, instructor, learner and administrator). The server side components of the author and instructor clients are respectively an authoring tool and storyboard graph and page composers. All of them use a common business API. Learning content is structured by means of usage of XML schema/DTD for LOs and metadata and is stores within a content database, while storyboards and learner models are saved in separate databases. The adaptation engine (AE) takes central part in the system and communicates to the business API and to the administrator and the learner applications. AE executes rules defined either in Drools and SWRL and, thus, assures adaptation of content delivery by means of using the pages and rules mastered with both the authoring and instructor tools.
B. The authoring tool

Our authoring tool is based on reusing the already existing authoring tool of ARCADE (Architecture for Reusable Courseware Authoring and Delivery) e-learning platform [14]. As far as it may run as a standalone application, we have integrated its extended version into our system. In this version (fig. 4), the learning content is presented by learning objects (LOs) connected each other within a semantic ontology tree. LOs may be primitive (containing plain text, table, image, audio, animation, video, external resources, or links) or composite (aggregating other LOs). Composite LOs have hierarchical structure – fig. 4 presents such a object containing two text paragraphs and an image. LOs may have various resource types – narrative content, task, assessment question, etc. Assessment questions are represented as LOs, as well - according the QTI specification. For the moment, the authoring tool supports three questions types – multi-choice, single choice and boolean. For example, for each LO or for several LOs the author may define one or several test questions. Then the adaptive engine may use them for test generation and learner knowledge examination at a control point pointed by the instructor.

LOs are organized by a semantic ontology which may be built by the author before starting creation of LOs or in parallel with it. For creation of the ontology itself, OWL (Ontology Web Language) is used as far as it “permits definition of sophisticated ontologies, a fundamental requirement in the integration of heterogeneous information content” [15]. We make use of OWL Lite to specify is-a and has-a relations, properties, data types, etc.

The LO content is constructed in a Web-based editor accordingly the Sharable Content Object Reference Model (SCORM) standards and specifications for packaging of web-based e-learning content developed by Advanced Distributed Learning Initiative. Each of LOs is described with its metadata accordingly IEEE Learning Object Metadata (LOM). LOM provides opportunities for more effective search for LOs, reuse of learning content and possibilities interoperability with other authoring tools, environments or repositories.

C. Integration between the authoring and instructor tools

The instructor tool is a Web application for creating courses adaptable to different users. Instructor composes a course in terms of interconnected pages represented as nodes.
of the narrative storyboard and connected each other. The narrative storyboard graph is to be processed by the adaptation engine (AE) in order to choose the best path for a particular user. The instructor may browse LO ontology as defined by the author, read LOs and, finally, compose pages with learning content. Moreover, he/she could copy or drag-and-drop branch of the ontology three or only a single LO. Instructors have the ability to filter LO of given type (task, assessment, essay, etc.) in order to choose a proper LO of such a type and to place it onto a page within the narrative storyboard. A course graph may have several control points (CP), i.e., course exams. A course exam is generated automatically based on the learning objects used in pages on the work path leading to that CP, and questions related to these LO (as far as they are designed by the course author and linked to correspondent LO within the ontology tree). Thus, it is not up to the instructor to determine every single question. The instructor can adjust CP thresholds values, i.e., assessment results for passed exam.

Instructor has also the responsibility to annotate page links and to set page weight parameters for each of the characteristics of the learner model (i.e., parameters showing how much given page with LOs is suitable for given learner character). These page parameters are very important for tuning the system. Adaptation engine use them to decide whether particular page would be useful for given user or not. If a page has high value of the parameter for given learner character and this character is dominant for a particular learner, then this page should be principally shown to that particular learner. Thus, if a work path (from the current control point to the next one) contains many pages suitable for particular user while other path do not, than this work path will be nominated for the best path for such a user. Links annotation labels can be added also by instructor to influence user’s decision when a particular user is choosing among several links.

V. CONCLUSIONS

Adaptive hypermedia platforms continue being a challenging issue for modern e-learning. The paper aimed at presenting the authoring process of adaptable content within the context of using the ADOPTA platform. A key issue was separation of the authoring and instructor environments unlike other similar approaches, as far as authors design learning content and instructors master learning design using that contents. In this way, given content may be reused in many courses, by different instructors.

Both the authoring and instruction tools described over are specially designed to support adaptability. The adaptation is based on creating different work paths within given narrative storyboard and adaptive navigation through it and, as well, on adaptive content selection and link annotation through the selected path. Learners are not obliged to follow that path but in any case their assessment will make the adaptation engine to recalculate path’s weight.

Among the key benefits we offer to authors and instructors, are the automatic build of assessment in control points, repurposing of LOs, inheritance of LOM, and metadata for ontologies. Learning object metadata records are going to be used to develop effective search and location of learning objects, and also to develop automated or semi-automated selection and composition tools.

ACKNOWLEDGMENT

The work reported in this paper is supported by the ADOPTA project funded by the Bulgarian Ministry of Education and Science via contract no. DVU01/030.

REFERENCES