

# Adaptive ontology-based navigation

Petr Šaloun<sup>1</sup> and Zdenek Velart<sup>2</sup>

<sup>1</sup> Department of Informatics and Computers, Faculty of Science,  
University of Ostrava  
`petr.saloun@osu.cz`

<sup>2</sup> Department of Computer Science, Faculty of Electrical Engineering and Computer  
Science, VSB Technical University Ostrava  
`zdenek.velart.feivsb.cz`

**Abstract.** This paper presents a proposal of navigation for the user, based on an existing ontology of the domain of the learning course. The navigation scheme of the course stands on interconnection of an ontology with existing learning material by the use of the keywords of the domain. On one side there is the user's needs and goals and on the other side are possibilities and limitation of existing systems.

## 1 Introduction

This paper presents our approach and results of research of navigation in a learning course. Navigation of the student in the information space is a very important task. We narrow our interest on course-based learning where the navigation is learning-oriented. We think that the student should be directly navigated to the desired information, which is seeking or which he wants to learn. According this we base our navigation scheme on an existing ontology of the domain, where relationship between knowledge is stored. The interconnection of the ontology with an existing course is on keywords of the domain.

As an integral part of the navigation a problem of user model initialization must be discussed – a problem of users first visit to the course. In our navigation scheme we discuss several possibilities how to deal with such a situation.

The basics of this idea were presented in [6]. Our problem domain are the courses of Programming in C++ and Electronical publishing. We taught the C++ course for two years (our results are presented in [5]) with about 400 student by the use of the adaptive hypermedia system AHA! [2]. The course is organized into twelve chapters, with 101 concepts. In our case every concept is realized by one XHTML file. The ontology we used we defined according the ISO specification of the C++ programming language and consists of about 300 elements.

## 2 Ontology-based navigation

The basic idea of the ontology-based navigation stands on the premise: *there is necessary to know all prerequisites to learn new information*. Every concept of

the domain (which can be defined and explained in paragraph, page or set of pages) is represents by the learning material of the domain.

The structure of the domain is captured in the domain ontology, where relationships and dependencies between elements are stored. Therefore the use of an ontology for the navigation in the course is preferable. Apart from the primary idea in [6] particular concept in the course is described by three distinct sets – *prerequisite set*, *outcome set* and *inout set*. These sets consists of the elements of the ontology. Element placed in *prerequisite set* represents prerequisites of the concept. Elements in *outcome set* represents what student can learn in the concept. Elements are placed into *inout set* when they should be at the same time in *prerequisite* and *outcome* sets. This indicates that the concept is dealing with “extended level” of the knowledge. Student’s knowledge learned from such a concept is on higher level than from a basic one.

## 2.1 Navigation

The information about visited concepts is stored in the student’s personal profile. The system tracks for the user which concept he visited and stores the elements from these concept. When the student learns a new concept all the elements form the *outcome set* are added to his *achieved knowledge set*, which is used in the navigation as is described later in the paper.

Presentation of concepts is based on the student’s actual goal, preferences and knowledge stored in his profile. Our model stands on the reference AHAM model, where user’s profile together with adaptation rules and engine are used for the actual presentation of the content.

Let us now discuss the navigation for the student in the course. The navigation is described by Figure 1 – used numbers corresponds with the description in the list:

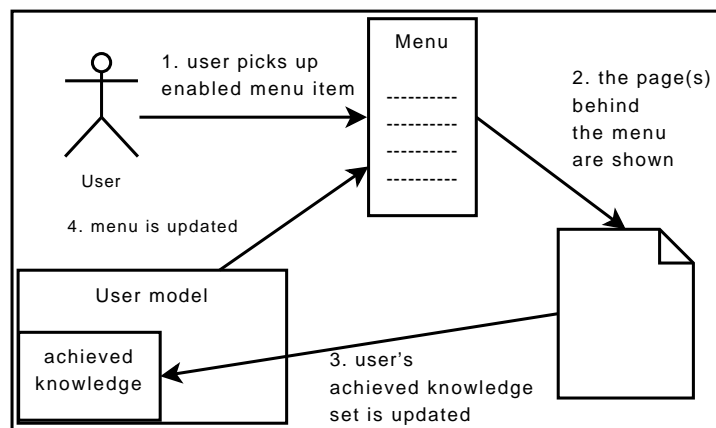


Fig. 1. Course navigation

1. Student chooses a concept from available concepts in the menu, which is presented to him.
2. According the chosen menu item the student is presented with learning material of the particular concept.
3. When student finishes the learning material, the system can test the students knowledge of the concept – this step is optional. The system then updates student’s *achieved knowledge* set with the elements from the *outcome* set of the concept.
4. Student’s menu is updated according his actual *achieved knowledge* set. Concepts which all elements from the *prerequisite set* are already in the achieved knowledge set are enabled. Navigation continues with the step 1.

The concepts presented to the student are ordered – on the first place, there are links to concepts for which the student has already all the necessary prerequisites or if achieves defined threshold value. Optionally other links can be presented – links to already visited concepts and not clickable links (only names) to concepts for which the student does not have all the necessary prerequisites.

For better navigation there is also possibility to construct a visual “map” of the course. In this map the current path of the student is highlighted. Concepts for which the students has all the necessary prerequisites are also rendered as clickable.

### 3 User model initialization

User model initialization represents a situation where the student visits the course for the first time. In such a moment the system have to set initial values for the newcomer.

- **Basic settings** – the student is presented only with concepts with empty prerequisite set or directly defined starting concepts of the course.
- **Knowledge test settings** – the student takes a “knowledge test” where the system tests the actual users knowledge about the concept.
- **Domain settings** – if the student comes from a similar course from the same domain (for example a student wants to take a C++ course afther finishing Java course) the system can reuse what the student have already learned. Mapping between the two ontologies is necessary in this case.
- **Course subset settings** – if the students comes to the system only with limited amount of goals, needs only to learn a limited amount of the information, the system can offer to the student only a subset of the concepts of the course, which lead to the desired goal. The system needs to make a subset of the concepts, starting from the goal and to choose concepts where the element is in outcome set. For the elements in the prerequisite set the system must find concepts, where the elements are in outcome set. This step is needed to do recursively for all found concepts.

## 4 Related works and conclusions

In this paper we presented our proposal of an ontology-based course navigation. Similar problem as ours can be found in other papers. One of these is [3], where authors present a visual education tool for efficient and effective learning. The prerequisite dependence relationship between concepts is based on extracted concept definitions. The authors assume existence of the concept definition database of the domain for the construction of the concept graph, which is then used for the navigation. Occurrences of one concept name in the others concept definition is used as a base for interconnection of the concepts. On the contrary our approach is based on the use of existing ontology of the domain. The difference between these two approaches is in what is used as the source for creating the relationships between concepts. Even if our approach is based on the ontology (in case of our C++ course build from ISO specification) the approach and course creation is transparent to the author of the course.

The authors in [1] presents a method aimed at creating content represented by an ontology and exporting such a content to a existing adaptive applications. Their approach is based on use of the core ontology, which is defined in the paper and is designed for adaptive application content modeling.

Our future research in this area will focus on extending the definition of the ontology-based course structure with situation where the course is enhanced with additional learning material or new concepts or on the other hand some material or concepts are removed from the course. We are also dealing with the automatization of the course creation process by the use of classification – some hints can be seen in [4]. We also prepared our C++ course with the described navigation together with a web-based system which is able to deliver such a course. The course will be offered to our students. After the run of the course we would like to compare our approach with data gathered in previous years, where the navigation in the course was mainly sequential with predefined fixed order of concepts.

## References

1. Bieliková M., Moravčík M. Modeling the Content of Adaptive Web-Based System Using an Ontology. *In proc. of 1<sup>st</sup> International Workshop on Semantic Media Adaptation and Personalization SMAP06*. IEEE Computer Society, pp 115-120.
2. De Bra P., Aerts A., Berden B., De Lange B., Rousseau B., Santic T., Smits D., Stash N. AHA! The Adaptive Hypermedia Architecture. *In Proceedings of the ACM Hypertext Conference*, Nottingham, UK, August 2003, pp. 81-84
3. Godbole S., Joshi S., Mehta S., Ramakrishnan G. Toward interactive learning by concept ordering. *In HT'07 Proceedings*. Manchester, UK, 2007, pp. 149-150.
4. Nepil M., Popelínský L.: Committee-Based Selective Sampling with Parameters Set by Meta-Learning. *In Proc. of ECML/PKDD Workshop on IDDM'02 Helsinki*
5. Šaloun P., Velart Z. Adaptive Hypermedia as a mean for learning programming. *In Workshop proceedings of the 6<sup>th</sup> ICWE 2006*. ACM Digital Library, 2006.
6. Velart Z., Šaloun P. Ontology Based Course Navigation. *In HT'07 Proceedings*. Manchester, UK, 2007, pp. 151-152.