

Adaptive Presentation of Evolving Information Using XML

Mária Bielíková

*Slovak University of Technology, Dept. of Computer Science and Engineering
Ilkovičova 3, 812 19 Bratislava, Slovakia
bielik@elf.stuba.sk*

Abstract

In this paper we describe the proposed framework for representation and presentation of evolving information. As an example we use administrative information about an educational module. Evolution is expressed by a set of scenarios and is used for adaptive presentation of information represented as hypermedia. Representation of meta-information is based on the XML (eXtensible Markup Language). We experimented with the proposed approach and implemented a software prototype, which enables the time view and the user view of the educational module information. All presented data are stored in the database. Each element has associated constraints at the level of presentation, intended group of readers and presentation time.

1. Introduction

In this paper we deal with a problem of adaptive presentation of evolving information represented as hypermedia. Our attention is devoted to presentation of information about an educational module, its objectives, reader, lab requirements, schedule of lectures, conditions to obtain a grade, etc. This kind of information is necessary for any kind of education delivery. Usually it is presented on the paper (on whiteboards), or electronically in the form of non-structured documents, or as a hypertext document on the Web (represented in the HTML).

Methods and techniques of adaptive hypermedia are largely designed and evaluated within educational environments. They often focus on the educational materials (e.g., adaptive hypermedia books). Some of them also incorporate administrative information [1, 7] but use the same presentation techniques as for educational materials.

The content of administrative module information is usually solid. The problem is that each overlooked word can cause a student later problems to fulfil requirements to obtain a grade. Although reading all available information can be useful in some cases, we have experience that for the most students it is helpful to see details when they are current (i.e., needed).

Selected example of administrative module information presentation enables discussion related to the specific requirements for such application. Proposed framework can be used for other application domains, which produce evolving documents and this evolution is important for the effective presentation.

Moreover, we exploit the fact that presented information is used repeatedly. This feature is specific for educational environments, where modules run several times (e.g., in subsequent years). Though the context of a particular module changes each time, some presentation patterns can be found and reused.

2. Context-Based Presentation

An important feature of effective delivery of administrative module information is the adaptivity based on the current presentation context. The adaptive presentation is one of the generally recognized ways of adaptation in adaptive hypermedia [6]. Irrelevant information and links that overload the user's memory and screen are hidden, labeled as not recommended, or removed. Often additional information related to recommendations for a reader is presented by different color [8].

Adaptive application uses information about the user (a *user model*) to deliver the content, which is the most interesting for the user (e.g., his knowledge about a particular domain, preferences, previous visits). Adaptive application should define also a *domain model*, describing how the information content of the application is structured ([5]). In adaptive applications with evolving information also a *presentation context* is important. For example, desirable administrative information depends more on time of reading rather than on a complex user model, which represents the user's knowledge related to the information content. Often a static *user profile* provides sufficient representation framework. Adaptation based on the current presentation context would improve delivering information to a reader (e.g., a student).

The presentation context is defined as a collection of data, which depend on the current state of a presentation. The context model does not depend on the user model. However, both models relate concepts from the domain

model in order to define an *adaptation model*. For the purposes of administrative module information delivery, the presentation context can be reduced to considering time of information reading. However, the context model can represent also data important for adaptive presentation in general. Example of such data is browser or language used. Recognizing the language is interesting in the case of a multi-lingual educational module, which is often the case in non-English speaking countries. In this case, we need several variants of the presented document and have to deal with the problem of hypermedia document version management (see for example [2]).

Keeping modifications introduced to the pages is also important feature of a good educational module information presentation system. In traditional learning environments, the modification of the module material is not frequent and usually not interesting for a student (student receives the package with learning material, and modifications are usually incorporated into a new version of learning material used in the next module run). Modification is frequent and important in presentation of organizational aspects of the module. Visibility of a single modification is related to both user model and presentation context.

3. Example of Adaptation

For adaptive presentation the granularity of the information content is important. In the case of administrative information about an educational module, a fragment is defined as a text paragraph (or sequence of several subsequent paragraphs) together with embedded media (e.g., pictures or video sequences). Another important entity is a hyperlink, which should be manipulated separately. Each of identified fragments (a sequence of text paragraphs, or a link) is described by

several attributes, which state fragment's visibility according the user and context models.

In our example we consider a simple user model based only on a specification of several groups of users. The user model does not evolve, i.e. the system is not personalized. Issues related to the user modeling aspects according to adaptive presentation or adaptive navigation are widely discussed in the literature (e.g., [5, 7, 8]) and can be used for extension of the proposed approach to the adaptation.

The presentation context is expressed by the current time. The information content is structured to fragments and fragments are grouped according an evolution scenario. The evolution scenario defines steps of presentation according the progress in the course of a module run. It represents a kind of pattern for information delivering during some period of time. The scenario can be defined in a week term for traditional course delivery. In the case of distance education, a user's achieved state is more relevant criterion for moving to the next step in the scenario.

Figure 1 illustrates part of the appearance of presented information for the Knowledge-Based Systems module (a master course at the Slovak University of Technology). Each page starts with a generated header, which contains text defined in the first element of the fragments' hierarchy and the current date. Header is automatically included in each page. The page content is generated according to defined domain model, which is the XML-based document. Each page contains a frame on the right hand side, which serves as a content navigation within a current page. Different data formatting can be expressed by a style sheet. Alternative style sheets can be applied at any time, changing the format to suite intended audience (e.g., format of the university web pages) and the capabilities of the used publishing media.

KBS, Friday, January 28, 2001		
Name of the module	Knowledge-Based Systems	Contents
Lecturer	Mária Bieliková	
Objective		Home page
<i>One major insight gained from early work in problem solving was the importance of domain-specific knowledge. A doctor, for example, is not effective at diagnosing illness solely because he possesses some general problem-solving skill; he is effective because he knows a lot about medicine. Expert systems are constructed by obtaining this (expert) knowledge from a human expert and coding it into a form that a computer may apply to similar problems.</i>		
<u>Syllabus</u> ►		
<u>Laboratory exercises</u> ►		
<u>Frequently asked questions</u> ►		
<u>Conditions to obtain a grade</u> ►		

Figure 1. Example of module information presentation.

Presentation is adapted both on the content and navigation according to the user profile and the context model. In order to reduce the amount of information displayed at once, the system can create for each fragment a link navigating to information content represented by a particular fragment, or split the content into several parts presented independently and reached by the hypertext links.

4. Data Representation

The module information is represented by the XML (eXtensible Markup Language). The XML is a meta-language, which describes a data format suitable for storing structured and semistructured text. The XML-based document is logically structured into named units and sub-units called elements.

5.1. Content

Each element represents a fragment in a domain model. It has defined adaptation conditions in the form of required values of defined attributes. Elements can form an arbitrary hierarchy and can be named by the user (who is responsible for the module development). Presentation of information copies this hierarchy. In order to present ‘pages’ with reasonable length, some elements with long text and/or many offsprings can be displayed by links.

Granularity of a document (the degree to which element’s content is organized into offspring elements) is determined by a developer. The presentation method only requires declaring conditions (as predefined attributes of elements) and the content of elements. Some elements however, do not have content in the form of text – they just name the hierarchy. This kind of elements is useful in case of need to split the presentation into several pages.

Each educational module has its own Document Type Definition (DTD), which defines the elements allowed in a particular document of the module type. Several modules can use predefined common DTD.

Bellow is an example of part of DTD declaration:

```
<!DOCTYPE Module [
  <!ELEMENT Module (Header, Body, Foot)>
  <!ELEMENT Header (Module-name,
    Module-Abbreviation?,
    Study-Type?)>
  <!ELEMENT Module-name (#PCDATA)>
  ...
  <!ELEMENT Body (Info?, Objectives,
    Syllabus, Labs, Faq?)>
  ...
  <!ELEMENT Foot (Modification-date,
    Administrator-name?)>
  ... ]>
```

A major decision in producing adaptive hypermedia systems is the decision about structuring information in

such a way that it will be possible to adapt presentation. In our approach, the XML element according to the module data type definition is the smallest information chunk that can be adapted. However, the structure of the module information and the structure of the content to be adapted and presented can be of different granularity. Developer can define tags, which reflect the structure of presented information (e.g., paragraph tag <p>), but are not related to the structure of the module information.

Information related to a new module can be developed by the use of predefined templates or from the scratch.

5.2. Meta-data

Actual data related to the particular module are represented by the XML-based document according to its DTD. Each element contains several conditions, which define the model of presentation according to a given context. Attributes and their values within each element represent conditions. We have experimented with attributes related to the type of a user, granularity of presentation, time and splitting of the presented information to several pages. Set of available attribute-value pairs can be enhanced. In this case, their interpretation should also be defined.

Bellow is an example of meta-data related to the syllabus element:

```
<syllabus description="Syllabus"
  from="01/01/2001" to=""
  user_operator=">=" user="guest"
  mode_operator="=" mode="medium"
  visible="yes" link="yes" include="no">
  <B>Background:</B>
  A brief history of expert systems;
  expert system concept;
  ...
</syllabus>
```

All meta-data defined in the syllabus element are evaluated according to the current state. If date of presentation is after the 1st of January 2001, this element is presented to any user ($user \geq guest$) who have selected the medium granularity ($mode = medium$). When the syllabus element is presented, the text “Syllabus” is displayed. This text is marked as link ($link = yes$), which results in hiding the actual text of syllabus into the separate page (reachable through this link, see Figure 1). When the link attribute is set to “no”, all text defined for this element is displayed.

6. Adaptive Presentation

We use the conditional technique for the adaptation [6]. With this technique, all information about a module is linked to the condition at the level of a user or context model. When presenting information in the particular

context and for the particular user, the system presents only information with the condition evaluated to be true. A simple example is hiding detailed information about the lab exercises in the first week of a semester. The purpose of hiding this information is not to disable view of the information, but prevent information overload at the beginning of the semester. Solution to this requirement is to disable some information in short or medium views in the first week, but enable it in the full view. Later, when information is current, it is presented also in short or medium views. When the presentation of the module information is required, attributes of each element are evaluated and compared against the current context and adapted module information is displayed.

We developed a software prototype for adaptive presentation of administrative module information [3]. The software prototype uses the following data about the current state: group of the users, granularity of presentation and context represented by the current time.

We also developed a tool for supporting the design of the adaptive hyperdocument, including definition of fragments, conditions of their adaptation and the content. A developer can create and modify a hierarchy of elements and define meta-data and the content of each fragment.

Design and implementation of a prototype was oriented mainly toward prototyping the proposed framework for adaptive presentation of the module information. However, the effective using of such system in the Web environment requires effective storing and manipulation of the data about several modules, effective storing and manipulation of the data about users and some level of security (in particular at the level of authentication for modification of the XML documents related to the particular module). Due to the aims of the prototype development, the problem of security was not solved. Data manipulation efficiency is tackled by the design and implementation of a relational database, which serves for storing the whole content of data elements together with the data about users and developers. We used the database server MySQL.

7. Conclusions

In this paper we discussed possibilities of improving current state in presentation of evolving information. As an example we used administrative information related to an educational module.

A domain model in our system is represented as a hierarchy of fragments. It is sufficient for this specific application area. However, in the case, where more complex relations between fragments are required, advanced representation be used ([1], or [7]).

We use the conditional content adaptation technique for adaptation, similarly like the AHA system [4]. Unlike

the AHA system, where conditionals are encoded in the structured HTML comments, we represent conditions as the attribute-value pairs in the XML. Conditions are expressed declaratively as values of attributes. They are interpreted in our prototype by the PHP3 scripts.

An evolution scenario can be defined manually or semiautomatically by developing a software agent watching on the first manual adaptation of presented information. We are working now on a development of such a software agent.

Acknowledgements

The work reported here was partially supported by Slovak Science Grant Agency, grant No. G1/7611/20.

References

- [1] L. Aroyo, and D. Dicheva. Domain and User Knowledge in a Web-based Courseware Engineering Course. *Knowledge-Based Software Engineering*, T. Hruška, M. Hashimoto (Eds.), IOS Press Amsterdam, 2000, pp. 293-300.
- [2] M. Bieliková, and P. Návrát. Modelling Versioned Hypertext Documents. *SCM-8 Symposium*, B. Magnusson (Ed.), Springer Berlin, 1998, pp. 188-197.
- [3] M. Bieliková. Adaptive Presentation of Educational Module Information using XML. *Knowledge-Based Software Engineering*, T. Hruška, M. Hashimoto (Eds.), IOS Press Amsterdam, 2000, pp. 301-306.
- [4] P. De Bra, and L. Calvi. AHA: A Generic Adaptive Hypermedia System. *Adaptive Hypertext and Hypermedia*, Pittsburgh USA, 1998, pp.5-11.
- [5] P. De Bra, G.J. Houben, and H. Wu. AHAM: A Dexter-based Reference Model for Adaptive Hypermedia. *ACM Hypertext'99*, Darmstadt, Germany, 1999, pp. 147-156.
- [6] P. Brusilovsky. Methods and Techniques of Adaptive Hypermedia. *User Modeling and User Adapted Interaction*. Vol. 6, No. 2-3, 1996, pp.87-129.
- [7] N. Henze, and W. Nejdl. Extensible Adaptive Hypermedia Courseware: Integrating Different Courses and Web Material. *Adaptive Hypermedia and Adaptive Web-Based Systems*. P. Brusilovsky et. al. (Eds.), Springer, 2000, pp.109-120.
- [8] G. Weber, and M. Specht. User modeling and Adaptive Navigation Support in WWW-Based Tutoring Systems. *User Modeling*, A. Jameson, C. Paris, C. Tasso (Eds.), Springer, 1997, pp. 289-300.