Empirical Analysis for the Design of a WWW Knowledge-based Authoring Tool

Maria Moundridou, Maria Virvou

University of Piraeus

Department of Informatics

80 Karaoli and Dimitriou str., Piraeus, 18534, Greece

mariam@unipi.gr, mvirvou@unipi.gr

Abstract-- **Authoring tools for Intelligent Tutoring Systems (ITSs) are meant to provide environments where instructors may author their own ITSs in varying domains. In this way, painful constructions of ITSs, which are not reusable, may be avoided. However, the construction of an authoring tool is associated with many problems, such as the generality of the techniques incorporated, domain-independence, effectiveness for the prospective authors (instructors), and effectiveness for the students who will use the resulting ITSs. In this paper we will report on an empirical study that we conducted in order to design and develop WEAR, an ITS authoring tool for Algebra-related domains operating over the Web. In the study we investigated several aspects concerning the attitude and behaviour of both students and instructors. We also considered aspects relating to the fact that WEAR would be operating over the Web. The study revealed important issues and was then used for the specification of the design of WEAR. A brief description of the developed system is also included in the paper so that the way that the design specifications were put into practice may be shown.**

Index terms-- **Requirements analysis, Intelligent Tutoring Systems, Authoring tools, User Modelling**

I. Introduction

The rapid growth of the Internet and World Wide Web offers new opportunities and challenges for many areas. One of them is education. Web-based education has numerous advantages such as the convenience of taking a course without leaving the workplace or home and the reduced cost [2]. In addition, teachers and educational researchers are encountering both unprecedented opportunities and challenges to adapt networks to their classrooms and research fields [5]. However, most of the educational applications that have been delivered through the World Wide Web lack any underlying reasoning mechanisms that would render the applications more adaptive to learners' needs. In this sense, the integration of Web-based technologies and Intelligent Tutoring Systems would be very beneficial for the purposes of education.

Intelligent Tutoring Systems (ITSs) are computer-based instructional systems aiming at reproducing the behaviour of a human tutor who can adapt his/her teaching to the learning needs of the individual student. As such, ITSs have the ability to present the teaching material in a flexible way and to provide learners with tailored instruction and feedback. This is achieved through the main components constituting an ITS: the domain model, the student model, the teaching model and the user interface. A number of successful evaluations of ITSs (e.g. [10], [11], [13], [7]) have shown that such systems can be educationally effective in comparison with traditional instructional methods either by reducing the amount of time it takes students to reach a particular level of achievement or by improving the achievement levels, given the same time on task.

The main flaw of ITSs and possibly the reason for their limited use in workplaces and classrooms is the complex and time-consuming task of their construction. As estimated by Woolf and Cunningham [18], an hour of instructional material requires more than 200 hours of ITS development time. Furthermore, an already constructed ITS for a specific domain, can neither be reconstructed to function for a different domain, nor can it be altered (i.e. to reflect a different tutoring strategy in the same domain) without spending much time and effort.

An approach to simplifying the ITS construction is the development of ITS authoring tools. The goal of such tools is to provide an environment that can be used by a wider range of people to easily develop cost-effective ITSs. However, the high degree of difficulty and complexity of the construction of an ITS, passes to the construction of an ITS authoring tool which is a yet harder and more complex task. The reason for this is that authoring tools should be able to operate effectively at two levels: at the first level, they should operate effectively for the instructors who intend to author an ITS and at the second level they should incorporate the expertise

M. Moundridou is with the Department of Informatics, University of Piraeus, 80 Karaoli & Dimitriou str., 18534 Piraeus, Greece (telephone: +3010-414-2133, e-mail: mariam@unipi.gr).
M. Virvou is with the Department of Informatics, University of Piraeus, 80 Karaoli & Dimitriou str., 18534 Piraeus, Greece (telephone: +3010-414-2269, e-mail: mvirvou@unipi.gr).

needed to produce ITSs that operate effectively for students. Moreover, they have to use domain-independent methods and be general enough to be used for the construction of many ITSs.

In view of the above, the development of an ITS authoring tool is an issue that needs a lot of attention. In particular, knowledge elicitation and acquisition from domain expert tutors are needed for constructing the framework of the knowledge bases of the ITSs. It is widely acknowledged that the process of knowledge acquisition is a bottleneck in the development of expert systems (e.g. [17], [3], [8]). However, as Twidale points out in [14], in ITSs, knowledge acquisition is necessary for all three main elements: domain model, teaching model and student model; therefore, knowledge acquisition for ITSs is even more difficult than for conventional expert systems. Moreover, in the case of the authoring tool, the expertise to be represented has to be as domain-independent as possible so that it may be used for ITSs of multiple domains.

If the authoring tool is meant to operate over the Web it has to address issues that concern its use by a wide range of users. As noted by Hendler & Feigenbaum [9], the knowledge acquisition bottleneck becomes particularly troubling in the era of the World Wide Web. Indeed, in order to match the large scope of student and instructor needs, backgrounds and interests, a careful and extensive knowledge acquisition phase should precede the development of a Web-based ITS authoring tool.

Knowledge engineering techniques include protocol analysis, observations, interviews and introspection, case analysis and questionnaires [1]. However, often the result is a "data dump" that results in knowledge that is poorly structured, inflexible and incomplete [14]. Therefore, in the case of ITS authoring tools which are complex systems, reporting on the knowledge engineering techniques used, would be an asset for the advancement of this technology. However, reviewing the ITS literature one comes up with the finding that there is a serious lack of any reports on the development of ITS authoring tools. The lack of reports in the literature does not promote the future achievements in the area of authoring tools. Collins in [6] notices: "We have had many technologies introduced in classrooms all over the world, but these innovations have provided remarkably little systematic knowledge or accumulated wisdom to guide the development of future innovations". By conducting empirical studies and presenting their results, future research efforts can both be advanced and made easier. This is even more the case for Web-based ITS authoring tools, which constitute a new demanding area of research.

In this paper we will describe an empirical study that we conducted and led us to the development of WEAR, which is a Web-based ITS authoring tool. What we intended to produce was an authoring tool for building Intelligent Tutoring Systems in Algebra-related domains. As Algebra-related domains we consider these domains that extensively make use of algebraic equations (e.g. physics, economics, chemistry, etc.). Before conducting the empirical study we decided upon the basic requirements of the software we would produce and we determined the aspects that needed further examination. These aspects formed the investigation study that we performed and will be discussed in this paper. In particular, we will describe the two parts, which our study consisted of, as well as the requirements resulting from the Web nature of the tool under development. After describing and discussing the conducted study, we will present in brief the developed system's operation both as a learning and as an authoring environment.

II. Requirements Specifications Concerning Students

The first part of the experiment aimed at identifying issues that related to the construction of a generic frame for ITSs. In particular, the main questions that had to be addressed concerned the domain knowledge, student models and the generation of advice of the ITSs that would be constructed automatically by the authoring tool. However, these issues had to be addressed in a way that would be as domain independent as possible. Therefore, one of the main targets of the experiment was to identify similarities in the problem solving approaches and categories of students' errors in these domains. The categories of error would serve as the basis for the design of the student modelling component of the ITSs. Finally, the experiment also aimed at identifying student characteristics that could be modelled and could be associated with the students' proneness to certain kinds of error and their general performance in problem solving. Such information would be useful for setting the aims of the inferential capabilities of the student modelling component.

The experiment involved two groups of high school students. The two groups originated from courses of two different Algebra-related domains; one was from a physics course (35 subjects) and another one was from an economics course (20 subjects). In order for our sample to be unbiased most of the students who participated were from different high schools. Students of each group were given problems from the corresponding subject and were asked to solve them. The students' answers to these problems were given to domain expert tutors to analyse them. The problems corresponded to what students were taught recently in their classes and were given to them as part of an informal exam. Students who participated in the experiment were also asked to provide their previous years' marks both in the corresponding subject (physics or economics) and in mathematics. This was done, so that the instructors who would analyse the students' protocols could be able to comment on the relation between students' past performance and the current performance.

All this information as well as the papers that the students turned in, were analysed by human experts. There were four human experts who participated in this part of the experiment. Two of them had a first degree and an MSc in Physics and two of them had a first degree and an MSc in Economics. All of them had at least 4 years experience in teaching at schools. The four instructors were different from the instructors that the students had at their schools. This was done so that the students who participated would not feel intimidated or nervous.

Concerning the possible similarities in problem solving approaches of different Algebra-related domains, the experiment revealed that there was a standard method for solving particular problems that were set to students of various levels:

- students should read and comprehend what is given and what is asked in the stated problem and they should form the equations(s) needed for the problem to be solved
- students should then solve the system of equations for the unknown variable(s) of the problem

For example, in the domain of physics one problem of this kind is the following: “A force of 100 Newtons is acting on a 25 kg object, which is initially stable. After 10 seconds how much is the impulse?” For a student to solve this problem s/he should use the equations F=m*a, J=m*v and v=v0+a*t. S/he should replace each known variable with its value and solve for the unknown variables. Similarly, in the domain of economics one such problem could be this: “Given that the Gross National Product – GNP is 1500, Net Factor Payments from abroad – NFP is 20, Investment – I is 250 and National Saving – S is 300, find the Gross Domestic Product – GDP and Net Exports – NX.” Again, students should replace the given data in the equations S=I+CA, CA=NX+NFP and GDP=GNP-NFP and solve for the unknown variables.

In such kind of problem and irrespective of the specific domain to which these problems belong, each of the above mentioned steps for solving the problem can be an error source. In particular, students may not form correctly the equations that are needed for the problem or they may not successfully solve the equations. The former kind of errors can be considered as “domain errors” and the latter as “mathematical errors”. These two error categories can be further divided into sub-categories (Fig. 1).

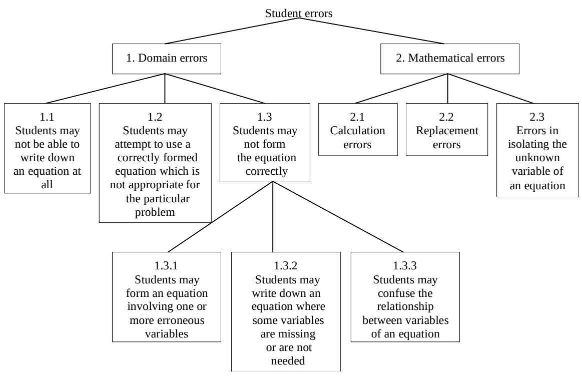

Fig. 1. Categories of student errors

Concerning the “domain errors”, the students’ answers manifested three cases: either students may not be able to write down an equation at all, or students may attempt to use a correctly formed equation which is not appropriate for the particular problem, or students may not form the equation correctly. In this last case, the possible error is either that students may form an equation that involved one or more erroneous variables, or students may write down an equation where some variables are missing or not needed, or that students may confuse the relationship between the variables of an equation. Concerning the “mathematical errors”, these are distinguished in calculation errors, replacement errors and errors in isolating the unknown variable of an equation.

The experiment showed how the errors were distributed along the aforementioned categories and sub-categories of error in each of the domains of the study. Most kinds of error were present at the papers of both domain groups. *This indicates the need for the ITSs to be able to recognise all these different errors and provide students with appropriate feedback.*

The comparison of the results of the two groups showed that although most of the error categories were present at the papers of both groups, the percentages of each error category differed between the physics and the economics domain. This means that the categories of error may be the same for different algebra-related domains but the distribution of these errors depends on the particular domain. This is not an unexpected finding: for example, the fact that 50% of the answers to physics problems contained mathematical errors whereas in the economics group these were only 11% probably results from the fact that in physics the equations that students should solve are far more complicated than in economics and the values of the variables are usually decimal; therefore, physics students are more prone to mathematical errors. As a consequence of that, *the importance of each kind of error varies considerably within different domains and this affects both the students’ grade and the instructional actions that a tutor should perform.*

III. Requirements Specifications Concerning Authors

The second and most extensive part of our experiment involved instructors only. The main aim of this part of the experiment was to identify issues that related to the instructors’ needs and expectations when they were building an ITS using an authoring tool. In that way, we would be able to decide upon the final form that each component of the generated ITSs should have and most importantly upon the way the authoring of the ITSs should be performed and supported.

This part of the experiment involved six instructors. The six instructors were different from those involved in the first part. They all had a first and/or higher degree in Physics or Mathematics and were all experienced as classroom instructors. Half of them had more than 10 years teaching experience and the rest of them from 3 to 5 years. Finally, we should note that all of the instructors were very competent computer users.

The six instructors were given a description of the first part of the experiment along with the results of it and were asked to categorise the most frequent student errors and associate each of them to a stereotype of student. Furthermore, we asked them to grade several student papers and also to define the difficulty level of each exercise that was given to students. Instructors were also asked to answer some questions concerning their teaching style and preferences. All these were done so that we could collect information about the instructors' attitude towards teaching and students' learning in real settings. This information would lead to design decisions concerning the following issues: (i) how the generated by the authoring tool ITS should be acting as a virtual instructor and (ii) what parts of the ITS should be parameterised for the instructors to be allowed to define their own structure or behaviour.

Based on the analysis of the information collected from the instructors in this part of the experiment, we reached the following general conclusion: The instructors' attitude toward the teaching/learning process varied a lot among them. As will be discussed in the subsequent paragraphs, different instructors associated student errors to student stereotypes in different ways; they also gave different weights to the same student errors, they graded student papers differently and finally the level of difficulty they assigned to each problem was in some cases very dissimilar.

In particular, the levels of difficulty (ranging from very easy to very difficult) that instructors assigned to five physics problems were in some cases very diverse (e.g. for a particular problem half of the instructors stated that it was difficult and the other half that it was very easy). Based on this result, we can assume that there are cases when an instructor may overestimate or underestimate the level of difficulty of the problems s/he poses to his/her class. In a real setting the instructor is able to explore immediately the results of his/her teaching and reconsider his/her opinion. *To be able to do the same (adjust his/her teaching decisions, such as the difficulty level of problems) in an e-learning scenario, the instructor should be offered feedback by the authoring tool concerning the performance of his/her class.*

Instructors were also asked to state for each kind of error the frequency it occurred in real settings and to give a weight for its importance. The instructors' answers indicated that there were errors for which instructors almost agreed on their frequency and importance and others for which they completely disagreed. For example, for the error that students make when they attempt to use a correctly formed equation which is not appropriate for the particular problem, almost half of the instructors stated that this happens rarely, whereas the other half stated that this is a frequent mistake. Accordingly, some of the instructors considered this kind of error to be very important and some others to be not so important. Twidale [15] reports on a similar finding and notes that the differing importance attached by different teachers to certain errors affects the degree to which they are likely to intervene in a student's action. He also argues that there appear to be at least three different ways that teachers can classify errors: the complexity of the concept, the difficulty of learning the correct rule and the personal preference of the teacher. Based on this, and on our observations we can argue that *instructors when authoring an ITS should be allowed to have the responsibility for defining the importance of each kind of error since this affects the students' marks and reflects the instructors' personal teaching style*.

From the instructors' answers to questions concerning their teaching style and preferences we came up with the finding that their attitude towards their students is sometimes based on certain students' characteristics and not only on the students' current and actual performance. For example, some instructors stated that the grade they gave to a student depended on the effort that s/he had made; some others stated that they were more strict in grading a student that repeatedly made the same mistake or less strict with novice or careless students. Thus, *an ITS authoring tool should monitor students closely, collect observable information of this kind and ask the instructor to specify which parameters should be used to grade students*.

IV. Requirements Specifications Concerning the Web

Beyond the design guidelines that the first two parts of the study brought to light, we had to consider also the aspects relating to the fact that the authoring tool would be operating over the Web. A main problem for a Web-based ITS is that it aims at reaching a heterogeneous group of learners in settings where no teacher is available to help users during the learning process. In order to match the large scope of student needs, backgrounds and interests, the Web-based ITS should adapt the interaction with different students to a greater extent than a standalone ITS should do. This means that the ITS should maintain fine-grained student models and exploit the information they hold in order to: (i) support students during the problem solving process and (ii) suggest to each of them the appropriate topics of the curriculum to study. This facility is called adaptive navigation support [4] and should definitely be provided by the ITSs that the authoring tool will be generating.

Similarly to Web-based ITSs, authoring tools for Web-based ITSs are addressed to a wide range of instructors with varying beliefs, preferences and teaching styles. This implies that the authoring tool should be adapting to different instructors in order to assist them in an effective way in authoring their ITSs. This can be achieved by maintaining user models not only for the students but also for the authors of the ITSs. Furthermore, the Web provides the opportunity of collaboration among instructors. The authoring tool should offer instructors the ability of communicating with each other and sharing parts of their courses. Enabling the collaboration among instructors would have the following benefits: (i) it would save them valuable time, since they would be able to use parts of the other instructors' courses, and (ii) novice instructors could be assisted by more experienced peers who have previously used the tool.

V. The Resulting Design of WEAR

The aforementioned analysis served as a basis for the design and development of a Web-based authoring tool for the construction of Intelligent Tutoring Systems. The tool is called WEAR, which stands for WEb-based authoring tool for Algebra-Related domains. An earlier prototype of WEAR is described in [16]. WEAR provides an authoring environment where instructors may author their ITSs and a learning environment where students may interact with the resulting ITS. In the learning environment, students are presented with a number of problems to work on and are provided with individualised feedback while they are solving them. They also have at their disposal an electronic textbook and are offered navigation support adapted to their individual knowledge. In the authoring environment, the instructor is able to construct new problems, retrieve previously created ones and author the adaptive electronic textbook.

1) Operation for Authors

The tool takes input from a human instructor about a specific equation-related domain (e.g. economics). This input consists of knowledge about variables, units of measure, formulae and their relation. An example of input to the system that an instructor could provide to describe a portion of the domain of economics is shown in Table 1.

TABLE I
VARIABLES & EQUATIONS DESCRIBING A PORTION OF THE DOMAIN OF ECONOMICS

Variable's description	Variable's name
Gross Domestic Product	GDP
Gross National Product	GNP
Net Factor Payments from abroad	NFP
Private Consumption	C
Investment	I
Government consumption and investment	G
Net exports	NX
Private disposable income	DY
Transfers received from the Government	TR
Interest payments on the Government Debt	INT
Taxes paid to the Government	T
Private saving	Spvt
Government saving	Sgovt
National saving	S
Current account balance	CA
Equations	
GDP=GNP-NFP	Spvt=DY-C
GDP=C+I+G+NX	Sgovt=T-TR-INT-G
DY=GDP+NFP+TR+INT-T	S=Spvt+Sgovt
CA=NX+NFP	S=I+CA

WEAR provides the facility of assisting the instructor to create problems where students are asked to give the solution step by step, or multiple-choice tests. When an instructor wishes to create problems s/he is guided by the system. At each step of this procedure the instructor should specify values for some parameters needed to construct a problem. In particular, the procedure of constructing a problem is the following: The system displays every variable that the human instructor has entered when describing the domain and requests the unknown. The system considers automatically all the variables, which depend on the "unknown" (according to the equations), as possible given data. These variables are shown to the instructor who should now enter their values. The system follows the instructor's actions and reports any inconsistencies. For example, if the instructor enters values for fewer variables than those needed for the problem to be solvable then the system points out the error. Finally, the system produces a simple problem text describing the given and asked data, which the instructor may change to make it more realistic and comprehensible. The information concerning the known and unknown variables is used by WEAR to examine the domain equations and isolate the ones that are needed for the problem to be solved. After the construction of a problem the tool lets the instructor preview the problem text and the solution of the problem as formulated by the system. At this point, the instructor is asked to assign to the problem the appropriate level of difficulty. The system uses this measure in order to suggest to each student (while in student's mode) what problem to try next.

Beyond constructing a problem by himself/herself, the instructor has the ability to explore the problems constructed by others and choose the ones that s/he desires to be accessible by his/her class. Since new problems (belonging to different domains, involving different variables, etc.) can be continuously added to the system, there is no way for the system to have fixed categories of problem. Every time an instructor constructs a new problem the system performs this problem's categorisation based on some parameters. The problems are first categorised according to the domain to which they belong. At a second level the problems of each domain are categorised according to the variables they involve and their level of difficulty. Instructors are allowed either to browse the collection of problems by selecting the categories and sub-categories that match their needs and interests, or to search the entire collection using some keywords. An instructor modelling mechanism incorporated in the system is responsible for tailoring the interaction of the instructors with the system to the instructors' needs.

Finally, WEAR allows the authoring of electronic textbooks by instructors and delivers them over the WWW to learners [12]. These textbooks offer navigation support to students, adapted to their individual needs and knowledge.

2) Operation for Students

Information obtained from student models as well as knowledge of the domain being taught, are exploited by WEAR to provide adaptive navigation support to students. To achieve this, WEAR makes use of the adaptive link annotation technique: students interacting with the system see visual cues (different icons next to each link) that inform them about the current state both of the available problems and of the topics constituting the teaching material. This is done in order to facilitate the student's choice about which problem to solve next and which topic to study, as well as to provide them with information concerning the already mastered topics and concepts.

When a student attempts to solve a problem, the system provides an environment where the student gives the solution step by step. At first the student is presented with a problem statement like the following: "A force of 100 Newtons is acting on a 25 kg object, which is initially stable. After 10 seconds how much is the impulse?". The student is requested to write down the equations that are needed to solve the problem and then s/he is requested to mathematically solve the problem. To detect the erroneous answers the system compares the student's solution to its own at every step. The system's solution is generated by WEAR's Problem Solver. The Problem Solver incorporates knowledge about how to solve systems of linear equations correctly and may generate the solution to a problem using information about the specific domain to which the problem belongs (e.g. physics). During the process of solving a problem the student's actions are monitored by the system. In case of an erroneous action, the Problem Solver passes the student's answer to the Student modeller, which is then responsible for diagnosing the cause of the error. Based on this diagnosis, the system provides the student with the appropriate feedback message. The errors that are recognised by WEAR's Student modeller are these that the empirical study indicated that exist.

VI. CONCLUSIONS

In this paper we reported on an empirical study that we conducted in order to design and develop WEAR, a Web-based ITS authoring tool for Algebra-related domains. In the two parts of the study we investigated several aspects concerning both students and instructors and the requirements of a system that would operate over the Web. This effort proved to be really useful in defining the design guidelines of WEAR's authoring and learning environments. Beyond the practical issues that we explored (e.g. kinds of students' errors that the ITS should recognise), the emphasis was given to the exploration of the instructors' beliefs and actions. This gave us insight for specific design guidelines concerning the authoring procedure. Finally, the fact that WEAR would be operating over the Web has set requirements for a high degree of adaptivity to both classes of user: students and instructors.

An important asset of WEAR is the fact that its design was based on the results of an extensive and multilateral investigation. The dearth of reports on knowledge acquisition for ITSs is a fact that has already been acknowledged in the literature (e.g. [15]). This is even more the case for authoring tools where such reports are completely absent. Reporting on studies like the one described in this paper is by itself a propulsion for the research in the area of ITSs and ITS authoring tools since these studies can assist in the accumulation of knowledge to guide the development of future systems.

REFERENCES

[1] J. Bell, and R.J. Hardiman, "The third role – the naturalistic knowledge engineer," in *Knowledge elicitation: Principles, Techniques, and Applications*, D. Diaper, Ed. Chichester, England: Ellis Horwood Ltd., 1989, pp. 49-85.

[2] M. Berz, B. Erdelyi, and J. Hoefkens, "Experiences with interactive remote graduate instruction in beam physics," *Journal of Interactive Learning Research*, vol. 10, pp. 49-58, 1999.

[3] J.H. Boose, "A survey of knowledge acquisition techniques and tools," in *Readings in Knowledge Acquisition and Learning*, B.G. Buchanan, and D.C. Wilkins, Eds. San Mateo, CA: Morgan Kaufmann, 1993, pp. 29-56.

[4] P. Brusilovsky, "Methods and techniques of adaptive hypermedia," *User Modeling and User-Adapted Interaction*, vol. 6, pp. 87-129, 1996.

[5] C. Chou, "Developing CLUE: A formative evaluation system for computer network learning courseware," *Journal of Interactive Learning Research*, vol. 10, pp. 179-193, 1999.

[6] A. Collins, "Toward a design science of education," in *New directions in educational technology*, E. Scanlon, and T. O'Shea, Eds. Berlin: Springer, 1992, pp. 15-22.

[7] A.T. Corbett, and J.R. Anderson, "LISP intelligent tutoring system: Research in skill acquisition," in *Computer-Assisted Instruction and Intelligent Tutoring Systems: Shared Goals and Complementary Approaches*, J.H. Larkin, and R.W. Chabay, Eds. Lawrence Erlbaum, 1992, pp. 73-109.

[8] C. Garg-Janardan, and G. Salvendy, "A structured knowledge elicitation methodology for building expert systems," *International Journal of Man-Machine Studies*, vol. 29, pp. 377-406, 1988.

[9] J. Hendler, and E.A. Feigenbaum, "Knowledge is power: the semantic Web vision," in *Web Intelligence: Research and Development, Proceedings of the First Asia-Pacific Conference, WI 2001*, N. Zhong, Y. Yao, Liu, and S. Ohsuga Eds. Lecture Notes in Artificial Intelligence, vol. 2198, Berlin: Springer, 2001, pp. 18-29.

[10] K.R. Koedinger, J.R. Anderson, W. Hadley, and M. Mark, "Intelligent tutoring goes to school in the big city," *International Journal of Artificial Intelligence in Education*, vol. 8, pp. 30-43, 1997.

[11] S.P. Lajoie, "Computer environments as cognitive tools for enhancing learning," in *Computers as Cognitive Tools*, S.P. Lajoie, and S.J. Derry, Eds. Lawrence Erlbaum, 1993, pp. 261-288.

[12] M. Moundridou, and M. Virvou, "Authoring and delivering adaptive Web-based textbooks using WEAR," in *IEEE International Conference on Advanced Learning Technologies - ICALT 2001*, T. Okamoto, R. Hartley, Kinshuk, and J.P. Klus, Eds. Los Alamitos, California: IEEE Computer Society, 2001, pp. 185-188.

[13] V. Shute, and R. Glaser, "A large-scale evaluation of an intelligent discovery world: Smithtown," *Interactive Learning Environments*, vol. 1, pp. 51-77, 1990.

[14] D.A. Thurman, D.M. Brann, and C.M. Mitchell, "An architecture to support incremental automation of complex systems," in *Proceedings of the 1997 IEEE International Conference on Systems, Man and Cybernetics*, 1997, pp. 1174-1179.

[15] M. Twidale, "Knowledge acquisition for intelligent tutoring systems," in *Cognitive Modelling and Interactive Environments in Language Learning*, F.L. Engel, D.G. Bouwhuis, T. Bösser, and G. d'Ydewalle, Eds. Berlin: Springer-Verlag, 1992, pp. 62-71.

[16] M. Virvou, and M. Moundridou, "A web-based authoring tool for algebra-related Intelligent Tutoring Systems," *Educational Technology and Society*, vol. 3, pp. 61-70, 2000.

[17] S. Walczak, "Knowledge acquisition and knowledge representation with class: the object-oriented paradigm," *Expert Systems with Applications*, vol. 15, pp. 235-244, 1998.

[18] B.P. Woolf, and P.A. Cunningham, "Multiple knowledge sources in intelligent teaching systems," *IEEE Expert*, vol. 2, pp. 41-54, 1987.