EVALUATION OF MOT, AN AHS AUTHORING TOOL: URD CHECKLIST AND A SPECIAL EVALUATION CLASS

Alexandra I. Cristea and Arnout de Mooij
Information Systems Department, Faculty of Computer Science & Mathematics, Eindhoven University of Technology, Postbus 513, 5600 MB Eindhoven
The Netherlands
+31-40-247 4350
a.i.cristea@tue.nl

ABSTRACT

In this paper we present some past and future evaluation goals, methods and results of an adaptive hypermedia system authoring tool, MOT, developed at the Eindhoven University of Technology (TU/e). Our first evaluation is a simple URD evaluation. The second planned evaluation is more complex, because it involves at the same time a test bed setting, as well as a classroom setting, and is due to the latter required to have educational value as such.

KEY WORDS
Adaptive hypermedia, remote authoring, evaluation

1. INTRODUCTION

Adaptive hypermedia (AH) is a promising new field that permits simple on-line user modeling and customization [1]. AH has been traditionally applied in education [2], but is now gradually moving towards other fields, such as e-commerce.

However, in the AH field, evaluation is a serious and difficult problem, signaled as missing and needing more focus by many researchers (most recently at the CEDEFOP AH workshop, Thessaloniki, Greece, 07/03/03).

MOT is an adaptive hypermedia adaptive authoring system built at the Eindhoven University of Technology (TU/e) for on-line adaptive course production. MOT is based on two frameworks:

1. an adaptive authoring framework for adaptive hypermedia systems (AHS), LAOS [3] based on AHAM [4], and
2. will in future integrate LAG [5], a layered view on adaptation functionality. The MOT system and architecture have been described in [1].

This paper focuses on the evaluation of the current version of MOT, from different points of view: firstly, in section 2, we are going to show the user specifications (requirements) of MOT, and in section 3 the extent to which these were fulfilled. These evaluations have been done last year. This year, we are preparing to use an Adaptive Systems class for a group of 20 user-interface (USI) post-graduate students at the TU/e to implement a more thorough testing of MOT, from an authoring point of view, with special attention to the interaction between system and its users (USI being the specialty the students are majoring in). These evaluation settings, as well as the implementation details and expected results are shown in section 4. Finally (section 5) we draw some conclusions.

2. MOT URD

We constructed the following user requirements (URD) to describe our goals in more details (see tables 1-3).

Table 1. General URD

GEN01: The system will be used by a teacher to construct adaptive hypermedia courses. The system will consist of two parts: a tool for manipulating concept maps and a tool for constructing lessons based on concept maps.
GEN02: The tools will be integrated in the existing MOT system. Hence, they will be web applications having an interface common to MOT and will run on the same platforms as MOT. *MOT runs under Apache. The scripts are in Perl. Java applets are also used, as well as some Javascript in the help pages. So all these have to be installed on the machine where MOT is to function. MOT works with Apache 1.3.20; ActiveState ActivePerl Build 626 v5.6.1; Java 1.4.0.*
GEN03: Information regarding users, concept maps and lessons is stored in a (MySQL or other) database (instead of text files).

MOT was built, as said, based on the LAOS [3] framework, which inherits its concept descriptions from an older version of the system, MyET [6].

In the following tables, more detailed descriptions of implementation requirements follow. Table 2 describes the concept maps and its relations, as well as its components (attributes). For a formal definition of the concepts used, please refer to [3].

Table 2. Creating and manipulating concept maps

CM01: A concept map has a hierarchical structure. The user can add and remove concepts to/from this hierarchy.
CM02: The user can add and remove attributes to/from concepts

in the concept map. There is a set of standard attributes that each concept has (*title, text, keywords, introduction, etc.*).
CM03: The user can re-order the concept hierarchy. That is, he can promote or demote a concept (and its sub-concepts) in the hierarchy.
CM04.a: The user can re-use concepts and attributes from other concept maps in the concept map he is working on. He will have a restricted view on this external concept map allowing him only to view and optionally to copy concepts or attributes. **CM04.b:** 'Relatedness' relations between concept attributes should be able to be added. They can be given a weight, depending on *how much* the attributes have in common. **CM04.c:** Given a certain concept map, correspondence weights between the attributes can be calculated automatically by the system. When a calculated weight exceeds an adjustable threshold, a 'relatedness' relation between two attributes is suggested by the system. The teacher can either accept or deny this relation. Also he can add a descriptive label to the relation.
CM05: The user can add (and remove) so-called 'relatedness' relations between concept attributes. These can be given a weight and a label.
CM06: The system will be able to suggest 'relatedness' relations between concept attributes based on correspondence calculations. The user can either accept or deny these relations. Some different views of the concepts should be possible in order for the teacher to understand the (manual/ automatic) changes made.
CM07: The view(s) the user has on the concept map will be such that - manipulations as described in CM01 to CM05 can be performed in an easy way, - the user will be able to overview the whole map, - the user will be able to understand the (manual/ automatic) changes made.

After adding 'relatedness' relations to a concept map, a teacher can start to compose lessons using this map.

The concept map can be considered equivalent to a *book* that the *presentation* (lesson) is based upon, while the lesson map is equivalent to a subset of the information contain in the book (or in *more* books on the subject).

This subset of information is actually a constrained version of the information, which is to be used in the presentation. To be useful, this information has to have a final goal. This is why, in the more general LAOS model [3], this lesson map is contained in the so-called '*goal and constraints layer*'.

In table 3, the user requirements for the lesson components and manipulation are defined. For a more formal definition of the lesson concept used here, please refer to [3].

Table 3. Creating and manipulating lessons

L01: A lesson has a hierarchical structure. The user can add or remove sub-lessons to/from this hierarchy. Sub-lessons of a (sub-)lesson can be weighted alternatives ('OR') or lesson-parts that will all be presented to the student in a given order ('AND').
L02: A (sub-)lesson corresponds to one or more attributes of a concept from a concept map. The user can select these attributes from a concept map.
L03: The system will suggest links between sub-lessons based on relatedness or hierarchical structure of the corresponding concept attributes. The user can either accept or deny these links. Also the user can add his own links between sub-lessons.
L04: The user can re-order the lesson hierarchy. That is, he can promote or demote a sub-lesson (and its sub-lesson) in the hierarchy and he can change the order of sub-lessons.
L05: The teacher can either create a branching of the course structure (AND/OR connection) to connect his course to the course of somebody else, or copy and paste parts of that other course into his own.

In our structure, (sub-)lessons correspond to concept attributes. The latter can be related to each other because, e.g., they belong to the same father-concept (superior) or they are related in some other way (e.g., their contents has commonalities). These last relations don't necessarily appear in the concept hierarchy and will be called 'relatedness' relations [1]. Moreover, they can be at concept or concept attribute level. For simplification, we will only treat (concept) attribute level relatedness relation. By user we mean here the person who is going to use the adaptive course design system, e.g., the teacher or course designer.

3. EVALUATION OF URD COMPLETION

This section lists one by one the results of the evaluation of the general user requirements, as well as the concept-map and lesson-map related URDs (Tables 4-6).

Table 4. General

GEN01: This requirement is met.
GEN02: This requirement is not yet met. The system created replaces a large part of the MOT system, but the part of user log-in and some multi-user security issues (users should not be able to change each others lessons, etc.) which are available in MOT, do not exist in the new system. Because the new system uses a database to store information about the user and MOT doesn't, a part of MOT had to be rewritten to link it to the new system.
GEN03: This requirement is met.

Table 5. Creating and manipulating concept maps

CM01: This requirement is met.
CM02: This requirement is met. The set of standard attributes can be changed per concept map.
CM03: This requirement is met. Concepts in a concept map can be moved by using cut and paste actions.
CM04: This requirement is met. Moreover, the user can create links to other concept maps. When creating a link, the data from the external map is not actually copied, only a pointer is set to a given concept in this map.
CM05: This requirement is met.
CM06: This requirement is met. However, the way in which the automatic correspondence calculation works can be improved. It is desirable to use some kind of artificial intelligence in this part, possibly also using some adaptation to the teacher who is creating the map. This is the part that mainly determines the

power and usefulness of the system.
CM07: This requirement is partially met. To be able to overview a large concept map, the user will have to use either scroll bars or a collapsible list view. This second option is to slow to be useful for very large maps. Whether the user interface is sufficiently transparent is a point of discussion.

Table 6. Creating and manipulating lessons

L01: This requirement is met. If there are alternative sub-lessons, they can also be given a label.
L02: This requirement is partially met. A sub-lesson can only correspond to one concept attribute. However, a sub-lesson can have other sub-lessons, which can each correspond to a concept attribute on their turn.
L03: This requirement is partially met. While adding a sub-lesson corresponding to a certain concept attribute, the user gets to see the relatedness relations of this concept, but the system won't give an extra warning or hint. So the user should be alert about relatedness relations himself.
L04: This requirement is met. Sub-lessons in a lesson can be moved by using cut and paste actions.
L05: This requirement is met.

4. SECOND EVALUATION

The main original points of MOT are:

- the usage of *concept maps* and *semantic atomic unit* ideas to represent the domain model in a flexible, well-connected way;
- the usage of *lesson maps* to represent the second stage of presentation development, by constraining concept maps and defining goal structures;
- the *teacher adaptive features* inserted for automatic concept binding (with possible extension of more adaptive authoring facilities, as defined by LAOS [3]).

The above represent the '*hot issues*' we would like to test.

Our second evaluation implies real system usage by a different group of people than the initial designers. In this way, we want to get some feedback on:

1. the extent to which our goals were realized with this system from an outsiders' perspective
2. the usability of the system.

To this end, we will give MOT for usage to a class of 20 students of an 'Adaptive Systems' course. In the following, the details of this experiment are described.

4. 1. THE STUDENTS

The group is formed of twenty students that follow a post-graduate two-year study to specialize in user interfaces and user-system interaction (USI). The study requirements of this module have two interesting special aspects, which make it appropriate for our test purposes:

a) it is very practice-oriented (hands-on experiences)

b) it is done via intensive two-week courses

The students themselves come from different backgrounds, and have different nationalities, genders, previous degrees, as listed in table 7.

Table 7. Student distribution USI

No.	M/F	Country	Degree	Education
1	F	U.S.A.	B.A.	Fine Arts (Graphic Design)
2	F	Netherlands	Drs.	Economische psychologie
3	F	India	M.A.	English Literature & Economics
4	M	Belarus	M.E.	Radiophysics
5	F	Indonesia	M.Sc.	Artificial Intelligence
6	M	Israel	B.E.	Industrial Design
7	F	Netherlands	M.E.	Ind.Engineering & Man.Science
8	M	Greece	M.Sc.	Comp.Eng & Informatics
9	F	Ukraine	M.E.	Organisat. Sc.& Econ.
10	M	Belarus	M.Sc.	Mathematics
11	M	Chili	M.Sc.	Design Visual Communication
12	M.	Belgie	Lic.	Integral Product Development
13	M	Netherlands	M.Sc.	Mechanical Enginering
14	F	Netherlands	drs.	Cognitive Psychology
15	M	Belarus	M.E.	Physics
16	F	Netherlands	drs.	Cognitive Ergonomics
17	M	Netherlands	M.Sc.	Physics
18	M	Netherlands	drs.	Neurolinguistiek
19	M	Netherlands	M.Sc.	Computer Science
20	F	China	B.Sc.	Comp.Math. & Appl.software

4. 2. THE EXPERIMENT IMPLEMENTATION

The whole course is a two-week intensive course with a great accent on practical work and hands-on experience, ideal for our goals.

The students will be exposed to the following teaching and testing procedure.

1. Background knowledge of adaptive systems, with focus on adaptive hypermedia
2. MOT system theoretical framework (LAOS, LAG)
3. Installing, experimenting and finally doing an assignment with MOT, as well as a system evaluation
4. Evaluation of assignments and student grading

Table 8. Details of the experimental settings

Time	April 14-18 2003 (week 16); May 5-9 2003 (week 19)
Form	- Lecture (20%-25%), - Discussion (question hours5%-10%) and - Project work (~60%-70%)
Evaluation	- Project work evaluation (40%) - deliverable, - Project report (20%)- deliverable, - Project presentation (20%) - Peer evaluation (veto) – deliverable - Course evaluation (20%) – deliverable

Based on previous courses	- Database Technology (USI), - Hypermedia (USI), - Mathematics, - Programming
Also desirable	Mathematics at basis levels (2Y930, 2Y940, 2Y950), some probability calculus and statistics, e.g. (2S970), basic programming skills
Starting situation	- Number of students = 20 - Students = Masters level or over - Orientation of students = practice and experience building - Students' background = various Master level studies, from computer science to philosophy - Connection to students' daily life = students study USI – so adaptive systems and user modeling from user system's interface p.o.v.
Course Goals	At the end of this course students should know the basic principles of adaptive systems, know at least 3 application fields for adaptive systems, should be able to list the requirements for the design of an own adaptive hypermedia system and should be able to either write the appropriately chosen content for such a system, or do the implementation of (parts of) such a system.
Central principle in course:	The course will discuss various aspects of complex adaptive systems, with a special emphasis on their many applications in information technology, computer science and engineering and education.

From an educational point of view, the lecture component of the course module has the following main goals:
- Attract bright graduate students to these exciting research topics, which are well reflected in the research interests of the faculty of several UI departments, both within and outside the USI.
- Demonstrate some of the applications of complex adaptive systems in the real world.
- Expose students to an outstanding invited speaker of international recognition, from research and education institutions, representing some of the research areas that make up the general themes of complex adaptive systems.

Educationally speaking, the project work component of the course module has the following main goals:
- Give the students a hands-on experience on actually building (parts of) an adaptive system.
- Let the students develop structural thinking for design and implementation.
- Let the students develop and familiarize themselves with collaboration strategies within a group during the group project work.

Table 9. Detailed program

Day of the week & date	**Hours**	**Activity**
Monday 14-04-2003	9:30 – 16:30	Introduction Lecture on General Adaptive Systems Lecture User Modeling
Tuesday 15-04-2003	9:30 - 16:30	Lecture on Methods: - Data representation - Data manipulation
Wednesday 16-04-2003	9:30 – 12:00 13:30- 16:30	MOT presentation Division into groups Selection of themes and sub-themes (contents discussion and approval) Project work start
Thursday 17-04-2003		Excursion to Philips Research Laboratories (Natlab)
Friday 18-04-2003	9:30 - 16:00	Project work Question hour
Monday 5-05-2003	9:30 – 12:00 13:30- 14:30 14:30- 16:30	Invited speaker: - Adaptive Systems: A different perspective Question hour (problem discussion) Midway peer evaluation (oral + discussion of problem cases) Project work
Tuesday 6-05-2003	9:30 – 12:00 13:30- 16:30	Project work Midway demonstration of project results (+ progress discussion)
Wednesday 7-05-2003	9:30 - 16:30	Project work
Thursday 8-05-2003	9:30 – 12:00 13:30- 14:30 14:30- 16:30	Project work Question hour Project work
Friday 9-05-2003	9:30 – 12:00 13:30- 16:30	Handing in final peer evaluation (oral / e-mail / on-line[1]) Handing in reports Demonstration of project results Handing in project results (floppy or e-mail) Projects course evaluation presentation (on-line questionnaire)
End of 05-2003		Notification of grades at secretarial office and on-line

The treated material contains, globally:
- Adaptive systems
- User modeling:
    - Definition and Aims

[1] On-line means accessible from my homepage.

  - History
  - Application areas
  - Adaptation goals
  - Techniques
  - Newer developments, Future trends
- Data representation techniques for adaptation:
  - Concept Maps
  - Rule systems
- Adaptation techniques

The required literature is given by:
- Powerpoint presentation files of lectures will be made available on-line, after the lectures.
- MOT will be also available on-line.

The students will be given projects to fulfill, as listed in table 9 and 8. The project themes include but are not limited to:

1. Writing a course on a chosen subject with MOT (maximal grade: 9-10; maximal group size: 2-3);
2. Writing a new interface module for MOT (maximal grade: 10; maximal group size: 3);
3. Writing a course on a chosen subject with MOT and the new interface module to display its facilities (maximal grade: 10; maximal group size: 5-6);
4. Reading, analyzing and discussing Adaptive Systems' related academic papers and evaluating MOT based in them (maximal grade: 7-8; maximal group size: 2);
5. Evaluating MOT from several perspectives (all students have to be involved in this, the perspective being given by their interaction with the system, as established by the main project).

4. 3. THE EVALUATION GOALS

In the beginning of section 4 we mentioned the main direction of testing we intend to pursue with this experiment. Here we are going into more details, showing the specific aspects that have to be evaluated:

Table 10. Goal point of view evaluation

Collaboration more authors collaborating at a course;	- what are the problems? Suggestions for solving them? How did you try and solve them? What are the good points? - Comparison - with collaboration (two or more working at one course together: experimental group) and without collaboration (one person only, with a smaller task: control group); the satisfaction degree should be measured, as well as the result evaluated.
Completeness looking at given goal (e.g., URD), LAOS two layer	- what is the perceived percentage of completeness? What is the expressivity? What is the (preceived) connectivity degree? Should there be more connections, or less? What would this possible extra connections be? What should be deleted (e.g., is superfluous)?
Adaptivity	- How much adaptivity to the design goal is perceived?
Design range	- How much more can be achieved in this way as compared to the linear model?

Table 11. USI point of view evaluation

Ease of use	- information display (e.g., on screen at one time), information order, distance of search (depth); - minor issues: color scheme, ease of access, ease of installation
Robustness	- parallelism (data overlap), security, recovery
Complexity of interface	- analysis of possible reduction of this complexity; analysis of MOT as an implementation of the LAOS model: reduction of complexity: 'you are here in the LAOS model' for teachers.

4. 4. THE EVALUATION METHODS

For evaluating the system, we want the students to become authors on a subject they are knowledgeable. As they are graduate students, we consider them capable of knowing what is needed to present some material that is known to them. They can choose, for instance, a previous interesting course from their module that they want to describe. Therefore, their assignments imply writing a course of their choice with MOT, in groups of 2-4.

In order for the students to evaluate the system, we are going to create some questionnaires on the different aspects we want to study, as listed in section 4.3 and table 10. These questions will be mapped on a Likert-type scale.

For this specific setting, it is not possible to compare MOT to a different version of itself without the respective functionality. Therefore, in order to have a control group and an experimental group, we will let another group of students make their own questionnaires that they want answered, just asking them to group them in design goal completion and user interface completion, as well as asking them to create a given, predefined, number of questions. In this way, we hope to obtain two things at the same time:

1. to collect new questions we might have overlooked;
2. to indirectly check if the goals of the system are clear to the potential author: a goal that would not be mentioned is clearly not expressed sufficiently.

Finally, we intend to have some discussion with the students about their overall impressions, the actual meaning of their questions and their answers, in order to eliminate any misunderstandings.

Here we need to mention that the students will be told from the beginning that their negative evaluation of the system will not affect their grades, but that the thoroughness and constructiveness of their answers will.

The students' MOT system evaluation results will be analyzed for:

- their means,
- standard deviation,
- significance testing,
- sensitivity analysis,
- content analysis of open-ended responses,
- cross-tabulation between different groups (Chi-square – for statistical significance, and phi and Cramers V. – for association strength),
- multivariate analysis: ANOVA, linear and non-linear regression, logistic response models, categorical methods.

The analysis will be made with the SPSS statistical software, the trial version.

Finally, the student course creation results themselves will be analyzed, in order to reflect on the:

- time necessary to familiarize oneself with MOT
- perceived flexibility of MOT
- perceived expression freedom of MOT
- time necessary to create some courseware with MOT, etc.

4.5. EXPECTED RESULTS

Our system MOT is at its α-version, and therefore we expect that the USI evaluation will be quite critical, thereby giving us much insight about necessary and possible further developments.

On the other hand, the goals we have set are new and challenging, so we hope to score some points with this part of the system.

We expect however that reality will not exactly correspond to the theory, and that the group of young authors will be confronted with many different issues that are not covered by the current system.

However, we would like to see at least our framework LAOS confirmed by these experiments.

MOT only covers at the moment two fifths of the LAOS framework, so that the present testing will be able to give us useful insight about the next implementation steps necessary to fulfill the LAOS requirements.

5. CONCLUSION

In this paper we have presented two evaluations of MOT, an adaptive hypermedia authoring system under development at the Eindhoven University of Technology:

1. a URD evaluation, that considered the goals of MOT broken into objectives and requirements
2. a real-usage evaluation within a class on 'Adaptive Systems' of a USI profile group of graduate students.

The first evaluation has already been performed, so results were shown. The URD requirements were simple, and therefore most of them are completed. Some of the reported uncompleted requirements (as, e.g., GEN02) are actually in a better implementation state at the moment, as the next generation of students to be involved in the project already started working on those issues (β-version).

The real challenge is represented by the second evaluation with the actual target users of our system, the authors. This second evaluation is only under preparation now, so we can only make prediction about the expected results. At the conference we plan to show the real results, which will be obtained after the completion of the class that ends in April 2003.

6. ACKNOWLEDGEMENT

This research is linked to the European Community Socrates Minerva project "Adaptivity and adaptability in ODL based on ICT" (project reference number 101144-CP-1-2002-NL-MINERVA-MPP).

REFERENCES

[1] P. Brusilovsky, Adaptive hypermedia, *User Modeling and User Adapted Interaction*, Ten Year Anniversary Issue (Alfred Kobsa, ed.) 11 (1/2), 2002, 87-110.
[2] P. De Bra, Adaptive Educational Hypermedia on the Web. *Communications of the ACM*, 45(5), pp. 60-61, '02.
[3] A.I. Cristea, LAOS: Layered WWW AHS Authoring Model and their corresponding Algebraic Operators, *WWW'03*, Budapest, Hungary, ACM, IW3C2 (to appear).
[4] H. Wu, A Reference Architecture for Adaptive Hypermedia Applications, *doctoral thesis*, Eindhoven University of Technology, The Netherlands, ISBN 90-386-0572-2.
[5] A.I. Cristea, L.Calvi, The three Layers of Adaptation Granularity, *UM'03*, Pittsburgh, US, Springer (to appear).
[6] A.I.Cristea, T. Okamoto, & S. Belkada,: Concept Mapping for Subject Linking in a WWW Authoring Tool: MyEnglishTeacher: Teachers' Site, *ANNIE'00*, Eds. Drs. Dagli et al, ASME Press.
[7] A.I. Cristea, A. de Mooij, Adaptive Course Authoring: My Online Teacher, *ICT'03*, Papeete, French Polynesia, IEEE, IEE.