# University Course Support by Web-Based Adaptive e-Board

Mária Bieliková and Rastislav Habala
*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*

## Abstract

*In this paper we present an adaptive hypermedia web-based system called TIM (TIMe-Aware Adaptive Hypermedia System). The time is used as a part of the context model, which along with a user model constitutes a base for the adaptation to the user needs. We have proposed an extension of known adaptation techniques by means of the notion of time. Proposed extensions are based on the model of time, which maps a real time to a symbolic time and provides necessary operations for working with both of them. TIM implements proposed adaptation techniques. Its domain model has been created to serve as a web-based adaptive e-board for a university course. The e-board system adapts presentation of the educational module administrative information according the user characteristics and the context represented by the time of information presentation.*

## 1. Introduction

In web-based educational systems, the adaptation to the user's or environment requirements is becoming a major requirement. Adaptive hypermedia systems build a model of the goals, preferences and knowledge of each individual user, and use this model throughout the interaction with the user, in order to adapt to the needs of that user [3]. For example, a student in an adaptive educational hypermedia system will be given a sequence of programming exercises, which suit best to his knowledge and current state of practice [13].

User knowledge, preferences or goals are major source for adaptation in educational hypermedia. Another important feature of effective information delivery is an adaptivity based on the presentation context. The presentation context is defined as a collection of data, which depend on the current state of presentation. It is often described using technology or user platform adaptation dimension (hardware, software, network bandwidth) or external environment adaptation dimension (user location, language, socio-political issues) [5]. In our work we emphasize one of important characteristics to the context – time. Time is fundamental characteristic since the context and user characteristics are unstable and changing. Taking the time into account can indicate for example that some information is not valid at the presentation time. The consequence is that it does not make sense to present the information (any more or in the specific period of time).

Nowadays there is a limited emphasis on using time in adaptive hypermedia (AH). The aim of this paper is to present a possibility to employ time-based adaptation in e-learning context. We have extended known adaptation techniques by means of a notion of time. Proposed extensions were experimentally evaluated in the domain of adaptive presentation of information related to educational university courses, which is characterized by frequent changes of presented information. Moreover, we support a reuse of the domain knowledge about the content or navigation in the context of time (e.g., reuse of administrative information in subsequent years of the course offering). We developed a web-based software system called TIM (TIMe-Aware Adaptive Hypermedia System). Its domain model has been created to serve as a web-based e-board, i.e. to present university course administrative information regarding a user model (with characteristics such as the user's previous visits, goals, preferences, etc.) and a context represented by the time.

The rest of the paper is as follows. In Section 2 we describe models of time-based adaptive hypermedia. We concentrate on the models designed for our TIMe aware adaptive hypermedia system, accordingly special attention is devoted to the model of time. Next, using proposed model of time, we show time-based extensions of known adaptation techniques. We present change emphasizing as a technique essential for the web-based adaptive e-board. In Section 4 we describe our TIMe-aware adaptive hypermedia system, which implements proposed model of time and time-based extensions to adaptation techniques. The paper concludes with related works, summary and future directions in this research.

## 2. Adaptive hypermedia models for a time aware system

The key elements of adaptive hypermedia application models are the following [16]:

- *a user model* containing the user specific data related to the information content; extended in later approaches by *a context model* describing an environment of the information presentation [2, 5, 7],
- *a domain model* describing a structure of the information content together with actual content (the information base containing the material presented to a user),
- *an adaptation model* consisting of a specification of adaptation knowledge (frequently represented using adaptation rules) and an adaptation engine (responsible for performing an adaptation based on the adaptation knowledge according the user model).

In this section we describe above listed models as designed for our TIMe-aware adaptive hypermedia system. We concentrate on the specific features designed for time-based adaptive hypermedia system, i.e. proposed model of time (which is a part of the context model).

### 2.1. Domain model

A domain model is used to describe the hypermedia basic contents and concepts. It consists of concepts and concept relationships [16]. The domain model is represented as a graph, in which each node represents a concept and edges represent relations among them. Concepts and relations have additional attributes (e.g., name, type, presentation, visibility). Concepts are atomic or composite, the first representing individual pages, the latter abstract groups of lower level concepts (in our system by means of the *collects* relation). The concept hierarchy is defined by the *isParent* relation. The *presents* relation allows to include into a concept the reference (a link) to the information content of another concept or directly include the content of one concept into another concept.

The *isPrerequisity* relation expresses suitability of a concept presentation according to the presentation of the dependant concepts. We proposed the *isPrerequisity* relation with a parameter of four possible values, which determines adaptation if all prerequisites of the concept have not been presented yet:

- *hide*: recommended adaptation is to hide the link,
- *asText*: recommended adaptation is to present the link as a text (navigation is not possible),
- *dissuade*: recommended adaptation is to present the link as non recommended concept (using an appropriate color),
- *recommended*: recommended action is to present the link as recommended (using appropriate color).

### 2.2. User model

The user model contains data about the user, assumed level of his knowledge, preferred learning style, etc. The two most widely used approaches to the user modeling are the overlay model and the stereotype model [3].

In the stereotype model each user is assigned a stereotype, which designates that he is a member of a group that shares some characteristics. The overlay model provides more detailed information about the estimated user knowledge because it stores individual value for each concept, page or information fragment. The result is the copy of the domain model filled with the user's data.

The TIM user model is the overlay type of the user model. However, we designed an alternative representation of the models using a set of instructions. An instruction represents the elementary operation on one of defined models (domain model, user model or model of time). Instructions can be defined in the time of authoring (e.g., concept creation, setting an attribute), or in the time of adaptive hypermedia system usage (changing the domain model or visiting a concept). Each instruction has defined a timestamp (the time of its creation). A set of instructions can be created by an authoring tool or by the adaptation engine. Models represented by instructions are used by means of the interpreter.

This feature enables simple definition and the usage of a stereotype user model. The stereotype is defined by a set of instructions. Though, we use a combination of both approaches to user modeling in dependence on the amount of available data about the user (start with a stereotype model and later switch to the overlay model).

### 2.3. Model of time

Primary goal for time-based adaptive presentation is the ability to designate *valid time* of an information fragment presentation (explicitly or implicitly). As a base we have used temporal models used for temporal databases [9, 10]. The valid time of an information fragment is the time when the content is actual for particular user within particular context (environment or technology used characteristics).

A simple technique uses a start and an end time of validity settings for each information fragment. For example, if there is a requirement to present a fragment between 14th May 2003 and 20th May 2003, the following meta-data is assigned to the fragment:

*start_time = 14th May 2003* and
*end_time = 20th May 2003*

If the presentation of a fragment that is valid in repetitive intervals is required, the time of validity is set in one of the following forms (or their combination):

*start_time = Monday* and *end_time = Friday*

or

*start_time = 14:00* and *end_time = 18:00*

Described approach of an explicit valid time specification is suitable in cases whenever we are able to specify exact time of the fragment validity. In most cases implicit specification is necessary. Let us take an example from an application of our adaptive system TIM. The primary objective of developed adaptive hypermedia system is efficient presentation of university course information. The content related to a university course is usually solid. The point is that each overlooked word can cause a student later problems to fulfill 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 only when they are current.

Following above stated requirement, a scenario of a consecutive course related information presentation should be defined. First, an initial set of information fragments is specified. The content evolves (and is possibly supplemented or modified) during the course running. For example, we would like first to publish a basic structure of required project documentation (at the beginning of the course). Then, in the 3rd week of term we would like to publish project documentation detailed structure.

With the model of time it is possible to handle described situation. The model maps a symbolic time to the real time and provides basic operations on both the symbolic and the real time. Several time types may be used when modeling the valid-time aspect, e.g., single time instants, intervals, or sets of intervals (scopes). An *interval* represents continuous time period. This period has a start time, an end time and a unique name. For example, `Mon_9week` is a name of the interval, which starts on Monday 19.4. 2004 00:00 and finishes on Tuesday 20.4. 2004 00:00 (see Figure 1).

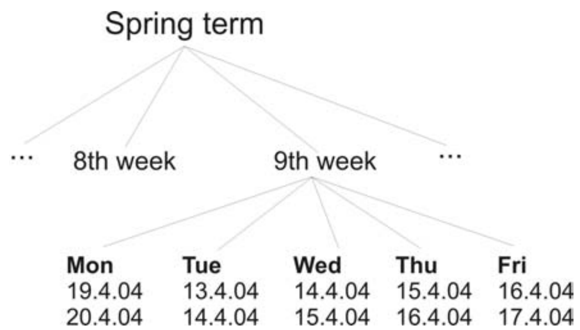


**Figure 1. Intervals and scopes**

*Scope* is an ordered set of intervals or other scopes. Each scope has a unique name. In Figure 1, 9th week is an example of the scope containing intervals (from `Mon_9week` to `Fri_9week`) and summer term is an example of the scope containing other scopes.

We use the model of time with defined intervals and scopes corresponding to the weekdays, the weeks and the term, and set intervals' start and end time of validity with actual values to handle mentioned situation regarding publishing the fragment from the 1st week to the 3rd week of a term.

Three types of time-based operations are defined:

– *real time* → *real time*, e.g., a test whether two real times are identical, determining the order of real times (*before, after*);
– *real time* → *symbolic time*, e.g., a test whether given real time occurs in the interval pertaining to a given scope, whether it is before or after a given symbolic time, or determining the order within interval/scope (e.g., consecutive number of a week day);
– *symbolic time* → *symbolic time*, e.g., a test whether two symbolic times are identical, determining the order of symbolic times.

Time constrained reasoning in the AH system exploits defined operations. It is possible to determine the real time corresponding to the given symbolic time using defined operations. The model of time is advantageous especially in cases when the domain model is going to be reused in various time periods, i.e. similar content is presented within various time intervals. Typical example of such functionality is a web-based whiteboard of a university course, web-site of a scientific conference or presentation of other repetitive activities.

The domain model reuse in next execution of the event (be it a university course or a scientific conference) requires only setting-up the intervals' validity times. We can simply model, for example, irregularities in the course schedule caused primarily by national holidays. A logical week of a term can consists of days, which do not come under one physical (calendar) week (notice that the logical date *Monday on 9th week of term* in Figure 1 is shifted to the next calendar week because the Easter holidays).

## 3. Time-based extensions of adaptation techniques

### 3.1. Adaptive presentation and navigation

Web-based adaptive hypermedia systems use several approaches to perform an adaptation. The most visible and therefore most popular is an *adaptation of the presented content or its layout*. For example, presentation of the same content is prepared by the adaptation engine, thus a user gets different output according to his current needs. The most widely used technique is the content highlighting by means of using various colors and/or pictures. The content adaptation involves inserting/ removing fragments, altering fragments, sorting or dimming fragments [3, 4]. For example, the content that is not likely to be seen by a user is hidden or grayed out. Or the new information fragment supposed to be useful for the user is inserted.

Typical adaptation of the web-based content providing systems is *adaptation of the navigation*. The following a link in a standard web application results in a display of the requested page. Adaptation featured systems may include additional processing into the link resolution. For example, if a user clicks a link to display the content for a concept, the system may offer him a series of pages that the user should read before the requested one (in case he had not already read them).

Typical time-based extension is a determination of an information fragment presentation according the time (along other characteristics represented in a user model and a context model). Obvious technique is to define meta-data related to the fragment, which serve for the adaptation. In our system a value of the *visible* attribute determines a visibility of the fragment (its inserting or removing). Adaptation rules define adaptive fragment inserting or removing in dependence of the time.

Simplest implementation of the inserting/removing fragments technique uses attributes *from* and *to*, which represent simple time dependent rule:

*the fragment is presented only if current time is in the interval defined by the from a to values.*

If the value of these attributes is not defined, we consider infinity instead. Values of *from* and *to* attributes can represent real time or symbolic time.

Sort criteria are also extended by time (real or symbolic). This technique is especially useful for the domain of adaptive e-boards. Typical information fragments that should be sorted by the time are organized as a list of announcements or messages to the students. The *originTime* attribute is used. Its value represents a start time of the announcement validity (notice that the origin time is usually different from a timestamp because the time of a fragment creation is obviously different from the time of its intended publication). The *orderBy* attribute determines sort order.

Sorting fragments technique is proposed along with the *collects* relation. The *collects* relation enables effective wrapping of an evolving list of fragments. Adding a new element into the list does not require any modification of the concept representing the list.

We provide bellow an example of the content of such wrapping concept. Figure 2 illustrates its visualization on adaptive e-board (in the Announcements part of the content).

```
<p class="Heading0">Announcements</p>
<p>Here you can find up-to-date
   anouncements devoted to the Principles
   of Software Engineering course....</p>
<ul>
  <collect accordingTo="originTime"
                 orderBy="descending"/>
</ul>
```

Similar extension is applied to adaptive link sorting and hiding. For example, fresh concepts are presented first. Sorting can be based on the concept creation time or the concept validity time. Adaptive link hiding regards the time while a decision about the link presentation is performed (along other characteristics represented in the user or context models).

### 3.2. Change emphasizing

For effective comprehension of evolving information it is important to have a mechanism for adaptive presentation according the changes, which have occurred from the last user visit and to which the user should pay attention. We proposed the change emphasizing technique as follows. The adaptation engine compares the content of presented fragment that has been valid in the time of its last presentation with its current content. The distinctions are determined and emphasized to the user. Moreover, changes can be propagated to the parents of the changed fragment according defined concept hierarchy (i.e., the change is presented also in the fragments, which occur on a path from the starting fragment to the changed fragment according the *isParent* relation). The changes are indicated by annotating links leading to the changed fragment.

Meaningful change discovery is not an easy task. The problem resides in an automatic recognition of the change significance. Sometimes the change presents a spell correction. Some other time the content is renewed but it is not necessary to read it again. Our system requires for change detection its explicit tagging in the content. The changed region is enclosed to the block with assigned time of the change.

For change emphasizing implementation we proposed a technique for the time versioning of adaptive hypermedia models (used mainly for the domain model). This technique enables us to determine the state and the content of each concept (or fragment) in arbitrary time in the history. It is a base for accomplishing a comparison of the content of two snapshots of the concept. Time versioning is also useful in cases when a user wants to see a state of the domain model in particular time. For example, a user requires inspecting the last year conference program. Selecting specific time one year ago the system performs an inference with the specified time and presents the last year conference program.

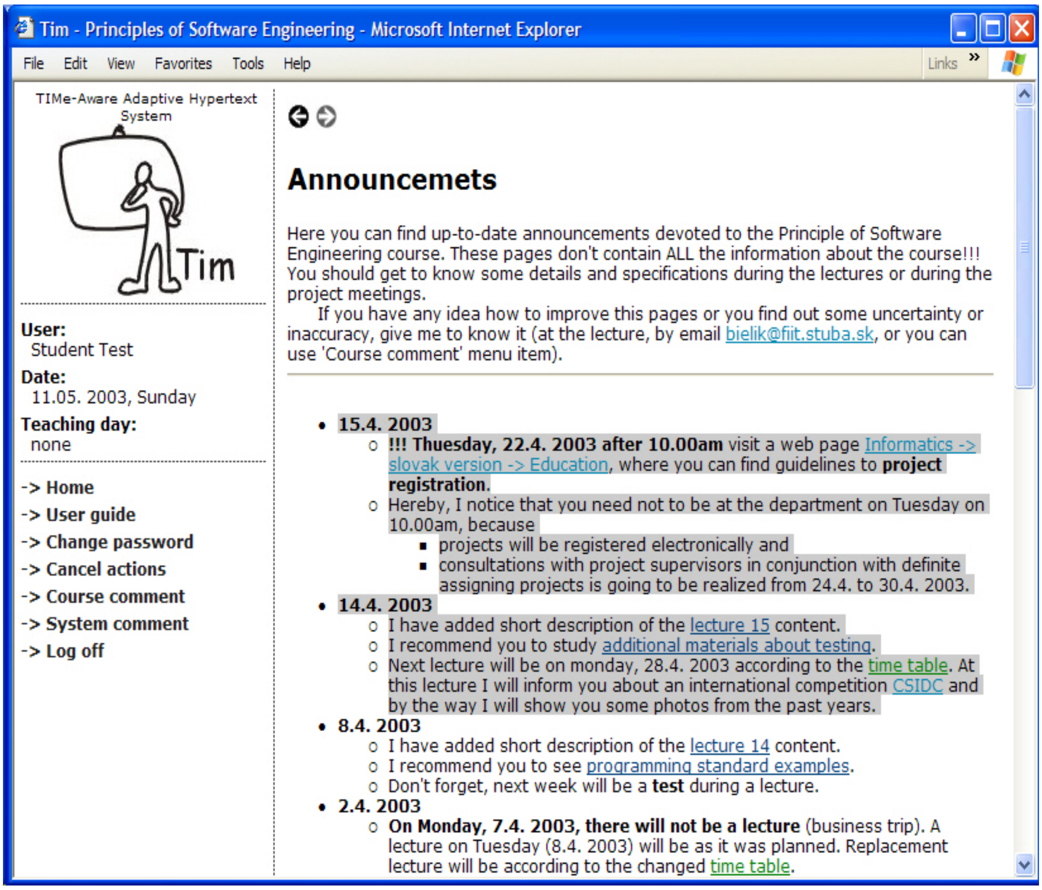


**Figure 2. Change emphasizing in TIM**

Change emphasizing was recognized as extremely useful feature for students who periodically visited the course information during the term. For a visual change highlighting we use a change of the background color (see Figure 2). Advantage is that it is possible to use the change emphasizing in a combination with known adaptation techniques for fragments annotation. Accordingly, links inside the changed content can be annotated (e.g., colored).

### 3.3. Time-based adaptive link annotation

Time-based adaptive link annotation technique supports an increase of a user attention to a change, which has occurred in the concept representing a target of the link. We distinguish two types of changes. Providing the concept content has changed or a new fragment has been added to the concept content it is denoted as a "change". Other case is adding a new fragment using the *collects* relation. This fact is visualized by annotating with the "new" icon (it reflects semantics of the *collects* relation and its intended use for e-board systems).

Adaptive link annotation by denoting changes or news is propagated (as mentioned in the previous subsection). However, sometimes it is necessary to stop the change propagation because as a result we will see the change indication in many places, which can confuse the user.

Three presentation states are possible:

– *none*: no changes are indicated,
– *explicit*: only explicit changes are indicated, i.e. changes that were initiated directly in the referred concept and indicate the content change,
– *implicit*: implicit changes are indicated; a concept is implicitly changed then explicit or implicit change is propagated from at least one referred concepts; implicit change is indicated also if a new fragment is added.

Naturally, known uses of the adaptive link annotation technique are applicable together with its time-based extensions. For example, the *isPrerequisity* relation is used for coloring recommended links.

## 4. Experimental e-board system

Proposed models and techniques for time-based adaptation have been evaluated through the design and implementation of time-based adaptive hypermedia system called TIM (TIMe-Aware Adaptive Hypermedia

System). We designed a domain model devoted to the presentation of administrative information related to an educational course such as 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 set of hypertext documents on the web (represented in HTML).

Important feature of such content is its frequent change, which leads to a time-dependence. TIM has been first time applied in summer term 2002/03 to the Principles of Software Engineering (PSE) course with about 130 students enrolled. TIM serves the students in two ways: it features as a leading familiarization with a course and it serves as a presentation of the course running (and evolving) information.

TIM is designed using the AHAM model [16]. Adaptation is performed according the adaptation rules defined at the level of concepts, groups of concepts or at the system level. An overlay user model combined with stereotypes is used. Stereotypes are employed for initial settings. For example, at the beginning of the PSE course running three stereotypes were defined: a student enrolling the course, a teacher and a visitor. Students were further divided into groups according the project meeting times.

Both the content and meta-data are described using XML. The TIM architecture is based on a three-tier model comprising the presentation, the application and the data layers (see Figure 3). Main task of the Adaptation Module is preparation of a presentation adapted to a user and the presentation context. The Models Management Module performs operations on defined models (domain model, user model, time model). In the case of a model change the Models Management Module except providing the changes creates a set of instructions corresponding to the change and ensures their storing in the database. The Interpret Module is able to transform a set of instructions to the graph representation.

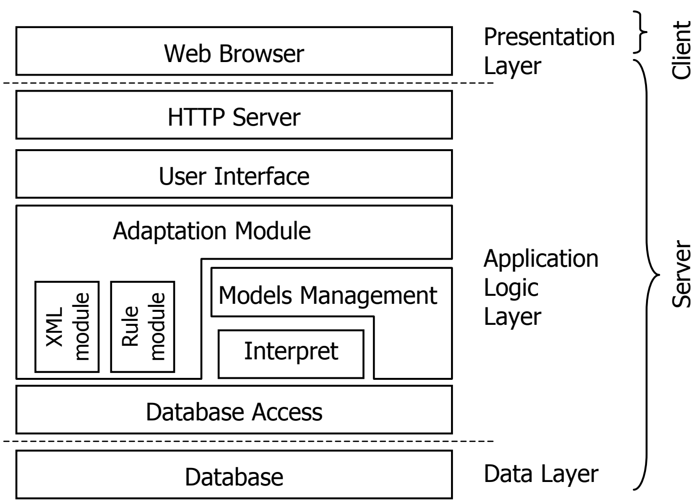


**Figure 3. TIM architecture**

The system is implemented using JavaServer Pages (Java SDK version 1.4), Tomcat web server and MySQL database server.

We evaluated developed system from several points of view. First, the usage of the system demonstrated its appropriateness for presenting evolving information. Invaluable feature for the content developer is the domain model reuse (in subsequent years of the course offering).

As the most frequent problem indicated by the students was known problem with adaptive web-based systems – the *back button*. The user expects a specific behavior when pressing "back" in a browser menu. However, in adaptive systems (in time-aware systems especially) there is no *conventional back* because the state of presentation meantime changed (i.e., the previous presentation state specification would be likely interpreted in a browser to a different information fragment as the last visited fragment). Even though the system provides own back and forward buttons with expected semantics, the students used these buttons infrequently.

The most valuable feature designated by students was the propagation of changes. The emphasizing based on highlighting the background was proved effective similarly to dimming adaptation technique based on shading background [12].

## 5. Related Works

Brusilovsky identified six kinds of AH systems [3] where educational hypermedia are presented as one of the most popular. In spite of the fact that the maturity of adaptive educational hypermedia systems is indicated [4], it should be considered only according some aspects of the adaptation. Adaptation decisions in majority of adaptive educational hypermedia systems is based on taking into account only various user characteristics represented in a user model. Adaptive (educational) hypermedia system is often described as "a hypermedia system, which reflect some features of the user in the user model and applies this model to adapt various visible aspects of the system to the use" [11] (various approaches use slightly different terminology with essentially the same meaning).

Considering the context in AH systems can be useful in many situations. A number of AH systems (mainly on-line help systems or on-line information systems) distinguish adaptation to a user model and a context model. A context model is entitled differently in various works, e.g. environment model [4]) or external environment and technology dimensions [6]. The context model comprises all aspects of the user environment that are not directly related to the current user. We emphasize *time* as a fundamental characteristic of the context model.

Time in AH systems can be exploited in several ways. First option is targeting the limited capacity of the human memory. A human receives amount of information and as time goes he is forgetting part of it. The task of the AH system in this case is to observe this attribute while adapting presentation (e.g., reminding important information after some period of time) [1]. This feature is especially useful for educational AH systems, which serve as a mean for learning.

Second possible use of time in effective adaptation is time constrained adaptive presentation. It is based on the fact that some information is not valid forever. Presented information has in this case defined an start time of validity and/or end time of validity [2]. Moreover, during the validity time, relevance of this information can evolve. The task of the AH system in this case is to observe current time of information presentation besides other characteristics (user knowledge, user preferences, etc.). Time constrained adaptation based on a time of the day is often used in tourist guide adaptive systems (e.g., [7] or [14]). The opening times of attractions represent important characteristic used in adaptation of the city guide. The time in [7] is captured in an environmental context and represents the time of day and the opening hours of attractions.

Third opportunity presents drawing a user attention to the domain model changes. Often the domain model evolves in time (e.g., new information is added or existing information is corrected). It is important to be able to observe this evolution. The AH system in this case serves a user the presentation of evolving information (e.g., changes of the content which occurred from the last user access to the AH system are highlighted).

Other option is to provide models of versioning related to the time (also known in software engineering as revisions [8]). A user can compare current state of models with a history state in some time. The user can also accomplish similar operation related to a history of models. AH system should handle this by providing time versioning – an ability to reconstruct any of defined models in certain time. Several existing AH systems consider versioning. However, in most cases they only define variants (alternative presentations of the same concept) [15].

We concern time-based adaptation, which extends second and third described possible uses. Defined model of time enables natural extensions of known adaptation techniques. New adaptation technique – change emphasizing – uses a change of the background color for emphasizing (unlike the approach of Hothi et al. [12] where the background color change serves for dimming fragments). Alternative representation of the models enables models versioning together a combination of overlay and stereotype user models.

## 6. Conclusions

In this paper we presented an approach to time-based adaptation of the web application presenting an information content. It extends existing modeling framework ([16]) with the model of time. The adaptation techniques employ the operations defined in the model of time.

Our approach enables adaptive presentation of time dependent information, tracking evolution of application domain model in time together with a support of drawing the user's attention to the content changes and time-based content versioning. Important characteristic of proposed approach is the support of a domain model reuse in various time periods. Using two representations of the models – traditional graph representation and instruction-based representation enables a reuse not only the content of a model (e.g., concepts and relationships) but also the evolution of the model.

Proposed extensions to known adaptation techniques add time as a new dimension into adaptation engine reasoning. Time-based adaptivity can be implemented as an independent dimension, which is related to various characteristics considered for adaptation (represented in the user model or in the context model). Such understanding of the time is more suitable for AH systems than considering the time just as one variable of the context model (as proposed in the XAHM model [5]).

Time-based adaptation is suitable especially in application domains where the content evolves (and evolution has a repetitive character). The model of time can be used also in adaptive guide systems where usually time-based adaptation (if present) is wired in an adaptation component.

Our future work concerns the research of time-based adaptation in different application domains together with improving mechanisms for debugging defined models by a simulation of the time pass.

## Acknowledgements

This work has been supported by the Grant Agency of Slovak Republic grant No. VG1/ 0162/03 "Collaborative accessing, analysis and presentation of documents in internet environment using modern software tools".

## References

[1] Ágh, P., M. Bieliková, "Considering Human Memory Aspects to Adapting in Educational Hypermedia", In *Proc. of AH2004: Workshop on Individual Differences*. Eindhoven, The Netherlands, August 2004, accepted.

[2] Bieliková, M., “Adaptive presentation of evolving information using XML”, In T. Okamoto et al. (ed.), *Proc. of IEEE Int. Conf. of Advanced Learning Technologies – ICALT 2001*, pp. 193-196, Madison, USA, 2001. IEEE Press.

[3] Brusilovsky, P., “Methods and techniques of adaptive hypermedia”, *User Modeling and User-Adapted Interaction*, Vol. 6, No. 2-3, pp. 87-129, 1996.

[4] Brusilovsky, P., “Adaptive hypermedia”, *User Modeling and User-Adapted Interaction*, Vol. 11, No. 1-2, pp. 87-100, 2001.

[5] Cannataro, M., A. Cuzzocrea, and A. Pugliese, “XAHM: an adaptive hypermedia model based on XML”, In *Proc. SEKE'02*, pp. 627-634, Ischia, Italy, July 2002. ACM Press.

[6] Cannataro, M. and A. Pugliese, “XAHM: an XML-based adaptive hypermedia model and its implementation”, In *Proc. of 3rd Workshop on Adaptive Hypertext and Hypermedia*, Arhus, Denmark, August 2001.

[7] Cheverst, K., K. Mitchell, and N. Daies, “The role of adaptive hypermedia in a context-aware tourist GUIDE”, *Communications of the ACM*, Vol. 45, No. 5, pp. 47-51, May 2002.

[8] Conradi, R. and B. Westfechtel, “Version models for software configuration management”, *ACM Computing Surveys*, Vol. 30, No. 2, pp. 232-282, June 1998.

[9] Etzion, O., S. Jajodia, and S. Sripada, *Temporal Databases: Research and Practice*, Springer LNCS 1399, 1998.

[10] Gregersen, H.C. and S. Jensen, “Temporal Entity-Relationship Models - a Survey”, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 11, No. 3, pp. 464-497, 1999.

[11] Henze, N. and W. Nejdl, “Logically characterizing adaptive educational hypermedia systems”, In *Proc. of AH2003: Workshop on Adaptive Hypermedia and Adaptive Web-Based Systems*, pp. 15-28, Budapest, Hungaria, 2003.

[12] Hothi, J., W. Hall, and T. Sly, “A study comparing the use of shaded text and adaptive navigational support in adaptive hypermedia”, In C. Strapparava et al. (ed.), *Proc. of Int. Conf. on Adaptive Hypermedia and Adaptive Web-Based Systems*, pp. 335-342, 2000. Springer LNCS 1892.

[13] Kostelník, R. and M. Bieliková, “Web-Based Environment using Adapted Sequences of Programming Exercises”, In M. Beneš, (ed.), *Proc. of Information Systems Implementation and Modelling – ISIM 2003,* pp.33-40, Brno, 2003.

[14] Petrelli, D., E. Not, M. Zancanaro, C. Strapparava, and O. Stock, “Modelling and adapting to context”, *Personal and Ubiquitous Computing*, Vol. 5, pp. 20-24, 2001.

[15] Schraefel, M.C., “Contexts: Adaptable hypermedia”, In C. Strapparava et al. (ed.), *Proc. of Int. Conf. on Adaptive Hypermedia and Adaptive Web-Based Systems*, pp. 369-374, 2000. Springer LNCS 1892.

[16] Wu, H., E. de Kort, and P. De Bra, “Design issues for general-purpose adaptive hypermedia systems”, In *Proc. of Hypertext 2001 – HT'01*, pp. 141-150, 2001. ACM Press.