*Carro, R.M., Pulido, E. and Rodríguez, P. (1999): An adaptive driving course based on HTML dynamic generation. Proceedings of the World Conference on the WWW and Internet WebNet'99 at Hawai, USA, October 1999. Association for the Advancement of Computing in Education (AACE), Vol. 1, 171-176. **Top Paper Award.***

An Adaptive Driving Course Based on HTML Dynamic Generation

Rosa María Carro, Estrella Pulido, Pilar Rodríguez
Escuela Técnica Superior de Informática
Universidad Autónoma de Madrid
Campus de Cantoblanco, 28049 Madrid, Spain
{Rosa.Carro, Estrella.Pulido, Pilar.Rodriguez@ii.uam.es

Abstract: In this paper we describe a new approach for developing adaptive Web based courses. These courses are defined by means of teaching tasks which correspond to basic knowledge units, and rules which describe how teaching tasks are divided into subtasks. Both tasks and rules are used at execution time to guide the students during their learning process by determining the set of achievable tasks to be presented to the student at every step. Adaptivity is implemented by presenting students with different HTML pages depending on their profile, their previous actions, and the active learning strategy. The HTML pages presented to the students are generated dynamically from general information about the type of media elements associated to each task and their layout. The whole approach is exemplified by means of a course on traffic signs.

1. Introduction

Since it was born, the World Wide Web has been used for learning because it allows students to have access to a great variety of information sources [Llamas et al. 1996][Vassileva 1997]. However, an Internet based learning system will be effective not only if it allows students to navigate between different pages but also if it helps students to achieve better their learning goals. It is true that a good design of the navigation space helps, but it is also necessary to provide more sophisticated mechanisms that modify the navigation alternatives by some sort of adaptation procedure.

There are important differences among students. Some of them are related to personal features as age, interests, preferences, etc and to their previous knowledge (if any) about the subject [Nill 1997][Eklund &Brusilovsky 1998]. There are others which have to do with their preferences for a specific learning strategy.

One of the desirable features of any Web based educational system is adaptivity, i.e. the ability to take into account all the above-mentioned features in order to customize the course contents as well as their presentation format [Brusilovsky 1997]. Different approaches have been proposed to incorporate adaptivity into hypermedia courses. One of them is the technique of sensitive links that are used to establish links between hyper-documents whose availability and contents change depending on the state of the teaching [Brusilovsky et al. 1996][Brusilovsky & Anderson 1998]. Other approaches use typed and weighted links to link concepts to documents and to other concepts [da Silva et al. 1998]. The student's knowledge of each concept is used to guide him/her towards the appropriate documents.

This paper describes an alternative approach to the construction of tutoring systems, which is based on the use of teaching tasks and rules in order to guide the learning process towards the desired goals. This approach has been implemented in the TANGOW system, Task-based Adaptive learNer Guidance On the Web [Carro et al. 1999a] [Carro et al. 1999b]. Courses designed by using TANGOW are dynamically adapted to each student by taking into account his/her profile and actions, as well as different teaching strategies.

2. Course design

When creating a new course, the designer must (1) make a conceptual decomposition of the course into tasks, and (2) establish the relation between tasks by means of rules. These steps will be described in the next two sections.

2.1 Teaching tasks

A teaching task is the basic unit in the learning process. A course is defined as a global task to be performed. Tasks can be decomposed into subtasks and students must perform some or all the subtasks to get the main task performed. A task can be defined by giving value to some general attributes such as its name, description, content type, composition type, and ending requirements. Content type distinguishes between theoretical, practical and example tasks. According to the composition type, tasks may be atomic or composed. Finally, ending requirements are implemented by means of functions which decide, at runtime, whether tasks have been completed. This decision is based on parameters directly related to student actions, in the case of atomic tasks, or on subtask finalisation, if the task is composed.

Teaching tasks have other specific attributes which include a list of media elements (text, images, videos, applets, sounds, animations, ...) which will be used for HTML page generation. The way in which these elements will be combined to construct the final pages is described by means of a description language which specifies their relative positions. If any of the media elements corresponds to an exercise, the correct answer is also given. This can be compared to the AHA system [de Bra & Calvi 1998] where filters for content fragments are encoded by means of conditional sentences included as comments in HTML pages. The main difference with our approach is that we create the HTML pages by linking media elements, whereas in the AHA system the pages are already created and it is decided at runtime which portions of them are shown to the student.

A course on traffic signs has been developed for demonstration and can be tested at http://helena.ii.uam.es/html/courses.html. The description of two tasks belonging to this course is shown in [Fig. 1]. The 'Circular signs' task is an example of a theoretical atomic task. Its final requirements associated function ('f_teo') will receive several input parameters such as the number of pages visited so far by the student ('pag_visited') and the total number of pages related to the task ('tot_pag'). The media elements used to generate the HTML pages related to this task appear as values for the HTML field. They will appear in a sequence up-down (M1).

A different example is the 'Traffic Agents: Exercises' task which corresponds to a practical task which contains exercises related to traffic agents. It is an atomic task. The input parameters of its final requirements method ('f_prac1') are the number of exercises done by the student ('exer_done'), the number of exercises correctly solved ('exer_ok'), and the total number of exercises that are available ('tot_exer'). The media elements as well as the correct answer for each exercise are specified as values for the HTML field.

NAME = CIRCULAR_SIGNS TYPE = T ATOMIC = Y DESCRIPTION = Description of circular signs END_METH = F_TEO PARAMS = pag_visited tot_pag HTML = CIRCULARES M1 STOP M1 C_PROHI M1 E_PROHI M1 EP_VEHI M1 EPV_SIDE M1	NAME = TRAFFIC_AGENTS_EXERCISES TYPE = P ATOMIC = Y DESCRIPTION = Exercises about signals by traffic agents END_METH = F_PRAC1 PARAMS = exer_ok exer_done tot_exer HTML = eA1 a M1 eA2 b M1 eA3 c M1 eA4 b M1 eA5 a M1

Figure 1: Two different task descriptions

2.2. Rules

Task decomposition is represented by means of rules. Each rule is given a name and includes information about the task to be divided ('LHS'), the list of subtasks that are part of the task ('RHS') and a keyword corresponding to the sequencing of the subtasks ('SEQUENCING'). XOR indicates that only one of the subtasks must be performed, OR means that at least one of the subtasks must be performed, AND indicates that all the subtasks must be performed in the order they appear in the list, and ANY indicates that all the subtasks must be performed, but that they can be performed in any order. In addition, the course designer can specify a precondition that has to be satisfied for initiating the task ('ACT_CONDITION') and the parameters ('PARAMS') used for evaluating this precondition (which may be taken from other tasks results). The last rule attribute ('CALC_PARAMS') indicates how parameter values for tasks in the RHS of the rule are propagated to the task in the RHS.

The description of two rules in the driving course is shown in [Fig. 2]. Rule R0 indicates that the 'Types' task is decomposed into three subtasks: 'Circumstantial Signs', 'Vertical Signs' and 'Traffic Agents'. The student must perform the three subtasks, but (s)he can do it in any order. There is no precondition for the rule to be activated, and the propagation of parameters from the RHS to the LHS of the rule is done by adding the values of homonym parameters. In rule R1, the precondition 'c_4' indicates that the rule will be active only if the execution of method 'c_4' returns true when it is given the value for the 'exer_ok' parameter from 'Vertical_Signs' as input.

```
NAME = R0
SEQUENCING =  ANY
LHS =     Types
RHS =     Circumstantial_Signs
          Vertical_Signs
          Traffic_Agents
CALC_PARAMS= time_in    msum3   time_in   Circumstantial_Signs
                                time_in   Vertical_Signs
                                time_in   Traffic_Agents [...]
```

```
NAME = R1
SEQUENCING= AND
LHS =     Circumstantial_Signs
RHS =     Circumstantial_Signs_Theory
          Circumstantial_Signs_Exercises
ACT_CONDITION = c-4
PARAM =  exer_ok       Vertical_Signs

CALC_PARAMS = [...]
```

Figure 2: Two different rule descriptions

Details about all the tasks and rules of the driving course can be found at the following address http://www.ii.uam.es/~rcarro/Webnet99/TaskTree.html, where each cell corresponds to a task. Those rules in which the task appear at the LHS of the rule (or the RHS) are written above (or below) the task name.

3. Why do we call it "adaptive"?

The set of tasks that the student is offered to tackle at a given moment can vary depending on the student profile, the current teaching strategy, actions performed by the student when tackling specific tasks, and the course design.

3.1. Student profile

The student profile includes personal features such as the student age and native language, the relation between the student and the studied subject, etc. The first time that a student enters a course, (s)he is presented with a test where some personal data and preferences fields must be filled. [Fig. 3] shows the test as filled by two different students.

A task may be defined in different ways depending on one or more of these features. As an example, in the traffic signs course, the "Traffic Signs" task is defined by means of two different rules:

(RR1) Traffic Signs -> Types, Priority

(RR2) Traffic Signs -> Types, Priority_Theory

The "Priority" task is defined by rule RP.

(RP) Priority -> Priority_Theory, Priority_Examples

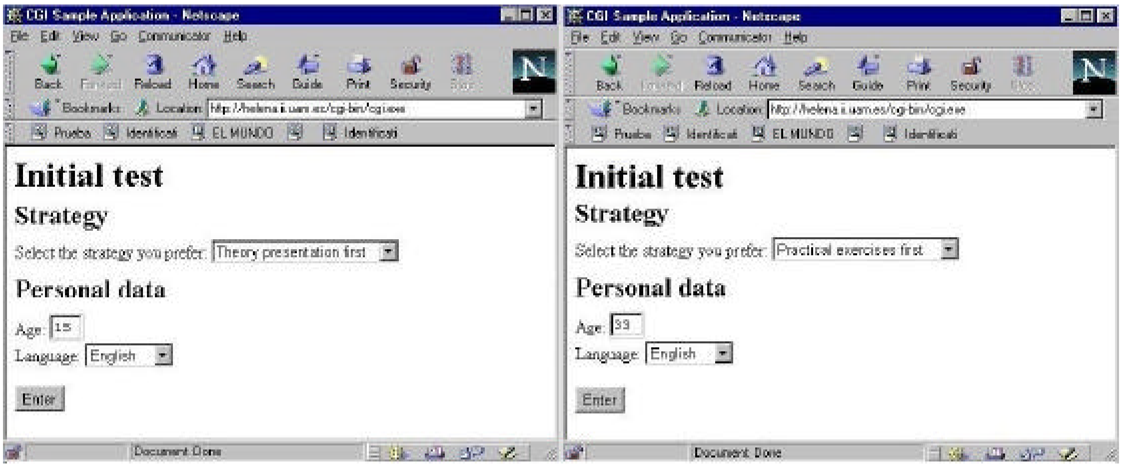

Figure 3: Different student profiles

Both rules RR1 and RR2 have an activation condition which depends on the student's age. This means that the "Signs" task may be achieved in two different ways depending on the student's age. Basically, if the student is younger than 18, (s)he will be presented with examples about the priority of signs. Otherwise, (s)he will study only theoretical aspects on the same subject (see [Fig. 4]).

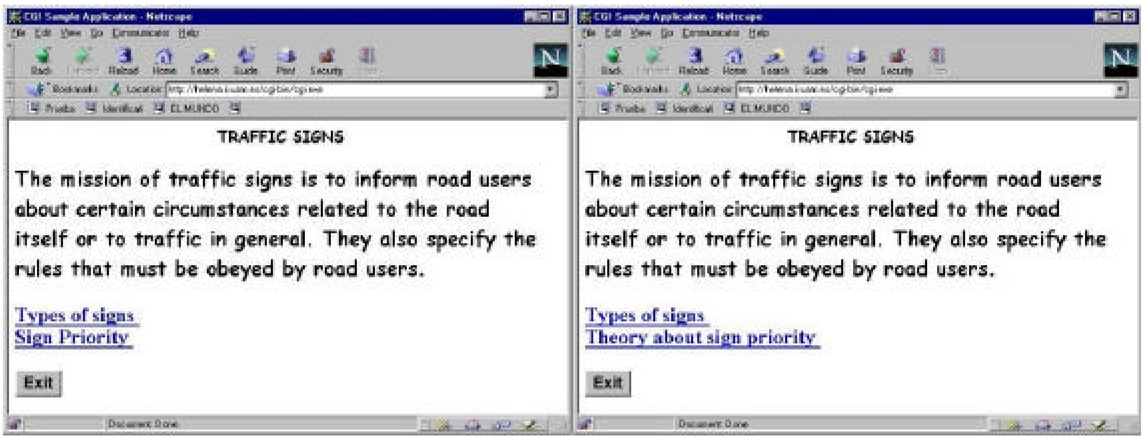

Figure 4: Adaptivity based on student profiles

3.2. Teaching strategy

Teaching strategies have an effect on the order in which subtasks are performed. For example, if the "theoretical before practical" strategy is being used, the student will be presented with theoretical tasks first, and only after they have been performed, (s)he will be able to tackle practical tasks. Similarly, for the "practical before theoretical" teaching strategy (see [Fig. 3] and [Fig. 5]).

3.3. Student actions

A task may be included in the set of achievable tasks at a given moment depending on actions previously performed by the student when tackling a different task. For example, we may decide that a student can only learn about circumstantial signs if (s)he has successfully solved a number of exercises related to vertical signs (see [Fig. 6]). Note that, in contrast to figure 6a, in figure 6b, the "Description of vertical signs" task is not in the choice set. The reason for this is that the task has already been performed. The "Circumstantial signs" task is accessible because the student has solved correctly the exercises associated to the "Description of vertical signs" task. Other student actions may be related to the time a student has spent learning a task or the number of pages he/she has visited.

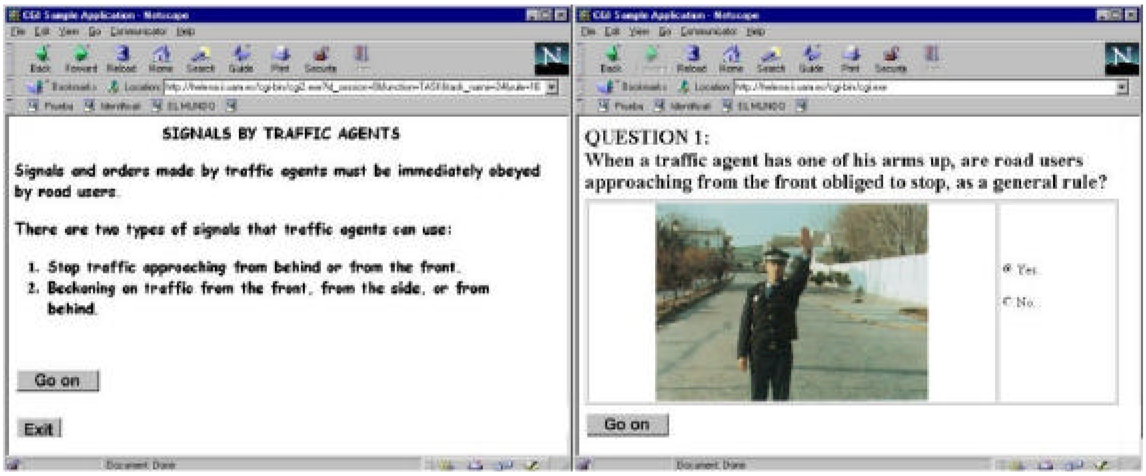

Figure 5: Adaptivity based on the learning strategy

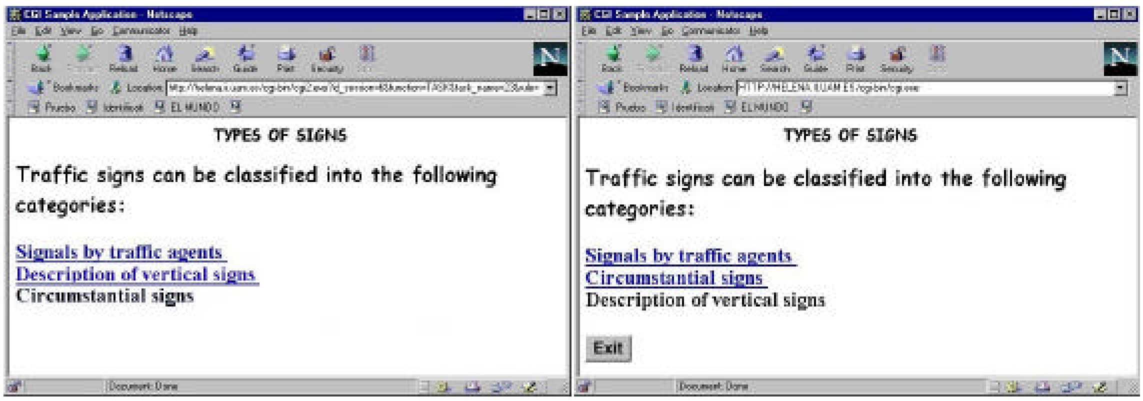

Figure 6a and 6b: Adaptivity based on student actions

3.4. Course design

The sequencing specified by the course designer has a great influence in the set of tasks that are offered to the student at a given moment. Initially, when the active task appears in the LHS of a rule with ANY, XOR, or OR sequencing, all the achievable subtasks in the RHS are offered to the student. This is the case for some of the preceding figures. However, if the rule sequencing is AND, only the first subtask in the RHS of the appropriate type (according to the current strategy) and that has not been performed will be made available.

In subsequent steps, three different cases can appear. If the rule sequencing is ANY or OR, the remaining subtasks will be offered to the student by presenting him with a menu page. If the sequencing is ANY and there is only one task left, then the system takes you directly to the information associated with it. In the case of OR sequencing, the student has the chance of exploring new paths before continuing with his (her) learning itinerary. If the rule sequencing is AND, the student will be offered a single subtask which will be chosen from the RHS of the rule depending on the active teaching strategy. Note that if the rule sequencing is XOR, there is no need to offer additional tasks to the student.

4. Work in progress

The mechanism of teaching tasks presented in this paper adds adaptability to interactive courses, by controlling events about the keystones reached by students at different stages of their learning process.

Work is continuing on the presented system with the immediate goals of (1) extending it so that the teaching strategy may be modified at runtime, and (2) allowing the students to revisit tasks already performed by including frames with task lists in every dynamically generated HTML page.

References

de Bra, P. & Calvi, L. (1998). AHA: A Generic Adaptive Hypermedia System, *Second Workshop on Adaptive Hypertext and Hypermedia*, pp. 5-11. Ninth ACM Conference on Hypertext and Hypermedia, Pittsburgh, USA, June 20-24, 1998.

Brusilovsky, P. (1997). Distributed intelligent tutoring on the Web, *8th World Conference of the AIED Society*, Kobe, Japan, August 1997, http://www.contrib.andrew.cmu.edu/~plb/AIED97_workshop/Brusilovsky/Brusilovsky.html

Brusilovsky, P. & Anderson, J. (1998). ACT-R electronic bookshelf: An adaptive system for learning cognitive psychology on the Web, *WebNet 98 World Conference of the WWW, Internet & Intranet*, pp. 92-97, Orlando, Florida, November 7-12, 1998.

Brusilovsky, P., Schwarz, E. & Weber, G. (1996). A tool for Developing Adaptive Electronic Textbooks on WWW, *WebNet-96* of the AACE, http://www.contrib.andrew.cmu.edu/~plb/WebNet96.html.

Carro, R.M., Pulido, E. & Rodríguez, P. (1999a) TANGOW: Task-based Adaptive learNer Guidance On the Web, *Second Workshop on Adaptive Systems and User Modeling on the Web* at the Eighth International World Wide Web Conference. Toronto, Canada, May 11-14, 1999.

Carro, R.M., Moriyón, R., Pulido, E. & Rodríguez, P. (1999b) Teaching Tasks in an Adaptive Learning Environment, *8th International Conference on Human-Computer Interaction*, HCI International '99, Munich, Germany, August 22-27, 1999, forthcoming.

Eklund, J. & Brusilovsky, P. (1998). The Value of Adaptivity in Hypermedia Learning Environments: A short Review of Empirical Evidence, *Second Workshop on Adaptive Hypertext and Hypermedia*, pp. 13-19. Ninth ACM Conference on Hypertext and Hypermedia, Pittsburgh, USA, June 20-24, 1998.

Llamas, M., Fernández, M.J., Gil, A., Rodríguez, R. & Suárez, A. (1996). A Cost-effective Approach to WWW Education and Training, Proceedings of the 3rd International Workshop on Protocols for Multimedia Systems (PROMS'96), October 1996.

Nill, A. (1997), *Providing Useable and Useful Information by Adaptivity and Adaptability*, GMD- German National Research Center for Information Technology, Sankt Augustin, Germany, http://zeus.gmd.de/~nill/flexht97.html

da Silva, D.P, Van Durm, R., Duval, E. & Olivi, H. (1998). Concepts and documents for adaptive educational hypermedia: a model and a prototype, *Second Workshop on Adaptive Hypertext and Hypermedia*, pp. 35-43. Ninth ACM Conference on Hypertext and Hypermedia, Pittsburgh, USA, June 20-24, 1998.

Vassileva, J. (1997). Dynamic Courseware Generation on the WWW, *8th World Conference of the AIED Society*, Kobe, Japan, August 1997. http://www.contrib.andrew.cmu.edu/~plb/AIED97_workshop/Vassileva/Vassileva.html

Acknowledgements

This paper has been sponsored by the Spanish Interdepartmental Commission of Science and Technology (CICYT), project number TEL97-0306.