



Learning Technology

publication of

IEEE Computer Society
Learning Technology Task Force (LTTF)



http://lttf.ieee.org/learn_tech/

Volume 3 Issue 3

ISSN 1438-0625

July 2001

[Editorial board](#)

[Subscription](#)


[Author guidelines](#)

[Advertising in the newsletter](#)

Contents

- [From the editor ..](#)
- [International Conference on Advanced Learning Technologies \(ICALT 2000\), August 6-8, 2001, Madison, Wisconsin, USA](#)
- [Supporting and Increasing University Faculty Use of Computers as Teaching Tools: "Geek Week!"](#) (Abbie Brown)
- [Project-Based Learning Just Became Easy: An Introduction to WebQuests](#) (Adam Garry)
- [Employing Case Based Reasoning in Asynchronous distance education](#) (Avgoustos. A. Tsinakos, Kostantinos. G. Margaritis)
- [Magic or Realism? Transforming learning styles into design features in net-based education](#) (Carl Eneroth)
- [ActiveMath, a Web-Based Learning Environment](#) (Paul Libbrecht, Erica Melis and Carsten Ullrich)
- [Chat Services with a Voice](#) (Hira Sathu, Ranjana Shukla and Zhong Tang)
- [Telemachus: A system for submission and assessment of students programs](#) (Maya Satratzemi, Vassilios Dagdilelis and Georgios Evangelidis)
- [A Pattern Language for Architectures of Intelligent Tutors](#) (Vladan Devedzic)
- [Webbus, Tailormade Transport on the World Wide Web: An Educational Webgame](#) (Joop van Schie and Mark Oostendorp)
- [Web-based Adaptive Testing](#) (T. M. Sri Krishna)
- [Context-bound evaluations of courseware in higher education](#) (Peter Hosie)
- [Students Building Communities—Funding Schools](#) (Avis Marie Haynes)
- [Visualizing Thinking with Digital Imagery](#) (Brian K Smith)
- [sitesALIVE! to Launch New Sailing Adventure for Classroom Study](#) (Cindy Collins)
- [Technology Based Training in Ancient Egypt](#) (Jan Seabrook and Nick Rushby)
- [Digital Environments: Monitoring Changes to Teaching](#) (Robert Fox)
- [Educational Software for Computer Engineering: A Case Study of an Interactive BDD Tool](#) (Rolf

 Drechsler)

 [Chaucer in Cyberspace: Linking the Medieval and the Modern Worlds through Technology](#) (Thomas J. Napierkowski)

 [Conference announcements](#)

From the editor ..

Welcome to the July 2001 issue of *Learning Technology*.

The IEEE International Conference on Advanced Learning Technologies, Madison, USA (August 6-8, 2001) is turning out to be a very high quality conference. The website of the event is <http://lttf.ieee.org/icalt2001/> . The call for participation is available in this newsletter below.

You are also welcome to complete the FREE MEMBERSHIP FORM for Learning Technology Task Force. Please complete the form at: <http://lttf.ieee.org/join.htm>.

Besides, if you are involved in research and/or implementation of any aspect of advanced learning technologies, I invite you to contribute your own work in progress, project reports, case studies, and events announcements in this newsletter. For more details, please refer author guidelines at http://lttf.ieee.org/learn_tech/authors.html.

Kinshuk

Editor,

Learning Technology Newsletter

kinshuk@massey.ac.nz

[Back to contents](#)

IEEE International Conference on Advanced Learning Technologies (ICALT 2000) (Call For Participation)

6-8 August 2001

Madison, Wi, USA

<http://lttf.ieee.org/icalt2001/>

Proceedings published by:

IEEE Computer Society Press

Keynote/invited speakers:

1. Tim O'Shea, Master of Birkbeck, University of London, United Kingdom
2. Elliot Soloway, University of Michigan, USA

The registration form, accommodation and other details are available on the website.

Further inquiries:

John Klus (klus@engr.wisc.edu)

Kinshuk (kinshuk@massey.ac.nz)

[Back to contents](#)

Supporting and Increasing University Faculty Use of Computers as Teaching Tools: "Geek Week!"

Although the observation is drawn from a single case study, it seems there may be a correlation between the distribution of propeller beanies and an increase in the computer-based technology skills of university faculty members. This is based on a recent, week-long series of workshops held at Washington State University (WSU) on the subject of using innovative technologies for teaching purposes (the beanies were included in each participant's registration package, and were worn with great pride during the week).

Washington State University's *teach.edu* program (a U.S. Department of Education PT3 initiative) provides ongoing support to faculty members interested in restructuring courses and lesson plans in order to make better use of innovative technologies for the purpose of educating pre-service, K-12 teachers. This year, as a part of the program, faculty members in WSU's College of Liberal Arts, College of Sciences, and College of Education were offered the opportunity to spend one summer month planning and developing the integration of innovative technologies into their own teaching practice. Twenty-three faculty members were selected based on proposals submitted to a review board; the selected proposals outlined their plan, referenced the National Educational Technology Standards for Teachers (NETS-T) and explained the plan's impact on pre-service teacher education.

In order to support the efforts of those faculty members involved in restructuring their courses, a week-long series of workshops was offered in May of this year; referred to by the participants (and ultimately the planners) as "Geek Week." The goal was to give faculty members an chance to work with new technologies and new teaching methods in a collegial and supportive atmosphere. Along with the faculty members who received summer funding, each college was allowed to invite seven graduate students to participate in the event.

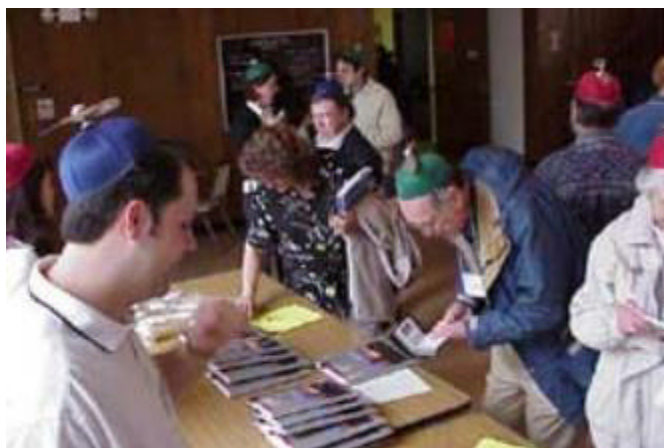


Figure 1. Registration for "Geek Week"

Workshop activities were been planned to respond to the needs of the funded faculty members. A qualitative

coding protocol was applied to the project proposals; critical issues and specific skill-development requests were identified, and a set of activities was developed based on the identified needs.

“Geek Week” Workshop Development:

- Step One: All of the accepted summer proposals were reviewed, paying particular attention to what each person was looking to learn more about in order to complete his/her summer project.
- Step Two: Using Post-It notes, we created an affinity digraph of the various topics revealed in step one.
- Step Three: 5 Superordinate headings were created to help organize the various topics (“The Basics,” “Designing Instruction,” “Getting Geeky,” “Deep Geek” and “Evaluation and Assessment”).
- Step Four: A set of thirteen workshop topics were created that (it was hoped) would most directly address the needs of the group.
- Step Five: The *teach.edu* project coordinators and director met to review the proposed workshop topics.
- Step Six: Workshop facilitators were identified, as were locations for each workshop. Facilitators were asked to supply a few sentences describing the goals and activities for their session.

Many of the workshops are offered multiple times and by multiple facilitators in order to allow each participant to create a schedule that is most appropriate for his/her project, and to offer multiple perspectives on some of the more popular issues. The thirteen workshops offered were:

The Basics

1. The Internet, WWW and e-mail
2. WWW: HTML and web-page editors

Designing Instruction (Human Factors)

3. Principles of Instructional Design
4. Principles of Instructional Media Production
5. An Introduction to Online Learning

Getting “Geeky” (Introduction To Software And Basic Concepts)

6. PowerPoint Workshop
7. HTML and Web-editing Workshop
8. Macromedia Director Workshop
9. Digital Imaging Workshop
10. Digital Sound and Video Workshop

“Deep Geek” (Media Production)

11. Video and Audio in a digital environment
12. Authoring Tools for CD-ROMS and Web-Based “Applets”

Evaluation and Assessment

13. Evaluating Instruction and Instructional Media

Workshop facilitators included members of WSU's College of Education and the university's Center for Teaching, Learning and Technology (CTLT). Most facilitators were expert or "super-user" level technologists. Since the participants came to the workshops with a wide range of abilities and experiences, some were complete novices, others had spent years using computing tools for teaching purposes, there was some concern that the presentations might be inadvertently daunting to some participants if the facilitators addressed the concerns of the more experienced (like most human endeavors, "showing off" seems to be a natural inclination during technology-oriented activities). For this reason, the following was included in the opening pages of the program provided to participants:

Unfortunately, even the most well-meaning geeks are sometimes difficult to understand (even among other geeks!). On behalf of all the workshop facilitators, here are two recommendations for making the most of the workshops and the labs:

Ask questions. We're working under the assumption that, during this week, everyone (including the facilitators) will need clarification about some of the most basic principles of instruction and technology. We're working under the assumption that all questions are good questions.

Insist that everyone speak plain English. Don't let anyone use an acronym or an abbreviation that you don't understand. Geeky acronyms and abbreviations are intended to make life easier; URL is much easier to say than "Universal Resource Locator," but they can sometimes be used (inadvertently, of course) to confuse people. At some point this week, just about everyone will need something, "translated from the geek."

One more thing to consider: This week's most important phrase is "User-Centered." And *you* are the user! In every lab experience, facilitators and hypernauts will assume that anything that goes wrong is the result of shoddy design on the part of the hardware and software manufacturers. *It's not your fault if the manufacturers made things difficult to use!* We will ask that you bear with us when problems arise; we'll do our best to figure out and explain how these tools are supposed to work and what we can do to make them work to your best advantage.

The combination of a qualitative-analysis approach to the creation of the workshops and the advice offered in the program seems to have contributed to the success of the event. Although formal evaluations have not yet been analyzed, the anecdotal information gathered indicates that most people felt satisfied with the workshops and confident in their abilities to further their own personal development. A common statement made many times toward the end of the week was, "Now I know what questions to ask."

Although analysis and a user-centered approach played a large part in the development of a successful program, it is the author's contention that the propeller beanies distributed at the beginning of the week played their own very special part in the process.

Acknowledgement

The author gratefully acknowledges the support of the U.S. Department of Education's PT3 (Preparing Tomorrow's Teachers to Use Technology) program; the work of event organizers Ian Quitadamo, Greg Hooks, Skip Paznokas; and Washington State University's Center for Teaching, Learning and Technology.

References

- International Society for Technology in Education (ISTE) (2000). *National Educational Technology*

Standards for Teachers. Eugene, OR: ISTE.

Abbie Brown, Ph.D.

Department of Teaching & Learning

Washington State University

Tel: +1 509 335 1631

abbie_brown@wsu.edu

[*Back to contents*](#)

Project-Based Learning Just Became Easy: An Introduction to WebQuests

Much of the teaching profession is beginning to incorporate project-based learning (a comprehensive approach designed to engage student investigation of authentic problems) into the curriculum, and many agree that it would be nice to have an aid to make this task easier. Sometimes we struggle to create an environment that allows students to marry technology with higher order thinking skills. To make this happen, a number of teachers are turning to the WebQuest--an inquiry-oriented activity, which some or all of the information that learners interact with comes from resources on the Internet (Dodge 1995).

WebQuests allow students to complete authentic projects and use technology to find and present information and, at the same time, alleviate some of the barriers teachers may find in their attempt to work in the confines of project-based learning. In planning for such learning, we should always start with the end in mind. A WebQuest can provide the educator with the project for a unit of study. After reading through a specific WebQuest, the teacher can then begin to select the standards to be mastered. An alternative way to start would be to select the standards, and then find a WebQuest that would help master those standards. Subsequently, the teacher can decide which enabling activities need to be taught. Every WebQuest has an Introduction, a Task, Resources, a Process, Evaluation (with a Rubric), and a Conclusion.

The WebQuest Way of Doing Things

The *introduction* provides the student with a chance to recall some prior knowledge and to become excited about the project. In the *task* portion, they are given the assignments that they will complete for this particular WebQuest. The *process* walks them through a systematic approach for completion. *Resources* offer the students and the teacher Web pages, books, magazine articles, and so on that will help with the project. By creating an *evaluation* component, the WebQuest allows the students to know how their learning will be measured before they begin their project, and it also helps the teacher as the evaluation is there for them to use or modify. Lastly, it includes a *conclusion* to summarize what the students have done. Some WebQuests offer Enabling Lesson Plans, while others contain extension activities or perhaps a teacher page.

Why WebQuests?

Students enjoy WebQuests because they are given the opportunity to use the Internet to find and apply information. The students also learn to use presentation software, which allows the students to impart their information in a creative way while educating others.

Teachers benefit from WebQuests in a number of ways. For example, rubrics for each project and Web resource addresses are provided, authentic learning occurs, ideas for projects are supplied, and learning is fun for students while they integrate technology. WebQuests can and should be modified by the teacher to fit the needs of the classroom.

The WebQuest Experience

As you can imagine, having the students complete a WebQuest during a project can be quite a wonderful experience. Here are some tips to help you start and succeed with your first few:

1. Always perform the initial WebQuest as a group so that the students may become familiar with the process and ask any questions as you proceed. A good example for this is *Create a Habitat* (<http://www.biopoint.com/WebQuests/dist204/Welcome.html>).
2. Before you create a WebQuest on your topic, search the Internet to make sure it doesn't already exist.
3. For your first WebQuest don't try to create your own--ease your way into the process.
4. Preview your WebQuest before you use it.
5. Change the WebQuest so that it will meet the needs of your class. Sometimes you will have to alter the procedure to suit the one-computer classroom or break up the steps.
6. Remember to let the students explore and have fun. There will be times when the WebQuest does not proceed the way you intended, but learning is still taking place--just go with it.

Once you have used a WebQuest as part of the learning process, you will share the experience with your colleagues. The higher levels of learning that can occur during any given WebQuest are amazing.

If you do not feel that you are technologically savvy enough to take advantage of a WebQuest, remember that there are plenty of resources on the Internet that can help you with software and hardware questions. For Microsoft questions, visit <http://www.microsoft.com/education/tutorial/classroom/default.asp>, and Macintosh questions can be answered at <http://ali.apple.com/>.

Time to Create My Own WebQuest--Where Do I Start?

You have scoured the Internet and you cannot find the WebQuest that you need. It is time to create your own. There are many tools to get you started--for example, a useful design flow chart can be found at <http://edweb.sdsu.edu/webquest/Process/WebQuestDesignProcess.html>. If you like to plan away from the computer, you can access <http://storywind.net/field/minisab/technology/> and print out the design sheet. Also, you can visit <http://edweb.sdsu.edu/webquest/materials.htm> for all the materials and resources you will need to create your own WebQuest.

Once you know what you would like to do and have created a plan, it is time to make your WebQuest. It is *not* necessary to know html (hypertext markup language) to complete a WebQuest. Just knowing the basics

of a Web page will be fine. Below is a list of ways in which you can make and store your WebQuest.

1. You can create the WebQuest in Microsoft Word or AppleWorks and save as a Web page.
2. Knowledge In Knowledge Out (<http://www.kiko.com>) has a WebQuest creation tool that you can use online (Internet Explorer only). It takes you through the process and all you do is click in the boxes and type.
3. Homestead.com (<http://www.homestead.com>) is also a drag-and-drop Web page creator that allows you to make your page online. The best part about this resource is that you don't have to worry about uploading.
4. Geocities (<http://www.geocities.com>) is a wonderful place to store your site. From the main page, you can click on Upload and FTP (File Transfer Protocol), and then you can upload your files.
5. Another great place to store your Web page is <http://nbc1.com>. This site is very unobtrusive and makes uploading easy. When you go to the site, you can click on *My Web Site* and it will take you through the process.
6. Of course, you can always use a Web editor like Microsoft FrontPage, Claris Homepage, Adobe PageMill, Netscape Composer, or Macromedia Dreamweaver. After you create it here you can use any of the above sites to upload your page.

Preparing a WebQuest to be published on the Internet can be frustrating, but there are many sites that make the process very easy. If you learn the basics of Web creation and uploading, then you will be well on your way.

Once you have mastered creating WebQuests, it will be time for you to transfer that knowledge to others. A great place to start is with your students. When students create their own WebQuests, they will be using many higher order skills. They will need to know a great deal about their subject area if they are creating a WebQuest that others must complete.

WebQuests are the curriculum drivers of the future. They provide teachers and students with the necessary tools to thrive in a project-based learning environment. Join the wave of teachers who are turning to WebQuests to facilitate such learning and create an environment in which students are using high order thinking skills to solve real world problems.

WebQuest Tips

1. A good place to start is the WebQuest about WebQuests:
<http://edweb.sdsu.edu/webquest/webquestwebquest.html>
2. Always name your home page index.html.
3. Try to blend some pictures in with your text.
4. Look at the WebQuest rubric before and as you create your site:
<http://edweb.sdsu.edu/webquest/webquestrubric.html>
5. Join the WebQuest listserv. This is a great community of people who can help you with questions. You can join by sending a subscribe message to webquest@egroups.com.
6. Visit your WebQuest periodically after it has been posted to the Internet and check for bad links within your resources.
7. If possible, create a teacher page that gives the teacher some tips and other resources that will help with the completion of the WebQuest.

Web Resources

The Main WebQuest Page

You will find all the material needed to complete and use WebQuests at this site:

<http://edweb.sdsu.edu/webquest/webquest.html>

EdHelper

Here is a site that has a tremendous number of WebQuests sorted by subject area: <http://www.edhelper.com>

The Big WebQuest Collection

This site has tons of resources that have been created by educators and pre-service teachers. I particularly like the Indiana State Extensions and New Mexico State University (WebQuests in both English and Spanish): http://edweb.sdsu.edu/webquest/webquest_collections.htm

My WebQuest collection

WebQuest pages that I have compiled for teachers like you:

<http://www.geocities.com/agarry.geo/webquest.html>

Grades K-2

<http://www.esc20.net/etprojects/formats/webquests/summer99/northside/bug/default.html>

Grades 3-5

Harry Potter WebQuests that ask students to take on some untraditional roles:

<http://www.plainfield.k12.in.us/hschool/webq/webq113/>

Invention WebQuest--This one will take you away from the traditional inventors:

<http://www.esc20.net/etprojects/formats/webquests/summer99/northside/Inventions/default.html>

Grades 6-8

Roller Coaster Madness--The students will be creating the fastest roller coaster ever, using the history and science of roller coasters to do this. Also, they will have a chance to simulate their creations online:

<http://www.esc2.net/TIELevel2/projects/roller/>

Grades 9-12

Personal Reflections on Vietnam--This WebQuest will have your class debating the war for many weeks:

<http://www.richmond.edu/~ed344/webquests/vietnam/intro.html>

Adam Garry

School Consultant, Co-nect

agarry@co-nect.net

[Back to contents](#)

Employing Case Based Reasoning in Asynchronous distance education

Abstract

The current paper describes the employment of Case Based Reasoning in the implementation of domain independent educational environment called See Yourself Improve (SYIM).

SYIM aims to the provision of personalized distance education services during asynchronous distance education sessions, contributing also to the construction of student models.

Introduction

Asynchronous distance education appears as one of the most appealing instructional delivery methods as it combines flexibility to access teaching material with time to reflect, self-study techniques with peer-to-peer collaboration, and the use of low-cost technology. However, a number of pedagogical problems appear when confronted with asynchronous distance education (Ritchie and Hoffman, 1997; Wulf, 1996; Pernici and Casati, 1997; Relan and Gillani, 1997; Hunt 1999; Pritchard, 1998; Khan, 1997; Harasim, 1990; Hall, 1997b; Holden and Wedman, 1993).

Students may feel isolated due to the distance factor or due to the lack of live interactions with the tutor or with classmates. On the other hand, due to the discontinuous character of asynchronous distance education, these sessions may also suffer from lack of intense supervision and lack of specialized feedback to students, according to their individual learning needs and preferences. So monitoring students' performance may become a problematic process, while conclusions about the teaching effectiveness of a particular instructor may also be difficult to draw.

SYIM is a domain independent educational environment which has been developed in order to remedy some of the educational problems appear in asynchronous distance education, such as lack of intense supervision, hazy monitoring of the students' performance progress, inability recording individual learning needs etc (Tsinakos and Margaritis, 2001).

The core idea of SYIM is to help the tutors to monitor the individual learning needs and the misconceptions of the distance students and to keep a track of the feedback provided to each student. Additionally SYIM provides to the students the benefit of the intense supervision related to their individual learning needs and the effective support and guidance on how to overcome a misconception or remedy a performance gap in order to improve both their performance and their context comprehension.

The idea of employing Case Based Reasoning

Case Based Reasoning (CBR) is a method used in the field of Artificial Intelligence, which explores a range of human cognitive behaviour, including learning, memory, planing and problem solving (Han 1993, p. 8).

A variety of definitions regarding CBR are available in the literature (Leake, 1996; Riesbeck & Schank, 1989; Aamodt & Plaza, 1994; Kolodner, 1991). A common node among these definitions is that CBR solves a new problem by recognize its similarity to a previously known problem and adapting the known solution to the new problem (Riesbeck and Schank, 1989; Kolonder 1990).

Research in CBR deals with a variety of problem solving techniques which make use of specific previous cases (use of experience) rather than general domain knowledge. Much of human reasoning is case-based

rather than rule-based. An experience tutor can instantly recognise the student's errors without having to execute a number set of rules stored in a rule base. On the contrary the tutor based on the experience can recall cases from the past (previous instructional sessions) which are similar to the current tackled, and use the to solve or to remediate the problem.

"When the tutor encounters an unfamiliar situation he/she attempts to retrieve the most comparable cases in his/her experience, and modifies these cases to fit the new situation. If the modified case works, then the tutor memorize the case for future use by updating his/her memory." (Han 1993, p. 9)

Research has been done on the potentiality of CBR to improve the process of student modelling as some researchers studied domain knowledge representation schemes, which might assist students' learning process by focusing on memory structures (Bumbaca, 1988; Riesbeck and Schank 1991).

The employment of Case Based Reasoning in the SYIM educational environment is believed that it will be beneficial both for the tutors and the students. Such employment aims to automate the process of replying to the student's misconceptions, and will contribute to the construction of tutoring paths in order to advice the student how to overcome a performance problem based on the SYIM's experience of previous cases.

The implementation schema

In the initial version of SYIM the communication schema between the tutor and the student follow the procedure that appears in Figure 1. According that procedure, each of the student's misconception was posted directly to the tutor.

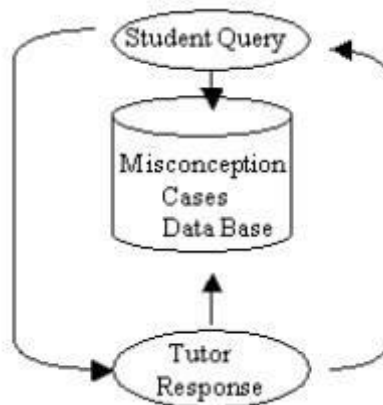


Figure 1. Misconceptions' communication schema

Similarly, each of the tutor's reply was posted directly towards the student and by the same time, both messages were saved in a Misconception Data Base which was responsible to record the tutor-student interaction.

The process of answering students' queries can become a time consuming procedure for the tutor as the number of misconception posting increases during an educational session. Additionally the delay of receiving

tutor's reply, on the student's side, is also increased by time.

To overcome this problem, in the new version of SYIM, CBR have been employed, resulting to the radical modification of the tutor/student communication schema (Figure 2).

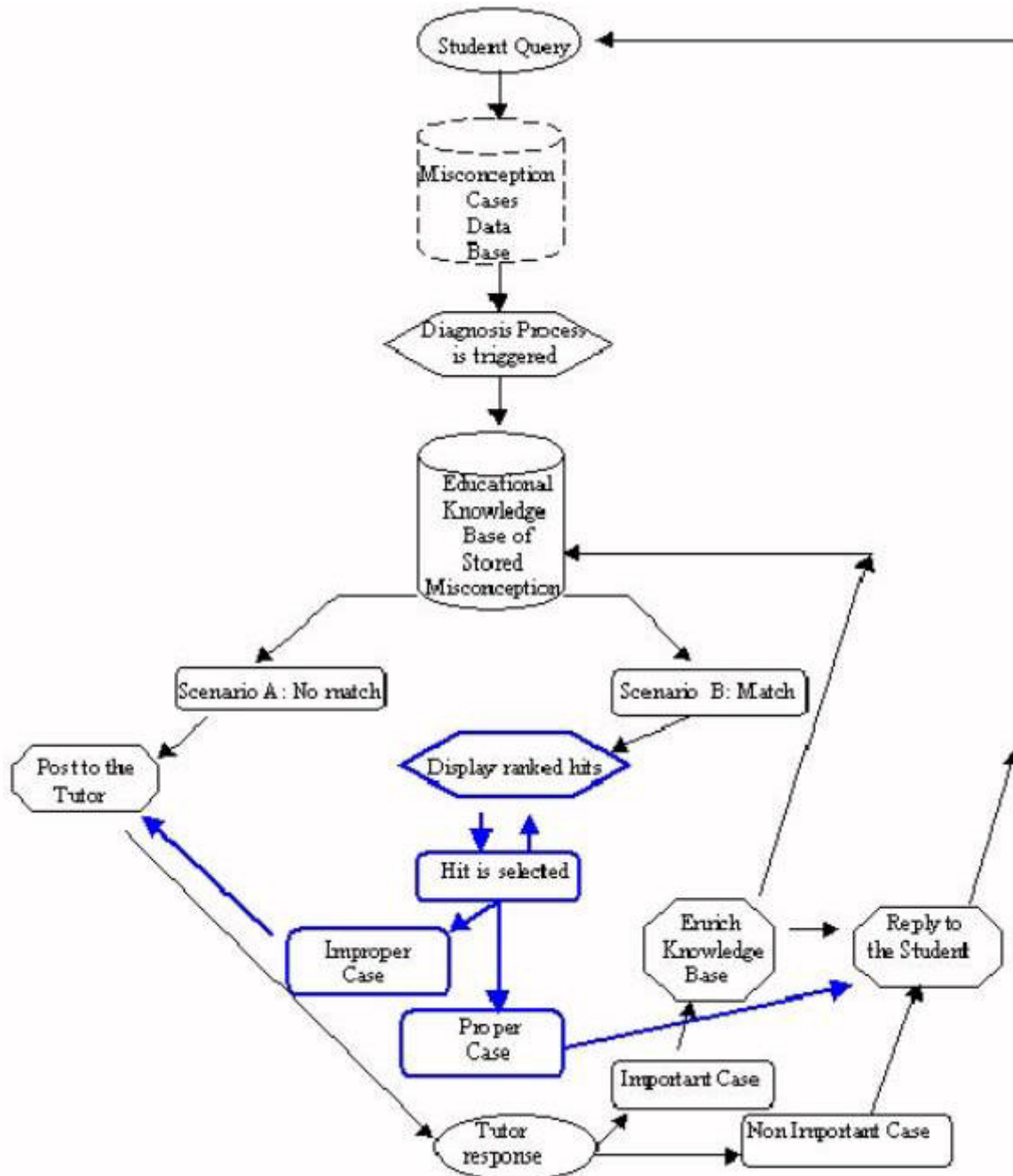


Figure 2. Employment of CBR in SYIM

According the new schema, when a student posts a misconception-query, the posting does not reach the tutor immediately. Each student's posting is recorded in the Misconception Cases Data Base. The scope behind this procedure is to preserve the student's personalized information used for the construction of the student model by SYIM. Furthermore this procedure enables the tutor to be aware of each student's the misconceptions postings, regardless of the reply source (system or tutor). As a next step, a Diagnosis

Process is triggered in order the SYIM system to be search for the location of a relevant case that might be the answer on student's original posting. The Diagnosis Process, in order to identify a similar case, applies controlled vocabulary search in addition with free text search among the contents of the "Educational Knowledge Base" where the misconceptions with further educational value are stored. Therefore two scenarios are possible:

Scenario a "No relevant case is found": This scenario is valid when the Diagnosis Process fail to identify and retrieve a similar case. The reason for this failure may be that the student's posting is an original misconception that has never been asked in the past. In such a case, the misconception is posted directly to the tutor.

The tutor now is responsible not only to reply to the student's question but also to decide if the posted misconception is an important *new case* which has a further educational value (helpful for other students) and therefore should enrich the contents of the Educational Knowledge Base.

If so, tutor's reply along with the student's question, beyond reaching the student, is also stored in the Educational Knowledge Base. If the tutor decides that the misconception is educationally unimportant, then the tutor's reply goes directly to the student without being registered in the Educational Knowledge Base. In this way, the Educational Knowledge Base is enriched only with those cases that are critical and facilitate the instructional process, preserving in that way, its internal integrity and validity.

Scenario b "One or more relevant cases are found": If this scenario is valid, then the retrieved set of cases is displayed to the student, in a ranked order according to their relevancy towards the posted query. Each case include the question posted in the past by a different student, which concerns a similar concept to the newly posted query, in addition with the tutor's reply on how the student could overcome the particular problem. Therefore the student can select among the listed cases, the most *proper case* which provide an answer to his query. Having done that, the process of "Reply to the Student" is automatically terminated without the interference of the tutor. Furthermore the student can vote for each case if it was really helpful or not.

If the student can not identify a proper case which answer the initial misconception (*improper case*), then the process of "Scenario a" is triggered.

It is worth to mention that a case can include as part of its contents a number student-tutor nested dialogue messages which formulate a chain of navigational tutoring steps (tutoring paths) on how a student can overcome a particular problem.

This feature is extremely useful and time saving for the tutor, considering that a number of students' misconception appears repeatedly during an instructional session. On the other hand, this feature is also beneficial for the students as they can easily find a pre stored answers to their question and therefore they can proceed in the instructional material without time delays due to the inability of the tutor to provide an immediate answer.

A further additional benefit of this feature is that a number of past related queries are displayed to the student and therefore the latter can identify, explore and resolve some other critical instructional concepts

In conclusion, the employment of Case Based Reasoning as described above, acts beneficial both for the tutor and the students. The Diagnosis Process is a time saving feature which assist tutors during their instruction by decreasing the number of the questions seeking for an answer, accelerates the instructional process and also contributes to the content comprehension on the students' side.

References

- Aamodt, A. and Plaza, E., (1994) Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches *AI Communications*, 7(1), pp. 39-59.
- Bumbaca, F., (1988). Intelligent Computer Assisted Instruction: A Theoretical Framework, *International Journal of Man-Machine Studies*, 29, pp. 227-255.
- Hall, B. (1997b). *Web-based training cookbook*. New York: Wiley Computer Publishing.
- Han, J. K., (1993). Case Based Cognitive Modeling: A student modeling methodology for an intelligent tutoring system. *Phd Thesis*, University of Nebraska-Lincoln.
- Harasim, L. (1990). Online education: An environment for collaboration and intellectual amplification. In L. Harasim (Ed.), *Online education: Perspectives on a new environment*. (pp. 39-64). New York: Praeger.
- Holden, M. and Wedman, J.F. (1993). Future issues of computer-mediated communication: The results of a Delphi study. *Educational Technology Research and Development*, 41 (4), 5-24.
- Hunt, D. L. (1999). Web-based training. In W. J. Rothwell & K.J. Sensenig (Eds.), *The sourcebook for self-directed learning* (pp. 121-145). Amherst, MA: HRD Press.
- Khan, B.H. (Ed.). (1997). *Web-based instruction*. Englewood Cliffs, New Jersey: Educational Technology Publications, Inc.
- Kolonder, J.L. (1990). An Introduction to Case-Based Reasoning, *Technical Report*, GIT-ics-9019, School of Information and Computer Science, Georgia Institute of Technology.
- Kolonder, J.L. (1991). Improving Human Decision Making through Case-Based Decision Aiding, *AI Magazine*, 12(2), pp. 52-68.
- Leake, D., (1996) Case-Based Reasoning: Experiences, Lessons, & Future Directions *Menlo Park, California*: AAAI Press.
- Pernici, B. and Casati, F. (1997). The design of distance education applications based on the World Wide Web. In B. H. Khan (Ed.) *Web-based instruction*. (pp. 245-254). Englewood Cliffs, New Jersey: Educational Technology Publications.
- Pritchard, C. L. (1998). *The ASTD handbook of instructional technology*. New York: MacGraw-Hill, Inc.
- Relan, A. and Gillani, B. B. (1997). Web-based instruction and the traditional classroom: Similarities and differences. In B. H. Khan (Ed.) *Web-based instruction*. (pp. 41-46). Englewood Cliffs, New Jersey: Educational Technology Publications.
- Riesbeck, C.K., and Schank, R.C. (1991). From Training to Teaching: Techniques for Case-Based ITS, In Eds. Burns, H., Parlette, J.W., and Redfield, C.L., *Intelligent Tutoring Systems: Evolution in Design*, Lawrence Erlbaum Associates, Hillsdale, NJ, pp 55-78.
- Ritchie, D. C. and Hoffman, B. (1997). Incorporating instructional design principles with the World Wide Web. In B.H. Khan (Ed.) *Web-based instruction*. (pp. 135-138). Englewood Cliffs, New Jersey: Educational Technology Publications.
- Tsinakos, A., and Margaritis, K., (2001). [See Yourself Improve \(SYIM\). An educational environment for the provision of personalized distance education services and the formulation of student models](#)", accepted in *European Journal of Open and Distance Learning (EURODL)*, ISSN 1027-5207.
- Wulf, K. (1996). Training via Internet: Where are we? *Training & Development*, 50 (5), 50-55.

Avgoustos. A. Tsinakos, Kostantinos. G. Margaritis

Department of Informatics

University of Macedonia

Thessaloniki

Greece

tsinakos@uom.gr

kmarg@uom.gr

[Back to contents](#)

Magic or Realism? Transforming learning styles into design features in net-based education

Abstract

Advantages in comparing net-based education to traditional teacher lead learning are recognized. However, in the hype of “e-learning”, the real needs of users/learners are not given sufficient attention. The present article points to the importance of developing net-based education in accordance with principles for user-centered design. The purpose is to demonstrate the use of learner-centered design in accommodating users with different types of learning styles. This article gives an account of a theory of learning styles and it goes on to show how different learning styles may be transformed into design features

(This article is a short version of a paper submitted to the Nordic Interactive Conference in Copenhagen (30/10 – 3/11/01) with the same title. It was written by Carl Eneroth, Cecilia Katzeff and Rasmus Larsson within the project “explore:e-learning styles” in the Explore studio at Interactive Institute, Stockholm, Sweden)

Learner-centered design

To link theories to practical design, it is important to possess adequate methodological tools. Learner-centered design (LCD) provides a framework for such tools [8][7]. The central claim of LCD is that interactive media may embody learning supports, which can address the learner’s knowledge level, motivation, and diversity. These are important aspects when viewing users as learners and designing for usability.

Learning styles

The concept of Learning Styles originated in the attempt among educators and researchers to look for the source of individual differences between children [6]. One such source was Learning Style or learning orientation, which More defines as: “the characteristic or usual strategies of acquiring knowledge, skills and understanding by an individual” [6]. Empirical research indicates that there is a correlation between learning style and the effect of learning. Learners may have different, mostly unconscious styles for learning.

Kolb’s Experiential Learning Model

Kolb found that the uniqueness of these learning styles or ‘most comfortable ways to learn’ is influenced by

“the combination of how people perceive and how people process” [5]. By joining two dimensions of how we perceive (concrete experience and abstract conceptualisation) with how we process (active experimentation and reflective observation), Kolb established four learning styles: diverger, assimilator, converger and accommodator (Figure 1) [4].

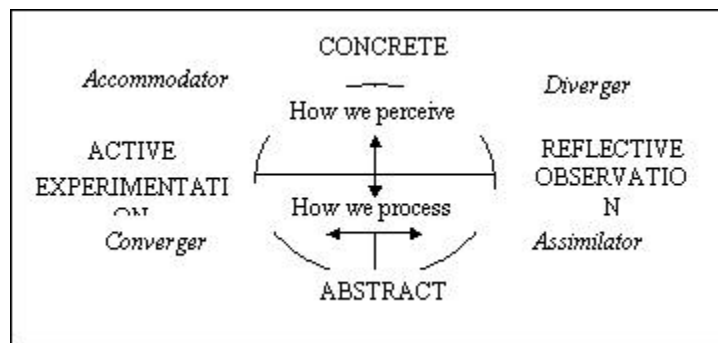


Figure 1. Adapted from Kolb's [4] Learning Style Inventory of how we perceive and how we process information with learning styles presented in italics

In Kolb's model, each learning style is thus a combination of two preferred learning skills: diverger (interpersonal and information skills), assimilator (information and analytical skills), converger (analytical and behavioural skills) and accommodator (behavioural and interpersonal skills) [1]. The focus in the project 'explore:e-learning styles' was on two learning styles: diverger and converger, while the assimilator and accommodator was left for later prototype developments.

Design criteria

Explore's intention when transforming learning styles into design features in Explore's e-learning styles project, was to use the principles of learner-centered design to first identify design criteria for two learning styles and then use these criteria to build a prototype. These were the design criteria used:

Diverger	Converger
<i>Design concept – overall idea</i>	
Prefers an imaginative and varied design that reflects a magic realism.	Prefers a strict and elegant design that reflects a realistic world.
Prefers a surrounding media in which the user 'steps into'.	Prefers hierarchical menus to get an instant overview.
<i>Dramaturgy – first impression and plot</i>	
Enjoys an unpredictable virtual world to experience.	Enjoys a predictable and clear virtual world to explore.
Accepts interactive tools with a dramaturgic effect that is <i>not</i> decisive for solving the problem, but provides a sense of exclusiveness and of having found it, which may act as a trigger for social contact.	Accepts interactive tools with a dramaturgic effect that helps to directly solve the problem and provides options to learn more at a deeper level of detail.
<i>Information architecture – text, speech and other features</i>	
Favours non-linear presentation of information with unintentional learning outcomes.	Favours linear presentation of information with intentional learning outcomes.
Likes non-relevant features in the educational program.	Dislikes non-relevant features in the educational program.
Prefers text and speech that builds relevant social relations in the selected context.	Prefers text and speech that mediates relevant facts in the chosen context.
<i>Collaborative learning – person orientation</i>	
Seeks readily answers in contact with other fictive and real persons, independently of whether they lead to the right answer or not.	Seeks readily answers independently of other fictive and real people, when they are not perceived as necessary to solve the problem.
Seeks guidance by motivated trainers with personal commitment that lead to the right knowledge.	Seeks guidance by an expert with formal merits that possess the right knowledge.

Table 1. Design criteria for two learning styles based on Kolb's model (Figure 1) divided into four themes

Design features – preliminary end results

This article is written at a time when Explore has presented its first functional programs adapted to the diverger and converger learning style, but before the observation, measurement and evaluation of end user behaviour has started. The empirical data gathered from these usability tests will be employed not only in the coming version of the diverger and converger programs, but also in adaptations made for individuals with the assimilator and accommodator learning style. The screen dumps in figure two to seven illustrates aspects design features in the educational program prototypes adapted to the diverger and converger learning style.

Dramaturgy

Figure two introduces an unpredictable virtual world to experience with interactive tools that the *diverger* discovers when clicking randomly on the computer screen. Finding a tool provides the user with more information. The task is to rescue people who are stranded on an island (Figure 2). The underlying idea is to present a meaningful task that answers to the diverger's principal question: why?

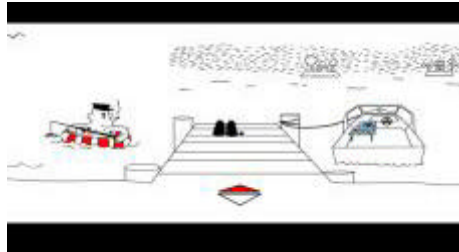


Figure 2. First impression in the Diverger program: an unpredictable virtual world to experience

Figure three demonstrates a predictable world for the *converger* to explore with interactive tools displayed in the hierarchical menu on the top left side of the screen. The menu links to specific topics of direct importance when taking out a course on a sea chart (Figure 3). The underlying idea is to present relevant information that answers to the diverger's main question: what?

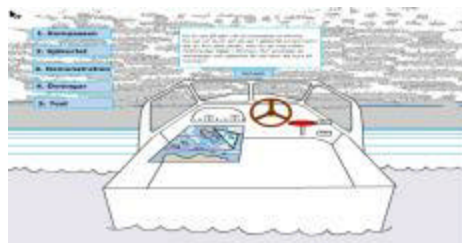


Figure 3. First impression in the Converger program: a predictable virtual world to explore

Information architecture

Figure four illustrates the information architecture and user processes for the *diverger* in the form of non-linear presentation of information. This setting allows for non-relevant features to be discovered in the virtual world, as a mean to indirectly reach the educational goal of the program. The learning outcomes are thus unintentional from the outset (Figure 4).

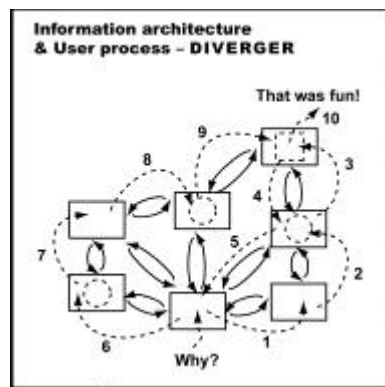


Figure 4. Non-linear presentation of information for the Diverger with unintentional learning outcomes

Figure five illustrates the information architecture and user processes for the *converger*, who favours linear presentation of information. Non-relevant features in the educational program are avoided to more directly guide the user towards the educational goal of the program. The learning outcomes are thus intentional from the outset (Figure 5).

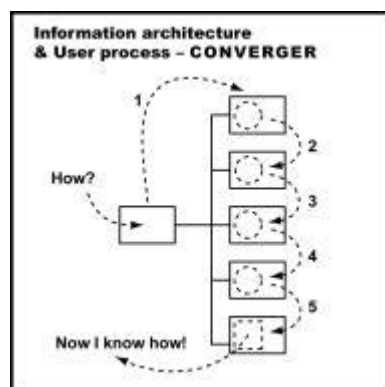


Figure 5. Linear presentation of information for the Converger with intentional learning outcomes

The key in figure six explains the meaning of the boxes and arrows shown in figure six and seven (Figure 6):

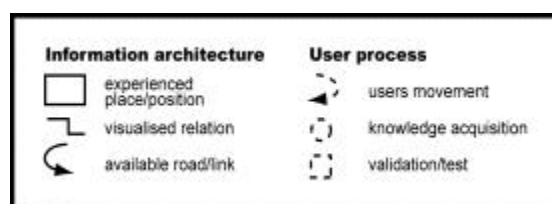


Figure 6. Key explaining the boxes and arrows of figure 6 and 7

Collaborative learning

Figure seven shows a mariner to cater for the *diverger's* social cravings. The engaging mariner introduces reasons for taking out a course on a sea chart, which is useful later on when solving the task to rescue the people stranded on an island (Figure 7).



Figure 7. The Mariner supporting the Diverger's need for social relation building

Figure eight exhibits a sea chart to satisfy the *converger's* need for factual information that leads to the right knowledge. The simple operations employed when taking out a course on a sea chart is shown in sequence with an instructional voice-over (Figure 8).



Figure 8. Sea chart supporting the Converger's need for hands-on factual information

Conclusions

The article have focused upon two major themes:

1. Ideas motivating the work of designing individualised interactive media for learning, and
2. Description of the process of transforming characteristics of learning styles into design criteria.

Although hypotheses have been made about our two prototypes being accommodated for users with two different learning styles, these hypotheses are yet to be tested. One of our prototypes has been designed to best suit divergers and the other prototype has been design to best suit convergers. To test whether we have succeeded in our design endeavours an experiment is planned involving learners predominated by either a convergent or a divergent learning style.

Experiences from using learning-centered design in practice show that:

- Kolb's model (Figure 1) provides target group segmentation that function well with the learner-centered design approach and is easy to communicate internally and externally, but Dunn & Dunn's model (Section 3.6) offers in general more practical design information;
- Testing and evaluating the effect and function of the program content necessitates the user-interface of the application to be well designed, completely implemented and debugged;
- Depending on the design criteria selected for the application, different development methods for steering the production process is required;

The design criteria for Kolb's learning styles (Figure 1) can be structured into at least four themes:

- Design concept that captures the overall idea;
- Dramaturgy that conveys the first impression and the plot;
- Information architecture that reflects the attitude towards the integration of text, speech and other features; and
- Collaborative learning that reveals the approach to learning with or without other persons.

The preliminary end results are that it is possible to design two different programs that transform two learning styles according to Kolb's model (Figure 1) into specific design features.

- For the diverger the program is produced as an unpredictable virtual world to experience, where information is presented in a non-linear fashion, the learning outcomes are unintentional and there is room for social relation building.
- For the converger the program is created to reflect a predictable virtual world to explore, where information is presented in a linear fashion with intentional learning outcomes and hands-on factual information introduced early.

Future directions

Regarding future directions, net-based programs that support process modelling for Kolb's assimilator and object modeling for Kolb's accommodator are ways to further materialise the idea of exchangeable user-interfaces to be used by individuals engaged in computer supported collaborative learning. In a not so distant future, net-based educational programs may be equipped with controls that after being adjusted to an individual learning style dynamically alters the program's user-interface accordingly (Figure 9). In practice, this may be constructed through a limited number of pre-set paths adapted to a specific learning style.

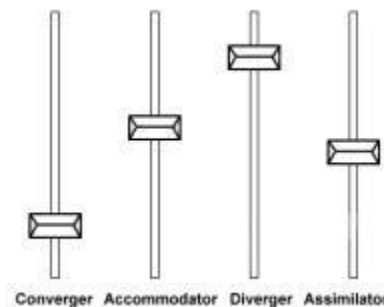


Figure 9. Future vision of a net-based educational program equipped with controls that after being adjusted to an individual learning style dynamically alters the program's user-interface accordingly

Finally, integrating Jung's theory on personality types into Kolb's model opens up for a more precise correlation between individual learning styles and specific design features. This may be accomplished by inviting the test persons to take Myers-Briggs Type Indicator test [5] that is built on the personality types of Jung. Other complementary models to Kolb's experiential model, such as Gardner's idea on multiple intelligences [3] and Dunn & Dunn's model [2] including external conditions, allows for a scientific analysis on separate aspects disregarded by Kolb.

References

- [1] Boyatzis, R.E. and Kolb, D.A. (2000). *Learning Skills Profile (LSP)*, McBer & Company, [On-line]. Available: <http://trgmcber.haygroup.com/Products/learning/lspus.htm> [2001-05-09]
- [2] Dunn, R. and Dunn, K. (1999). *The Complete Guide to the Learning Styles Inservice System*. USA: Allyn & Bacon
- [3] Gardner, H. (1983). *Frames of mind*. New York: Basic Books
- [4] Kolb, D. A. (1976). *Learning Style Inventory: Technical Manual*, Boston, MA: McBer and Company. In Hälsouniversitetet, Linköping. (1994). *Learning Style Inventory: A Self-description of Preferred Learning Modes*, abridged and excerpted from Kolb (1976).
- [5] Kolb, D. (1984). *Experiential learning: Experience as the source of learning and development*. CITY, New Jersey: Prentice-Hall.
- [6] More, A.J. (1987). Native Indian learning styles: A review for researchers and teachers. *Journal of American Indian Education*, 27 (1), 17-29, [On-line]. Available: <http://jaie.asu.edu/v27/V27S1nat.htm> [2001-04-19]
- [7] Nielsen, J. (1993). *Usability Engineering*. Academic Press.
- [8] Quintana, C., Eng, J., Carra, A., Wu, H-S., Soloway, E. (1999): *Symphony: A Case study in extending learner-centered design through process space analysis*, *Proceedings of CHI'99*, 473-480.

Carl Eneroth

Tel: +46 8 783 24 85

carl.eneroth@interactiveinstitute.se

[Back to contents](#)

ActiveMath, a Web-Based Learning Environment

System Overview

In Saarbrücken we are developing the *web-based, user-adaptive, interactive learning environment* ActiveMath. ActiveMath does not rely on predefined courses but offers the dynamic construction of courses. Currently, its major features are user-adapted content, sequencing, and presentation, support of active and explorative learning by mathematical services, support of teachers by information about their students, and a semantic encoding of content that is the basis for reusability.

For an ActiveMath session, the user chooses her learning goals from the concepts in the knowledge base. Then the systems recursively retrieves all concepts the goals depend upon and applies pedagogical rules and information from her user model to select the most appropriate content and form to present the content. This yields an instructional graph whose linearization is converted into HTML pages and presented to the user (see Figure 1).

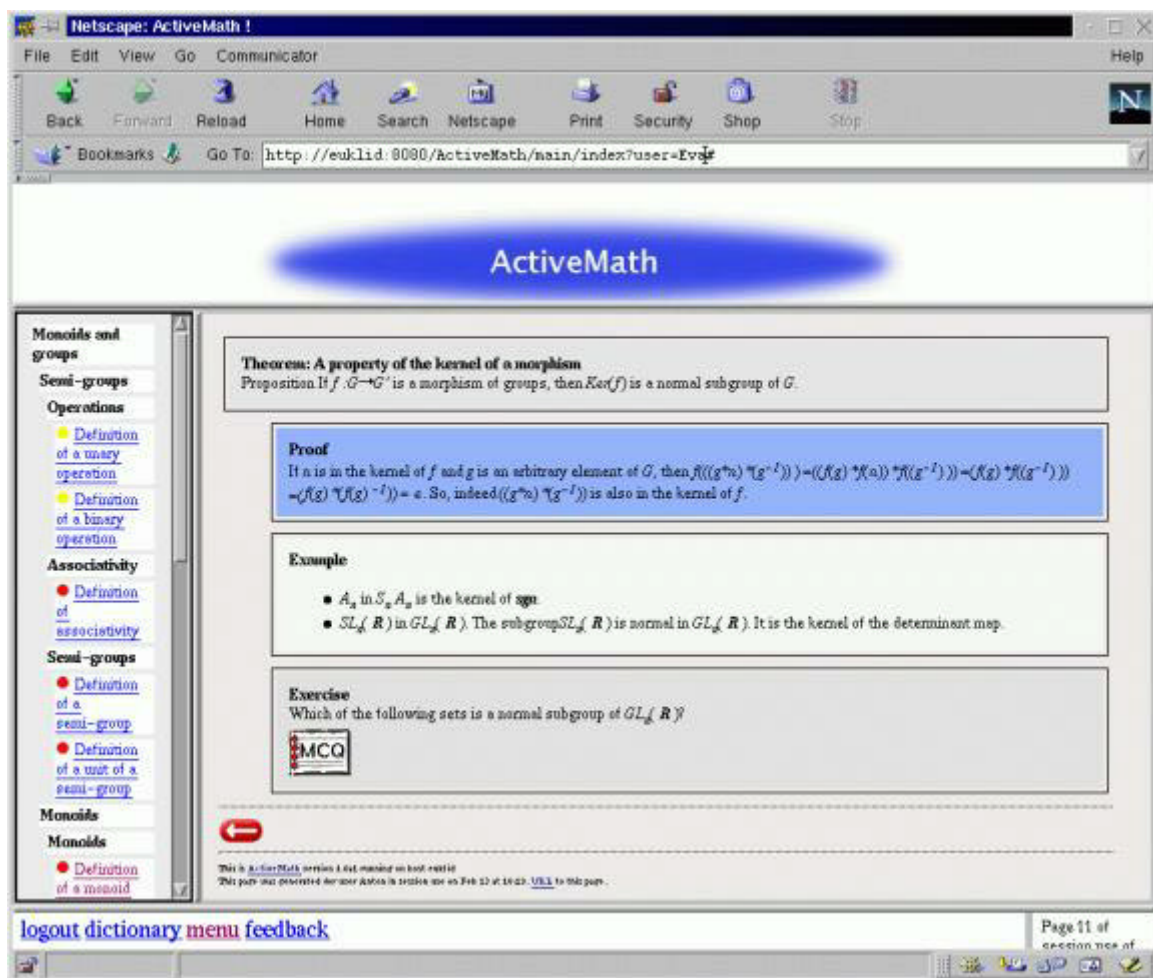


Figure 1. An ActiveMath session

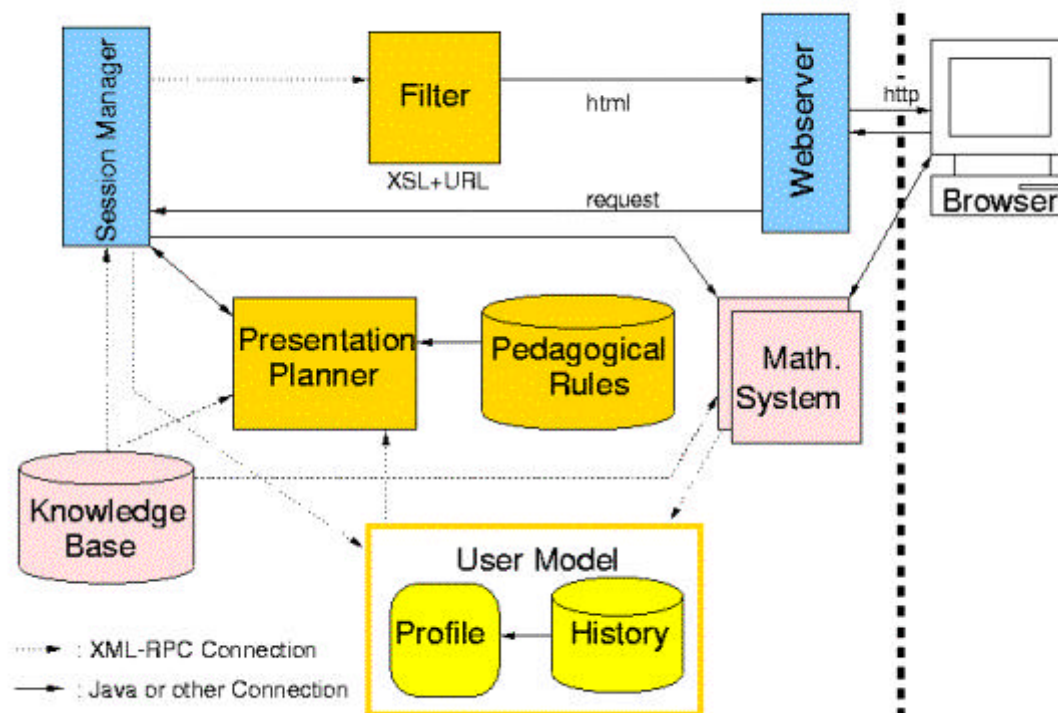


Figure 2. Architecture of ActiveMath

Figure 2 depicts the client-server web-architecture of ActiveMath. Currently, ActiveMath integrates the following components: a session manager, the knowledge base, a presentation planner, a user model, a pedagogical module, and mathematical services such as the proof planner Omega [3] and the Computer Algebra System (CAS) Maple. Requests of the user and (in the other direction) HTML-pages are communicated via a web-server to the session manager. The session manager stores the generated courses and translates URL requests into actions (e.g., the request for a new course about the topic *group*) that are passed to the responsible component. The presentation planner generates the personalized learning documents by requesting and processing information from the knowledge base, the user model, and the pedagogical module. Information about the user's actions, such as the success of solved exercises, is passed from the session manager to the user model where it is used for updating.

ActiveMath was designed with the goal in mind that the different components can be easily exchanged. For example, we will soon replace our simple table-based user model by an Bayesian net. Furthermore, our architecture makes it relatively easy to integrate new mathematical services and to design and implement interactive exercises.

Knowledge Representation

The knowledge base contains mathematical knowledge represented in the XML-based [OMDoc](#) [2] format. This allows a fine-grained representation of mathematical content by items such as definition, proof, theorems, motivation, etc. The items may include natural language formulations as well as formal objects. These formal objects (e.g., symbols) relate to actual mathematical objects, i.e., semantics. For instance, independent of whether the presentation is "plus" or "+", both presentations relate to the unique mathematical operation.

This semantics provides an *ontology* for the content of the course which is indispensable for a reuse of learning material and for a combination of material from different sources. In addition to the actual mathematical content, our knowledge representation contains meta-data for structures, dependencies, and pedagogical information which can be used for the dynamic generation of interactive documents.

Presentation Planning

The central component of ActiveMath is the presentation planner. It generates a personalized course in a three-stage process:

1. First, the content is retrieved from the knowledge base. Starting from the goal concepts chosen by the user, all concepts they depend upon and corresponding additional information (e.g., examples *for* a concept) are collected recursively. This grabbing process uses the *dependency* metadata information. The result of this retrieval is a collection of all concepts plus additional information about them that need to be known by the learner in order to understand the goal concepts.
2. Then pedagogical knowledge is applied. According to the information in the user model and the pedagogical module the collection of content items is processed and transformed into a personalized instructional graph of learning materials. ActiveMath employs pedagogical information represented in pedagogical rules (see Figure 3 for an example). Pedagogical rules are separately stored and can be employed to configure the system with pedagogical strategies and knowledge. This makes ActiveMath an excellent tool for experimenting with different configurations.
3. Finally, the instructional graph is linearized.

The result of the presentation planning is a linearized instructional graph whose nodes are *OMDoc* items. Filters transform this collection into HTML pages by XSL-transformations.

```
(defrule ChooseExerciseWithMathematicalService
  (need-exercise-for ?concept ?difficulty)
  (exercise (exercise-id ?id)
    (exercise-for ?concept)
    (difficulty ?difficulty)
    (uses-service ?service))
  (user-knows ?service)
  =>(assert (choose-exercise ?id)))
```

Figure 3. A pedagogical rule used by the presentation planner

The rule in Figure 3 chooses exercises that involve the use of a mathematical service. It determines that if there is an exercise needed for the concept *?concept* with difficulty *?difficulty* and if there is an exercise for the concept with the appropriate difficulty level that requires the mathematical service *?service* and the user knows how to use the service (indicated by *(user-knows ?service)*) then choose this exercise for presentation.

User Modeling

The user model consists of two subcomponents: the history that stores the data about user's actions, and the profile that stores the user's preferences and knowledge mastery data. When a learner registers, she can enter her estimation of her knowledge mastery values which will be constantly updated according to her actions.

The user model is inspectable and modifyable. The knowledge mastery assessment is represented by values for a subset of the competence features from Bloom's taxonomy [1] namely knowledge, comprehension, and application.

Conclusion and Further Work

We briefly described the current version of ActiveMath. The next versions will increasingly offer suggestions for next learning steps, feedback for exercises (in particular interactive exercises), and communication with the user. A [demonstration of ActiveMath](http://www.activemath.org/) is available at <http://www.activemath.org/>.

Bibliography

- [1] B. Bloom (1956). *Taxonomy of educational objectives: The classification of educational goals: Handbook I*, New York: Longmans.
- [2] M. Kohlhase (2001). OMDoc: Towards an internet standard of mathematical knowledge. In E. R. Lozano (Ed.) *Proc. of Artificial Intelligence and Symbolic Computation, AISC'2000*, LNAI, Berlin: Springer Verlag.
- [3] E. Melis and J. Siekmann (1999). Knowledge-based proof planning. *Artificial Intelligence*, 115 (1), 65-105.

Paul Libbrecht

paul@ags.uni-sb.de

Erica Melis

melis@ags.uni-sb.de

Carsten Ullrich

ullrich@ags.uni-sb.de

Saarland University
Postfach 151150
D-66041 Saarbrücken, Germany

and

DFKI GmbH
Stuhlsatzenhausweg 3
D-66123 Saarbrücken, Germany

[Back to contents](#)

Chat Services with a Voice

Abstract

This paper investigates techniques to provide additional dimension to student-student and student- teacher interaction at an academic institute. The technologies adopted also take into account the constraints posed by the Internet infrastructure that is widely used for this interaction and the affordability of the solution for the user community. The research adopts an approach that does not involve the user community to subscribe to higher bandwidth connections. The research involved a background study of current tools and software packages to integrate voice and image functionality into a chat service. The results presented in the paper indicate the possibility of using voice in addition to text chat. However, the results of the developed interface indicate that group discipline is needed to permit and facilitate a smooth voice and text chat conference. The concept involves the transmission of text over the Internet, which is suitably interfaced with text-to-speech software at the client end. This enables voice in addition to text at the receiving ends. In the future it is intended to investigate other applications of this technology.

Introduction

A previous study (Shukla, Sathu, Zhong, 2001) shows that adding voice and image helps by way of adding a human touch to the impersonal on-line text chat and thereby improves overall learning. Real audio over the Internet would be most desired. However, in view of the prevailing data communication environment this has significant shortcomings. Firstly, it requires committed bandwidth and other quality of service features that are not required for ordinary text chat. Secondly, even where high-speed networks with quality of service are available these are expensive and beyond the scope of the student community. Finally, even where such access is made available the overall connection may not provide the desired features from end-to-end. Alternative means to achieve additional dimensions in the learning process need to be investigated at the same time avoiding any of the above mentioned constraints.

The Approach

The research was based on the premise that communication access to the Internet would be required as for ordinary text chat. Hence this does not involve the user community to subscribe to higher bandwidth connections. To permit reasonable quality of audio conferencing, the authors opted to integrate a text to speech engine (TTS) into the chat conference services. Several existing voice application software were studied and trialed. Use of available voice application software was not adopted since these did not provide the desired flexibility for customisation. The first part of this research involved establishment of a development environment for the chat service. This was followed by the detailed interface development to provide a secure and user-friendly environment. Text input by a member (student) at the client end is sent to a chat server to be multicasted to other members of the group. When group participants receive the text, it is converted to audio in addition to the normal text message. The conversion of the text to audio at the receiver end provides a synthesised voice to accompany the usual text message.

Overview of Available Web Based Voice Application Software

There are a large number of application software that empower a user to convert text to speech. A significant number of these are available for downloading free of charge (trial version). The authors downloaded some of these and related voice application software and studied their features. These included: *SpeaksForItself*¹, *Digalo*², *ReadPlease*³, *Bell Labs Text-to-speech*⁴, and Microsoft products like *Speech SDK*, *Chant Kit* and *SR & TTS engine*⁵.

Implementation

Keeping in mind the study and the premise as mentioned in the above approach, the implementation was carried out under the following heads:

Application Architecture

A web-based application was developed using client server architecture. The full nature of the developed interface is achieved through embedding ActiveX controls into HTML. Active Server Page (ASP) provides the basic platform for this dynamic interaction.

Chat server

Chat server used for the project was set-up over the Windows NT 4.0 platform. Relevant Microsoft Internet Chat (MIC) Server was installed and configured from the Microsoft Exchange Services. An embedded chat client has been chosen to seamlessly integrate into the designed code for the web pages, which can tie into the MIC server.

Database

A database was created using the "MS access" software and was installed on the server. The Database can be easily upgraded and modified.

ILS

An ILS (Internet Locator Service) is also installed on the server for searching people across the Internet. The ILS directory service (email addresses) can be used to form a chat group.

User interface

The user-friendly interface for the client end comes up as depicted in the communication window below. The software for this is invoked by typing the specified URL. This brings up the first window (Fig.1) on the desktop. On entering the correct password the voice chat room with the images of the group members appears (Fig. 2). In the voice chat room, the name of the student initiating the session appears in the "username" text area. If the student/user would like to join in the voice chat room he/she can do so by clicking on "join the chat". There is a verbal announcement too to inform that he/she has joined the voice conference. From this time onwards student/user can hear and participate in audio conferencing.



Figure 1.



Figure 2.

To end the session the user/student has to click on "Leave Chat Room" button. An option is also available for private, one to one chat. It is possible to set up a voice option that suits a user. Accordingly one selects this through "set up your voice" button.

Conclusion

To achieve the additional dimensions of voice and image the authors have used Microsoft components since they are modular and could be modified to suit the project software design and development requirements. Further the authors intend to conduct field tests of the developed software. The field test will be conducted using students/staff. It is planned to use the feedback from the student to assess its success and modify it to

suit a broader spectrum of students and the community at large. The research group expects that the addition of voice and image components will improve the overall communication process between the students as compared to the plain text chat. The metrics for this would be the student and the staff feedback questionnaires.

Bibliography

- Shukla, R., Sathu, H. and Tang, Z. "Giving Voice to Discussion Boards" NACCQ 2001.
- SpeaksForItself. <http://www.speaksforitself.com/speaksforitself>
- Digalo. <http://www.digalo.com/>
- ReadPlease. <http://www.znet.com/>
- Bell Labs Text-to-speech. <http://www.bell-labs.com/projects/tts/voices.html/>
- Microsoft. <http://www.microsoft.com/download>

Hira Sathu

hsathu@unitec.ac.nz

Ranjana Shukla

rshukla@unitec.ac.nz

Zhong Tang

Ztang@unitec.ac.nz

UNITEC Institute of Technology
Private Bag 92-025
Auckland, New Zealand

[*Back to contents*](#)

Telemachus: A system for submission and assessment of students programs

Abstract

The software tool called Telemachus has been developed to test and grade students' programs. Students submit their programs via WWW and then the software compiles, tests and grades them and also generates statistical results. The aim of Telemachus is not only to grade the students' programs but more importantly to provide reliable performance data that could give a reasonable gauge of student knowledge and in this way contribute to teaching programming skills.

1. Introduction

In an introductory programming course students write programs in order to develop programming skills. The task of marking the solutions students produce in their programming assignments is laborious and error-prone. Thus, a number of researchers have been investigating the possibility of integrating technology into Computer Science examinations. Among others, Mason and Woit [7, 8] report their experiments from on-line programming examinations; Preston and Shackelford [9] describe a prototype for an on-line assessment software tool; Jackson and Usher [4] developed ASSYST a system for grading student programming exercises; Tinoco et al [11] develop QUIZIT a system for online evaluations in WWW-based courseware; Joy and Luck developed BOSS [5] a system for submission and assessment of students programming assignments.

Most of these systems interested us, but were inappropriate to be used in our courses for two main reasons: such a system must match exactly the requirements of the specific course on which it is intended for use and furthermore, the system must ensure compatibility with the University databases so that electronic marksheets can be integrated into the broader process of assessment administration. Thus we decided to develop a system called Telemachus handling not only submission, program testing and marking students' assignments but also providing reliable performance data that could give a reasonable gauge of student knowledge

Our system consists of two components: the first and the simplest one provides the means by which a student submits a program electronically for grading; the second component, which is used by the tutor, directs the assessment process. In the subsequent sections of the paper, we present the capabilities of our system and we show how technology can help the teaching process. We will show that this electronic marker does not only help in the process of testing and grading programs but can also give data for further didactic research. For example, using students' performance data we can detect their misconceptions and further we can pinpoint any particular notion that students may not have grasped fully.

2. Motivation

In the Department of Applied Informatics every year about 130 students are required to attend the CS1 and CS2 courses. These compulsory courses are offered during the first and the second semester and are comprised of a two-hour lecture and a two-hour laboratory session per week. In the laboratory session students solve some programming exercises with the instruction of a tutor. They are given a number of programming exercises as homework, whose solutions they have to submit in the next laboratory session. Almost all the exercises are small or medium sized programming problems and the average number of the programming assignments is 35-40 per course.

Up until now (i.e. before Telemachus), the following rudimentary examination procedure occurred. Since in our department there was only one assistant (like in most Universities of our country, assistants are a rarity) it was impossible to check manually all the students' programs. Thus, what happened was that at the end of the semester the tutor along with the assistant had to examine orally every student on a small number of programs (about 5 in all) as it was humanly impossible to check the entire listings (over 5000).

Obviously, this situation could not satisfy either our students or us and it was the weak point of both courses. We admit that accurate and meaningful assessment is vitally important for many reasons. First, it provides meaningful feedback to students and instructors; quality assessment informs students of their mistakes and successes and informs instructors of student knowledge. Second, it establishes confidence in the measurement of student performance; without accurate assessment, neither students nor instructors have a

reasonable gauge of student knowledge. Third, it provides instructors and administrators with the ability to perform quality control; collecting reliable performance data enables examination of the instructional process for courses. Finally, accurate assessment makes new educational research opportunities possible; customized courses, better use of class time, and student performance trend analysis are a few examples of possibilities [9, 3].

Thus, we decided to develop Telemachus, a system that tests and grades students' programs and also provides reliable performance data that will help us to detect potential students' misconceptions.

3. Description of the system

Telemachus consists of two main components: the one that students see and by which they submit programs electronically and access the results via WWW; and the second one that a tutor views and by which he/she can test and mark programs and obtain statistical results.

Students submit programs or access their results via WWW. Telemachus, in order to permit students' access to these operations, asks for their normal login name and password and then it permits access only to those who are students of the Department and have to attend CS1 and CS2 courses.

The second component, that of the tutor view, is composed of 6 main modules (see figure 3): Exercises, Students, Options, Reports, Marker, New Semester.

3.1 The Module "Exercises"

We can see, in figure 1, the form that handles the exercises' database. We have added until now 200 programming exercises into the database. We have categorized them into different topics (worksheets): basic statements, operations and types, selection structures, repetition structures, arrays, strings, records, files, pointers etc. Every year we choose a number of 35-40 exercises, among those included in the database, that are different from those of the previous year.

At the top of the form of figure 1, we see the buttons, which allow the tutor to add/ delete/ edit or find an exercise. In the section below on the left, we see the worksheet's number that the exercise refers to and the total number of exercises in this particular topic. On the right, the tutor gives the data concerning an exercise, such as: the exercise code (worksheet No, exercise No, Question No); the total number of data sets (input, output data sets); if the exercise is included in the marking process. In the middle of the form, the tutor writes the exercise. At the bottom of the form, the tutor gives some extra settings concerning the data sets.

3.2 The Module "Students"

The system handles students' database with a form similar to that of figure 1. The tutor can add/ delete/ find a student and can edit some elements.

3.3 The Module "Options"

Using the form of figure 2, the tutor sets the options that refer to the electronic marker. This form is divided into 3 sections: the left top section contains information about the exercises that will be marked. In the top right section the tutor can choose the compiler that will be used for compiling the program and by mouse clicking the button “Compile Now” Telemachus starts the compilation process. The results of the compilation process are recorded. The bottom section contains information about students.

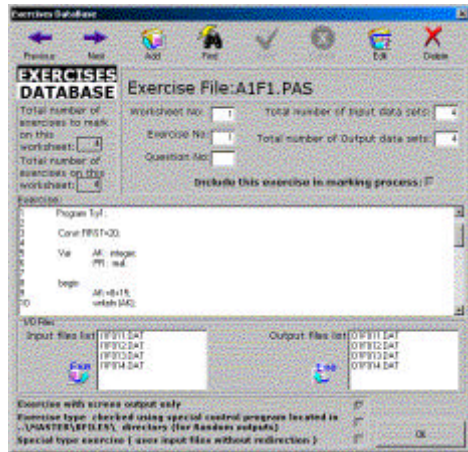


Figure 1.

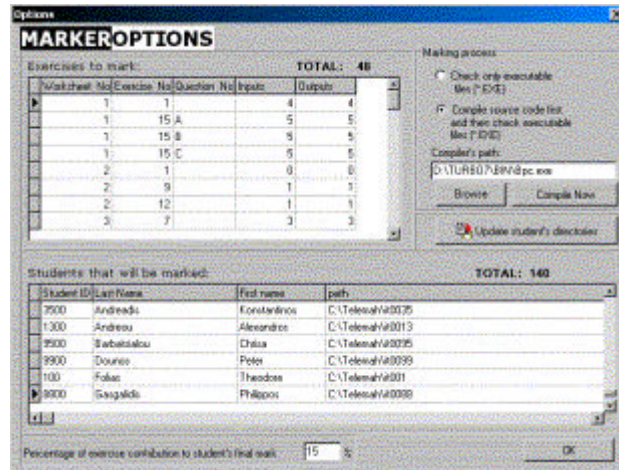


Figure 2.

3.4 The Module “New Semester”

The form titled “New Semester” shown in figure 3, initializes students’ database every new semester or for each different course. Running this module, the tutor can add all students to the database by giving the total number of students who attend the particular course and the year of their enrollment in the Department (student’s IDs in the University’s database are formed in this way). This module also creates new students’ directories and deletes students’ directories of the previous year.

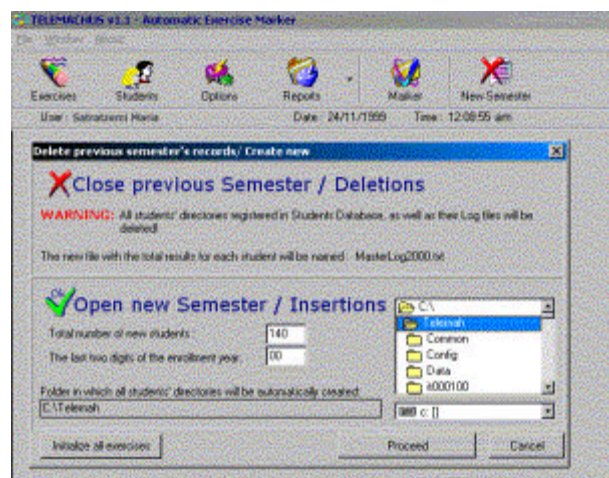


Figure 3.

3.5 The Module “Marker”

The button “Marker”, in figure 3, runs the module that checks the programs’ correctness. The executable code of a student’s program (which was previously generated by module “Options”) is run against the sets of test data. In the next step the module determines whether the program’s output is correct or not and marks the program. The checking approach is to match the student’s output produced by every set of the test data with the one produced previously by the system.

3.6 The Module “Reports”

Button “Reports” (figure 3) produces 3 different types of electronic reports. i) An extended report for every student, where he/she can see the following information: which programs were not successfully compiled; for every set of test data which program produced a correct or an incorrect output. ii) A report for all students with their grades. iii) A statistical report where the tutor can see for every exercise and every set of test data the rate of programs that were correct or had errors or were unsuccessfully compiled or did not give an output due to an infinite loop.

“File” menu (figure 3, main form) handles the produced reports (open, print a report etc).

4. Test Data Adequacy & Students’ Conceptions

As we have already mentioned in the introduction, not only does Telemachus help the tutor in grading students’ performance but also in providing useful data concerning students’ conceptions.

As it is known, many errors in students’ programs [1, 2, 10] have an element of chance and are thus unpredictable. There are some errors, however, which are more systematic and more persistent and since they are due to students’ misconceptions can be predicted. Students produce programs that are correct for most of the cases but when these programs are tested for some data sets they get incorrect results since students do not take into consideration all the cases. Telemachus validates students’ programs, running them against a number of predefined data sets rather than against random data sets so as to detect logical errors. We chose adequate data sets in such a way that a program with logical errors will produce an incorrect output or it will have an incorrect performance (infinite loop). Therefore, some data sets will cause students’ programs to give incorrect outputs whereas other data sets will cause correct outputs. Of course, the combination of incorrect and correct outputs does not guarantee the detection of a misconception; there are other types of errors that might be associated with the same combination of correct and incorrect outputs. Nevertheless, the combinations of the chosen data sets give valuable insight into students’ conceptions. Following, we give two examples in order to show the above.

4.1 Example 1

The first example is a series of programs which deal with the binary search [6]. We present 3 programs. The first is correct and the other two are incorrect. Table 1a gives the data sets and Table 1b summarizes the results.

No	Elements	Searching element
1	1 2 3 4	1
2	1 2 3 4	2
3	1 2 3 4	3
4	1 2 3 4	4
5	1 2 3 4	5

Table 1a. Data sets

<i>Programs</i>	<i>Results</i>
<pre> start:=1; fin:=n; found:=false; while ((not found) and (start <= fin)) do begin l:= (start+fin) div 2 ; if matrix[l]=element then found:=true else if matrix[l]<element then start:=l+1 else fin:=l-1 end; if found then writeln(l) else writeln('not found');</pre>	It gives correct output for all data sets
<pre> start:=1; fin:=n; found:=false; while ((not found) and (start <= fin)) do begin l:= (start+fin) div 2 ; if matrix[l]=element then found:=true else if matrix[l]<element then start:=l else fin:=l end; if found then writeln(l) else writeln('not found');</pre>	Infinity loop for data set No 4 and 5
<pre> start:=1; fin:=n; found:=false; while ((not found) and (start <= fin)) do begin l:= ((start+fin) div 2); if matrix[l]=element then found:=true else if matrix[l]<element then start:=l else fin:=l; if fin-start=1 then if matrix[start]=element then begin l:=start; found:=true end else begin l:=fin; found:=true end end; if found then writeln(l) else writeln('not found');</pre>	Incorrect answer for data set No 5

Table 1b. The programs and their performance

The second program gives, according to our observations, the most common error students make, while the third program shows the most usual modification that they make to the second program when students realize that it is incorrect.

4.2 Example 2

The second example shows a more trivial but equally frequent error in students' programs. The proposed problem was to write a program which erases from a string any leading and trailing blank characters. Obviously the use of the appropriate repetition structure gives correct output and vice versa. Table 2 summarizes the results given when the applied code is the following:

```
Readln(st);
repeat delete(st,1,1); until copy(st,1,1)<>#32;
repeat delete(st,length(st),1); until copy(st,length(st),1)<>#32;
```

Symbols used: S =any string without any leading and trailing blank characters, B= a string of blank characters

Given String	Results
S	Error: erases the first and the last character of S even though are not blank characters
ÂS	Error: Correctly erases the leading blanks B but also the last character of S even though it is not blank
SÂ	Error: Incorrectly erases the first character of S even though it is not blank but correctly erases the trailing blank characters B
ÂSÂ	Correct

Table 2.

5. Conclusions

Telemachus is very simple to use. Students use their normal email login names and passwords to log into the system, submit their programs electronically and access their report. Submitting programs electronically helps students save time, otherwise they would have to print their programs, which is a time consuming task. In addition in receiving a report on their submitted programs it gives them feedback on their mistakes and successes.

The help that Telemachus offers to the tutor is likewise invaluable. Besides the fact that the system saves the tutor from the laborious task of checking and marking students' programs manually, it also gives information that a human might have completely missed: it spots errors that could be difficult to pinpoint from the visual examination of listings. Furthermore, students' performance scores give the tutor the possibility to evaluate the success of a course. Finally, collecting reliable performance data over a long period of time the hypothesis concerning students' errors will empirically be confirmed.

6. Acknowledgments

The "Operational Program for Education and Vocational Training" of the Second Community Support Framework, EC, financially supports this work. We acknowledge the significant help given by Theodore Folias and Maria Myari during the development process.

References

- [1] Hoc J. M. Analysis of beginners' problem-solving strategies in programming, in Psychology of Computer Use, Green T.R.G., Payne S.J., van den Veer G.C. [eds], Academic Press, (1983), 143-158.
- [2] Hoc J., Green T., Samurcay R., Gilmore D., Psychology of Programming, Academic Press , (1990).
- [3] Hopkins K., Educational and Psychological measurement and Evaluation, Allyn & Bacon, Boston, (1998), 2-25.
- [4] Jackson D., Usher M., Grading Student Programs using ASSYST, In Proceedings of SIGCE'97, ACM, 335- 339.
- [5] Joy M., Luck M., Effective Electronic Marking for On-line Assessment, In Proceedings of ITiCSE'98, ACM, 134-138.
- [6] Lesuisse R. Some Lessons Drawn from the History of the Binary Search Algorithm, The Computer Journal, Vol. 26, n° 2, (1983), 154-163.
- [7] Mason D., Woit D., Integrating Technology into Computer Science Examinations, In Proceedings of SIGCE'98, ACM, 140-144.
- [8] Mason D., Woit D., Providing Mark-up and Feedback to Students with Online Marking, In Proceedings of SIGCE'99, ACM, 3-6.
- [9] Preston J., Shackelford R., Improving On-line Assessment: an Investigation of Existing Marking Methodologies, In Proceedings of ITiCSE'99 (Crakow Poland), ACM , New York, July 1999, 29-32.
- [10] Soloway E., Spohrer J., Studying the Novice Programmer, Lawrence Erlbaum Associates, 1989.
- [11] Tinoco L., Fox E., Barnette D, Online Evaluation in WWW-based Courseware, In Proceedings of SIGCE'97, ACM, 194-198

Maya Satratzemi

Dept. of Applied Informatics
University of Macedonia
156 Egnatia Str., P.O.Box 1591
54006 Thessaloniki
Greece
maya@uom.gr

Vassilios Dagdilelis

Department of Educational and Social Policy
University of Macedonia
dagdil@uom.gr

Georgios Evangelidis

Dept. of Applied Informatics
University of Macedonia
gevan@uom.gr

[Back to contents](#)

A Pattern Language for Architectures of Intelligent Tutors

Abstract. This paper introduces PLA_{IT}, a specific pattern language for architectures of intelligent tutors. A pattern language is a structured collection of interrelated patterns in a specific domain. PLA_{IT} is based on the idea of using patterns in the architectures of intelligent tutors, as well as on a number of patterns that have been discovered in the existing architectures of intelligent tutoring systems.

1. Introduction

In software engineering, patterns are attempts to describe successful solutions to common software problems [6]. Software patterns reflect common conceptual structures of these solutions, and can be applied over and over again when analyzing, designing, and producing applications in a particular context. Each pattern has a context in which it applies. When several related patterns are woven together, they form a *pattern language*. Pattern languages cover particular domains and disciplines, such as concurrency, distribution, organizational design, business and electronic commerce, human interface design and many more.

There are also patterns in intelligent tutoring systems (ITSs). Such patterns are, however, mostly implicitly present in ITSs. Patterns exist in architectures of ITSs, in the way learners learn from such systems, and in the way ITSs convey domain knowledge to the learners. This paper describes explicitly some patterns that exist in ITS architectures. The patterns described are all interrelated, and together represent the core of PLA_{IT}, a Pattern Language for Architectures of Intelligent Tutors.

2. The PLAiT Pattern Language

A recently conducted research in the domain of architectures of ITSs has shown that designers of different ITS architectures have used a number of common solutions in their designs. As a result of conducting that research, seven patterns in existing architectures of ITSs have been discovered. They have got the names *Inserted Layer*, *Top*, *Cascade*, *T-join*, *Cross*, *Multiplexer*, and *Store*. They are basic constituents of the PLAiT pattern language, which is still evolving. Five of these seven patterns are briefly described below.

2.1. Inserted Layer

This is the basic pattern in PLAiT. *Inserted Layer* is used in ITS architectures whenever a new functionality is needed in the ITS, or the architecture must be adapted to a new requirement, or some translation (negotiation) between the existing layers has to be performed. In order to better decouple the existing layers, it is a good idea to insert a new layer to implement and encapsulate the new functionality, as in Figure 1a.

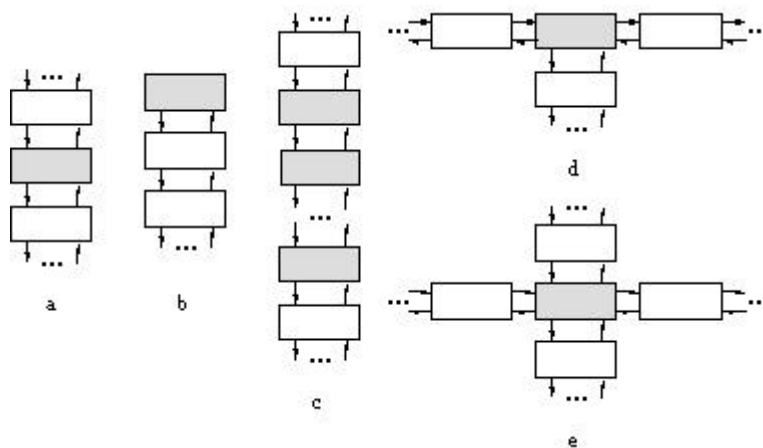


Figure 1. a) the *Inserted Layer* pattern b) the *Top* pattern c) the *Cascade* pattern d) the *T-join* pattern e) the *Cross* pattern

This pattern has been used in a dozen of existing ITS architectures. An example is introducing the believability layer in ITSs [1].

2.2. Top

The name *Top* reflects the idea of putting a new layer on top of other layers, Figure 1b. It is typically used when it is necessary to hide the complexity of the layered architecture from the outside world, as when the user or another system should have a specific or a well-defined "interface" for communication with a layered ITS. For example, A. Mitrovic has put the Constraint-Based Modeling (CBM) layer on top of the pedagogical module and the user interface in her SQL-Tutor, in order to overcome computational intractability of learner modeling [2].

2.3. Cascade

Here we have a cascade of new, distinct layers, hence the name *Cascade*, Figure 1c. This pattern is good to use when a new functionality is needed in a layered intelligent tutor, and it can be represented as a strict series of simpler modules. Also, sometimes a pipeline-style agent communication should be provided in an agent-based ITS, or a complex existing module should be broken into a sequence of simple functions (and there must be a strict order of function calls). In such cases, the solution is to represent each distinct functionality/agent/function as an individual module with strictly specified input and output, and make a cascade of such modules according to the observed strict processing sequence. An example is Knowledge Awareness (KA) agent of the Sharlok learning environment has its History observer, KA generator, KA filtering and KA monitoring connected in a cascade [3].

2.4. T-join

Figure 1d itself explains the name of this pattern. At a certain layer, a new functionality is needed but it doesn't only decouple existing layers - it has to communicate with another, "lateral" module. In fact, T-join can be viewed as an extension of Inserted Layer to accommodate communication with another module at the same level of abstraction in the hierarchy. For example, decomposing the learner's input in a Web-based ITS may require a new layer - a parser - between the user interface and the translator (see [5] for a concrete instance of this common problem).

2.5. Cross

This pattern has got its name from the fact that "vertical" and "horizontal" communication between some modules cross at a specific module (see Figure 1e). A new functionality is needed and it has to communicate with two distinct "lateral" modules. The Vincent animated pedagogical agent [4] is a good example of using *Cross* in an ITS architecture - its Action module is at the cross between the Mind module, the Body module, Micro Learning Environment, and the user interface (Vincent's appearance).

3. Conclusions

The knowledge of patterns can provide simple and elegant solutions to specific problems in ITS design. It can also help better organize the vocabulary of different learning and teaching styles and strategies, or the vocabulary of design elements and styles in ITS architectures. Patterns also have a great potential in generating and structuring explanations, hints, simulation, and other feedback that learners require from ITSs. They could represent the cores of solutions to analysis, design, architectural, instructional and other problems in ITSs that have been used more than once in different systems. Patterns can capture both static and dynamic structure of these solutions in a consistent and easily applied form. The PLAITS pattern language presented in this paper is the first attempt to define a specific pattern language in the domain of ITSs.

4. References

- [1] S. Abou-Jaoude and C. Frasson, Integrating a Believable Layer into Traditional ITS. In: S.P. Lajoie and M. Vivet (eds.), *Artificial Intelligence in Education*. ISBN: 90-5199-452-4. IOS Press, Amsterdam, 1999, pp. 315-324.
- [2] A. Mitrovic and S. Ohlsson, Evaluation of a Constraint-Based Tutor for a Database Language. *International Journal of Artificial Intelligence in Education* 10 (1999) 55-72.
- [3] H. Ogata and Y. Yano, Combining Knowledge Awareness and Information Filtering in an Open-ended Collaborative Learning Environment, *International Journal of Artificial Intelligence in Education* 11 (2000) to appear.
- [4] A. Paiva, I. Machado, and C. Martinho, Enriching Pedagogical Agents with Emotional Behavior: The Case of Vincent. Proceedings of the Workshop Animated and Personified Pedagogical Agents, Le Mans, France, July 1999, pp. 47-55.
- [5] S. Ritter, PAT Online: A Model-Tracing Tutor on the World-Wide Web. Proceedings of the Workshop Intelligent Educational Systems on the World Wide Web, Kobe Japan, August 1997, pp. 11-17.
- [6] D. Schmidt, M. Fayad, and R.E. Johnson, Software Patterns, *Communications of The ACM* 39 (October 1996) 37-39.

Vladan Devedzic

Department of Information Systems

FON - School of Business Administration

University of Belgrade, POB 52, Jove Ilica 154

11000 Belgrade, Yugoslavia

devedzic@galeb.etf.bg.ac.yu

devedzic@fon.fon.bg.ac.yu

[Back to contents](#)

Webbus, Tailormade Transport on the World Wide Web: An Educational Webgame

Abstract

Webbus is an educational webgame based on Letterbox, a nature-game originated in Dartmoor, Cornwall. In this nature-game people are searching for little boxes hidden in the hills, woods and mountains. Webbus is played on the World Wide Web. Its aim is, as in Letterbox, to wander but this time through the resources on the educational web. In the Webbus players are provided with a virtual landscape in which 'busroutes' give directions for wandering. Important fun & learning elements are: - solving a puzzle, which pieces are

delivered once hidden 'busstops' on the visited sites are found and - the opportunity to wander through the educational 'webscape'. While playing the surfer learns to browse, search & find the webbased information. Kernel of the game is a database on a Windows NT- server that holds information on the players, the busroutes, the hidden busstops and the status of the game for every participant. It also logs the traffic on the websites and the visitors' appreciation of the websites that take part in the game.

1. Rationale and aim of the game

Most novice and even advanced users of the (educational) World Wide Web find it difficult to find their way. Most teachers, very often fairly unexperienced users of the web, hesitate to apply content and applications in their daily classroom practice because of this lack of guidance and its potential risk of use. This reluctance also applies for parents of young children. In education there is also growing need for curriculum and/or subjectrelated structuring of potential educational content. Content providers on the other hand are in constant search for means to attract the attention of websurfers to give access to their information, services and products.

Webbus offers entertaining and safe routes along a selection of rich webbased educational content. The quality of the selected sites is assured by the editorial board of Kennisnet. By developing and maintaining the game Kennisnet draws attention to the content and services of educational suppliers and enriches the content of the educational portal Kennisnet.

2. About Kennisnet

Kennisnet is an initiative of the Dutch Ministry of Education. The project Kennisnet includes:

- the development of a national educational intranet; Kennisnet aims to provide inter- & intranet access to over 10.000 schools and almost 3,5 million users by 2002.
- a variety of intranet services and content-portals for children in schools & colleges, teachers, parents, managers and ICT-coordinators.

The Kennisnet organisation employs approximately 60 people occupied with the delivery of access and content. For the development of content and services Kennisnet contracts public and private content & service providers. The development of special services like the development of the Webbus game is usually outsourced to external partners such as Bit-IC.

3. Target group

The game is targeted at nine-year olds in education or at home. Together with Rowan Atkinson however we believe that there is 'a nine-year old in all of us'. The look and feel of the game and the interface are not specifically targetted. Webbus includes, and hopes to attract players of all ages and background.

4. The Game

Webbus is played on the World Wide Web. Its aim is, as in Letterbox, to wander but this time through the

resources on the educational web. Underlying metaphor for Webbus is the public transport system. The look & feel of the game interface is closely related to the bus transport system in one of the major Dutch cities. In this way the game hopes to attract the passenger to participate in an exciting 'magical mystery tour'. In the Webbus players are provided with a virtual landscape in which 'busroutes' give directions for wandering. To participate the surfer creates him-/herself a 'traffcard' and gets into the Web'bus'.

The busroute brings the surfer from one site to another. The route is thematic, e.g. museums, funsites, educational content, music, etc. Players are stimulated to proceed by offering them the possibility to solve a puzzle, which pieces are delivered once hidden 'busstops' on the visited sites are found. To complete the puzzle at least 7 pieces must be found. To give the player some idea where to look a hint on the hiding place of the busstop is given. Once a piece of the puzzle is found the player is asked to give a short evaluation of the website visited.

The main attraction however is the opportunity to wander safely through the educational 'landscape'. While playing the surfer learns to browse, search & find the webbased information. The play ends when the surfer found all the hidden elements on the visited sites or decides to stop. In some cases there is an extra reward on finding all pieces of a puzzle within a set time or contest. Gamers can skip busstops and change routes while playing. A separate window on the busroutes provides the players with data on progress and number of players (passengers!) in a busroute.

5. Participants

In the Webbus game three kinds of participants are active:

- the Webbus-editor, a member of the editorial office of Kennisnet, who designs the busroutes & invites the providers of websites to take part in the game. the player who gets in the webbus, wanders and plays the game
- the provider of the website who shows participation, hides the busstops in the site and gives a hint.

6. Technology

Webbus uses javascripts and SQL. Kernel of the game is a database on a Windows NT- server that holds information on the players, the busroutes, the hidden busstops and the status of the game for every participant. It also logs the traffic on the websites and the visitors' appreciation of the websites that take part in the game. The player is monitored while logged on and playing the game.

Webeditors design the webroutes and write the html-code for the webbus-pages. To participate the content providers have to embed a piece of html-code into their homepage and a small javascript in one of the other pages. Players log on with an identitycode (name) and a password and thus start a session. During the session there is constant contact with the database server. A session is ended when the player logs of or is inactive for a set period of time. Players can log on from any computer anywhere and start the game from any site that shows the Webbus-icon. A content provider can stop participating by removing the codes and scripts from the pages.

Together with the game an editor to create the routes and a tool to check the status of the participating websites was developed.

7. Added value & success

The main interest of the developers of the portal of Kennisnet is the exposure of its rich content and abundance of services to its users. The participating websites are invited or take part on their own request. The most interesting aspect for content-providers is the attention that Webbus gives to their content and that the Webbus-database keeps record of the routes that have been followed by the webbus-players.. Players enjoy the structured delivery of content and the fun of gaming.

Since its start Webbus has attracted a fast growing number of players amongst children, teachers and parents. A fast growing number of content providers have shown interest or participate. Webbus' success is illustrated best by its need for a dedicated editor to keep up with the demand for new routes and games.

8. Present status

It is developed for the Dutch educational web by the Ministry of Education, Culture & Science: bureau Kennisnet.nl in Zoetermeer and Bit-IC, Rotterdam in the Netherlands. At this moment themes are schoolsubject-related, bring the player along cultural sites likes museums or is related to an event. Some 2500 participants play the game in three different routes. The development of Webbus took 8 months and it was launched at the Educational Show in Utrecht, the Netherlands on the 22d of January 2001. The game can be played on the educational portal <http://www.Kennisnet.nl>.

Joop van Schie

Min. Of Education, Kennisnet, the Netherlands

j.vanschie@Kennisnet.nl

Mark Oostendorp

Bit-IC, the Netherlands

mark@bit-ic.nl

[Back to contents](#)

Web-based Adaptive Testing

Abstract

We describe the adaptive features of our system, which is a web-based assessment system for OOPS concepts. The system has a module called "Test Module" which generates the questions adaptively, based on the student's ability and also calculates the results. This system can be used as self-assessment kit for novices,

migrant-programmers from procedural languages to hone their object-oriented concept skills.

1 Introduction

The Web's explosion onto the world has changed the way of life in almost all areas especially in the educational field. Nowadays, educational institutions have started to deliver education through electronic i.e. using the Web as their medium of communication. There are lots of tools to support this "e-learning".

Adaptive hypermedia is one of the most effective disciplines to personalize E-learning. The techniques used here are quite similar or almost the same as used in Intelligent Tutoring Systems (ITS). [See 1 for a detailed account on ITS, past, present and future]

Using these techniques, several systems came into existence. Some of these systems are full-blown systems which can create an hyper-course [2],[3] while some systems are adaptive in presentation and adaptive navigation to make the learning more easier [4]. The systems mentioned above tailor the learning for each user based on their needs and abilities. Some other systems were developed to assess the student's knowledge based on Computer Adaptive Testing (CAT) and Item Response Theory (ITR). For a detailed account on adaptive testing, see [5] by Peter B and systems like SIETTE, TEST++ and ILESA [6][7][8].

Here, we describe the adaptive nature of our system to assess the student's knowledge based on his abilities to answer the questions, given in the form of quizzes (mainly, multiple-choice and true or false). The main goal of the system is to tailor the quiz session according to the ability of the student. This system can be used a self-paced training kit or self-assessment kit for would-be programmers, who want to migrate to Java.

2 How does the system adapt?

In this section, we describe how the system selects the question adaptively so that the student's knowledge is assessed properly. The architecture of the system is based on simple client-server architecture. A Java applet will be downloaded onto the client's machine, where the questions will be given to the students. Once the student answers the question, the applet send the reply back to the server to calculate the result and send the next relevant question. This is possible because of Java's powerful networking classes, JDBC capability and able to access any platforms.

The questions are selected from various topics set by the teacher/author, who is in charge of the system. There are 3 difficulty levels and the teacher sets these difficulty levels, based on their difficulty. Initially, the student is given a pre-test, which is nothing but basic concepts about programming and computers. These questions itself will be in random so as to eliminate cheating. The scores obtained from this pre-test will be used as the starting estimated ability value. This pre-test is given to give equal opportunities to students when they start the actual quiz session.

Now we talk about how the system adapt the questions. The policy of adapting the question is based on the logic of CAT, which is given below.

- All the items that have not yet been administered are evaluated to determine which will be the best one to administer next, given the currently estimated ability level.
- The "best" next item is administered and the student responds.
- A new ability level is computed based on the responses of the student
- Steps 1 to 3 are repeated until a stopping criterion is met.

This is the logic [9] our system is based on. Questions would be given randomly to eliminate cheating. The new ability level is computed based on the following equations [10]. This has been used in MANIC system during the quiz sessions to compute the new estimated ability level.

$$\text{NewValue} = \text{OldValue} + \text{OldValue} * (\text{level of Question} / 10) \quad \rightarrow 1$$

$$\text{NewValue} = \text{OldValue} + \text{OldValue} * (5 - \text{level of Question} / 10) \quad \rightarrow 2$$

The first equation is for correctly answered questions and the second one is for wrongly answered questions. This is because a student should not be punished too much if he could not answer a hard question. When the student starts the quiz, the OldValue will have the value scored by the student during pre-test. After that, for every question, this value will be updated, based on student's response. The quiz grade is in the boundary 0 to 1.

The algorithm we use is given below in the pictorial form.

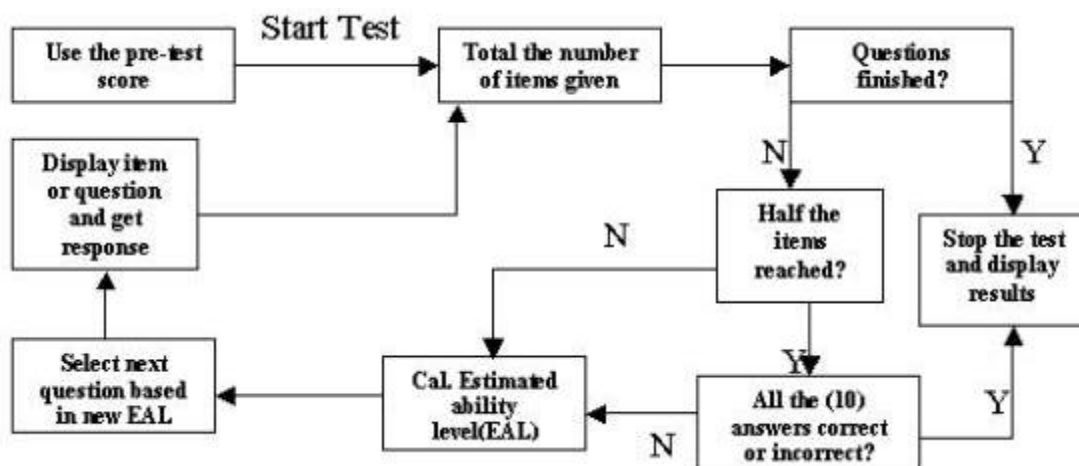


Figure 1. Pictorial representation of adaptive questioning

The parameters for ending the test would be either half of the questions answered by the student are right, in this case, he/she doesn't have to continue the test since the student would've reached the threshold value or the student has answered the maximum questions allowed by the system.

Finally, at the end of quiz, the results will be displayed on the screen. It will contain the student name, id, the total number of questions he/she has taken, number of right answers and wrong answers, whether the student has to revise the topic or not will be shown on the screen. These results will be stored in a user profile manager, which keeps track of all users, their ids, question ids, test ids etc.

References

1. Valerie J. Shute ,Joseph Psotka, D. Jonassen (Ed.), *Handbook of Research on Educational Communications and Technology*. Scholastic Publications, 1995.

2. P.Brusilovsky, J.Eklund and E.Schwarz. "Web-based education for all: A tool for developing adaptive courseware". *Computer Networks and ISDN systems*. Proceedings of Seventh Intl. WWW conference, 14-18 April 1998.
3. P. Brusilovsky. "Efficient techniques for Adaptive Hypermedia. C. Nicholas and J. Mayfield (eds), *Intelligent hypertext: Advanced techniques for the WWW*" Lecture Notes in CS, 1326, Berlin, Springer-Verlag, 12-30.
4. Specht, M., Weber, G., Heitmeyer, S., and Schöch, V., AST: adaptive WWW-courseware for statistics, in: P. Brusilovsky, J. Fink and J. Kay (Eds.), *Proceedings of Workshop "Adaptive Systems and User Modeling on the World Wide Web" at 6th International Conference on User Modeling*, UM97, Chia Laguna, Sardinia, Italy, June 2, 1997, pp. 91–95.
5. P.Brusilovsky and P.Miller. "Web-based testing for distance education. In: P. De Bra and J.Laggett (eds) *Proceedings of WebNet'99 World conference of WWW and the Internet*, Honolulu, Hi, Oct 24-30, 1999, AACE, pp. 149-154.
6. A.Rios, J.L. Perez de la Cruz, R.Conejo. "SIETTE: Intelligent evaluation system using tests for TeleEducation". Workshop "WWW-based Tutoring" at 4th Intl. Conference on ITS (ITS'98).
7. Maria Barra, Giuseppina Palmieri, Simon Napolitano, Vittorio Scarano and Luca Zitarosa. "Adaptive Testing by Test++. Lecture Notes in CS, Intl. Conference for Adaptive Hypermedia and Adaptive Web-based systems", AH200, Italy, Aug 28-30, pg.264-267.
8. J.M. Lopez, E. Millan, J.L. Perez de la Cruz and F. Triguero. "ILESA: Design and implementation of web-base tutoring tool for Linear programming problems".
9. Computer Adaptive Testing Tutorial (On-line article) <http://ericae.net/scripts/cat>
10. Mia K. Stern and B. P. Woolf (1998). "Curriculum sequencing in a Web-based tutor". Lecture Notes in Computer Science, 1452, 574-583.

T. M. Sri Krishna

Faculty of S&T
Assumption University
Bangkok-10240
Thailand
krishtm@yahoo.com

[Back to contents](#)

Context-bound evaluations of courseware in higher education

Abstract

This article reviews what different commentators have written about the evaluation of educational courseware, particularly the suggestion that more 'context-bound' evaluations are desirable. Speculation is made on the reasons for the lack of such evaluations. Contextual evaluations as a complementary and valuable alternative to traditional forms of evaluation of educational courseware, such as checklists. The case is made that more

effort could be productively devoted to context-bound evaluations. A case is made for using a checklist developed at Edith Cowan University specifically for assessing aspects of online pedagogical learning materials in higher education. This Checklist is intended to be more useful in indicating areas of the Online learning materials which are strong and areas those that may be deficient rather than to arrive at a numeric score. This form of checklist proposed is a valuable screening device to use before undertaking a context-bound evaluation of courseware.

Implications for web-based evaluations of courseware are considered. Quality of the instructional design remains an important consideration in evaluating courseware. Continuing comment and dissent is invited on the value of contextual evaluations of educational courseware to re-invigorate the debate over appropriate ways of evaluating educational courseware that will provide helpful information for higher education. Principles similar for the evaluating courseware, such as CD-ROMS, may be applied to evaluating on-line learning material.

Introduction

Evaluating of educational courseware is a vital activity in the use of information technology in education. Unfortunately, there is not enough agreement about what evaluation involves and who should carry out such evaluations. This paper considers the stage of development of the literature in this crucial area.

A useful beginning is the set of categories for evaluating courseware proposed by Hawkrigge (1990): descriptions, analyses, critiques and evaluations. A similar set of categories, as suggested by the OECD/CERI, was given by Cheung (1994); courseware description (a description of the main features of the courseware), courseware review (one individual's judgment about the courseware) and courseware evaluation (which must include actual use by the intended audience). Schibeci (1985) has been labelled this last process 'the acid test' in the courseware evaluation process because it is often considered to be a crucial test of the worth of educational courseware.

Analyses and critiques of computer courseware are commonly found in computer magazines. A new computer game, application, or utility will be described, analysed or critiqued by someone on the magazine's staff and this information helps in deciding whether to purchase a copy. Similarly, educational courseware may be described, or analysed, or critiqued. Is educational courseware evaluated? Much less often, it seems.

The promise of educational technology

In a provocative article, Ely (1997) posed: "Technology is the answer! But what is the question?" He wrote that we should ask of technology (hardware and courseware): *How* is it being used? By *whom*? For what *purpose*? *How often* and with what *result*? Additionally, Ely questions:

- How can we create the conditions for learners to become increasingly responsible for their own learning?
- How can we help learners to use the tools that are required for survival in a technological society?
- How can we 'humanise' technology in the service of all people?
- How can we help learners to ask the right question?

When conducting courseware evaluations it is crucial to ask ourselves similar questions. Despite the large investment in information technology, there is still scant evidence to establish the proposition that the new technologies have led to significant learning gains among students. This may be due to a number of reasons;

including lack of funding to support such research and a reticence of courseware 'evangelists' to subject their work to the rigours of academic scrutiny.

Perhaps this response is indicative of the unrealistic expectations that are often held for any new field. Also, it may be that the expectation that technology above all will lead to significant changes in learning is naïve. Contemporary economic considerations may well outweigh educational considerations. Rowley and Slack (1997) claimed that the "literature on CD-ROM interface evaluation and design is surprisingly sparse" (p. 11). In their view, this is a result of the disproportionate effort devoted to, "product development and the launch of new products to the market place" (p. 11). Alternatively, this may be a result of the relatively short-time CD-ROM courseware has been available.

Technologies which can be used to deliver education have suddenly become cheap, ubiquitous and pervasive (Lundin, 1993). As a consequence, the potential to deliver education locally and at a distance is poised to expand exponentially. Substantial opportunities exist for institutions prepared to adopt a mixed mode of delivering education based on models of open learning (Hamer, 1993).

Evaluation

Educational technology texts invariably assume that faculty should conduct their own reviews and provide checklists to help them conduct such courseware reviews (to use the OECD/CERI term). For example, Sharp (1996, p. 156) provided a checklist with these headings; courseware type, hardware, courseware contents, instructional design, courseware appearance, ease of use, consumer value and support. Checklists and frameworks are two common approaches to predictive evaluation which have been criticised. Predictive evaluation has been defined by Squires and McDougall (1996, p. 147) as "the assessment of the quality and potential of a software application before it is used with students". Rowley and Slack (1997) reviewed checklist approaches to evaluation and developed an integrated list from these two lists.

McDougall and Squires (1995) have reviewed checklists for courseware evaluation and concluded that there is a role for checklist, such as the one given by Rowley and Slack, in the formative evaluation of courseware, not in its selection. McDougall and Squires (1995) also noted a confusion between *evaluation* and *selection*. These two types of reviews serve two very different but equally important purposes. Tergan (1998) has also critically reviewed checklists and suggested ways in which they can be used.

A number of authors have suggested procedures for educators to follow as they decide what courseware to purchase. For example, Roblyer, Edwards and Havrilvk (1997) recommend that faculty adopt the following sequence in selecting courseware: identify a need; locate titles; conduct 'hands-on' reviews; and collect students reviews' (ie. field test the courseware). However, it must be recognised that the teacher's intent and approach are critical. Faculty perceptions are a highly variable aspect of the context of evaluating courseware.

They also provide a list of criteria for courseware evaluation. Cheung (1994) suggested that faculty conduct a five-stage process for evaluating courseware. While these suggestions may be ideal, it is debatable if many faculty have the time for such a laborious procedure. Cheung also suggested that faculty "should be given the time and opportunity to prepare themselves to carry out the evaluation" (p. 195). Certainly, faculty are crucial in courseware evaluations, which include a test of the courseware with the intended student audience.

What does 'evaluation' entail? Levine (1996) suggested that there are four aspects of evaluation, which apply also to 'courseware' evaluation: "(a) courseware design; (b) the effects on courseware selection; (c) courseware implementation; and (d) the formation of a theoretical base for courseware use, including its effects on cognitive, social, and instructional process" (p. 261). Levine also noted the following about

courseware evaluation as a field of study: “Although a vast amount of literature exists on this subject, a basis of agreed criteria is still lacking” (p. 261). Among the approaches to courseware evaluation listed are: “a checklist, an analytical (open-ended) review, a panel evaluation, an interview, automated measures, observations and, experimentation” (p. 264). Levine suggested that “Observation of courseware in use can provide the basis for valid and effective evaluation for either formative or summative purposes” (p. 265). Regarding experimentation, he wrote that it is 'controversial' in courseware evaluation, but that it is “the only methodology that checks the origin of the effects rather than the effects themselves” (p. 265). Finally, he concluded his review of courseware evaluation as follows:

Effective evaluation produces judgments that are context-related; regardless of the methodology used, it takes into account the pedagogical nature of courseware, the classroom milieu the desired goals and the type of usage. Such evaluation is driven by educational needs, as reflected in the curricular, and by learning theories, rather than by technology (Levine, 1966, pp. 266-7).

Squires and McDougall (1996) noted recent acceptance by many educators of a 'situated' view of learning, in which what is learned and how it is learned are not separated. They believed this view has important consequences for educational courseware evaluation. Importantly, they distinguish between predictive and interpretative evaluations:

Predictive evaluation of software is the assessment of the quality and potential of a software application before it is used with students . . . Interpretative evaluation is concerned with assessing the observed use of an application by students. By definition, interpretative evaluation is conducted in context (p. 147).

Squires and McDougall (1996) offer an approach they label the 'Perspectives Interactions Paradigm':

a comprehensive framework for thinking about educational software and moves ... toward more educational uses such as learning processes, classroom activities, teacher roles curriculum issues, and student responsibility for learning (p. 155).

Squires and McDougall's (1996) approach considers interaction between pairs of perspectives among faculty, student(s) and designs. They illustrate the approach, which they claim can offer an effective predictive evaluation tool, with an evaluation of the CD-ROM *From Alice to Ocean*. This exemplifies the potential in applying the approach to other courseware.

Implications for 'on-line learning'

What is learnt evaluating the various forms of courseware may translate into useful approaches to evaluating on-line learning material. Technologically Mediated Learning (Hosie, 1993) offers cost effective interactive learning for students over long distances while maintaining high quality outcomes (Lundin, 1993). Creer (1996: 20) predicted that, "the virtual classroom is here and is set for a rush".

Of all the various permutations of Technologically Mediated Learning the Internet has the greatest capacity to integrate the existing technologies and emerging technologies. The Internet has captured the imagination of the community and educators alike. As the fastest growing form of information exchange within our society, the Internet allows multimedia technologies to be available to anyone, virtually anywhere. The Internet has profound implications for promotion, delivery and administration of international education.

Enter the Internet

Students at all levels of education already make extensive use of the Internet to assist learning. The Internet is accessible in around 150 countries. It provides the hardware support for many technologies including - e-mail, the World Wide Web, file transfers and remote systems access, which can be used for entertainment, business, social and educational applications. While it is not possible to state, with any great accuracy, the number of world-wide users who access the Internet, estimates place it between 30 and 50m people with the expectation that *these numbers will double every six months*.

Key finding Daniel Bieler and Iain Stevenson's (1998) report; *Internet Market Forecasts: Global Internet Growth 1998 – 2005*, indicates that:

- the US market will saturate after 2002 and most continuing growth will derive from developed Asia and Western Europe. Aggregate traffic will grow to over 6 Tbits/s by the year 2005
- there will be major capacity growth but no redistribution of traffic patterns. The most important routes in 1998 will still be the most important in 2005
- the international Internet backbone is on the point of collapse. If new fibre build across the Atlantic does not progress to plan there will be a chronic shortage of capacity to the US. This will result in poor service that could threaten the development of the Internet on a global basis
- the Japanese market will dominate the Internet in Asia because it has a large number of highly educated users. Although China and India will become major markets as infrastructure and their economies develop they will not be significant in the period to 2005
- value added service provision is key to long term survival online. Internet access revenues from dial up connections will start to decline in per-capita and real terms despite significant growth in subscriber numbers. Revenues from permanent connections will rise initially, but will also encounter a decline in per-capita revenues around 2004.

([http://www.gsmdata.com/repovum2.htm#Key findings](http://www.gsmdata.com/repovum2.htm#Key_findings))

Regardless of the precision of these guesstimates, a colossal number of people have access to the Internet worldwide and this is growing exponentially, as is the perceived demand for higher education. A new 'megatrend' has emerged. An alternative prognosis for where education and global learning is heading is worth considering (Logan & Pritchard, 1996). There is every indication that the impact of information exchange on the Internet will rapidly translate into a viable international distance education service. Scores of institutions are planning to, or already offer, education via the Internet.

Implications of the MIT *OpenCourseWare* (MITOCW) initiative

Of all the recent innovations in educational technology, the Massachusetts Institute of Technology's decision to make the materials for most of its courses, freely available on the Internet, over the next 10 years, is the most impressive in scope. By the 2002, MIT claim that materials for more than 500 courses would be available on the *OpenCourseWare* site (<http://web.mit.edu/newsoffice/nr/2001/ocw.html>)

MIT sees a variety of benefits coming from the MIT *OpenCourseWare* project:

- Institutions around the world could make direct use of the MIT *OpenCourseWare* materials as references and sources for curriculum development. These materials might be of particular value in developing countries that are trying to expand their higher education systems rapidly.
- Individual learners could draw upon the materials for self-study or supplementary use.
- The MIT *OpenCourseWare* infrastructure could serve as a model for other institutions that choose to make similar content open and available.

- Over time, if other universities adopt this model, a vast collection of educational resources will develop and facilitate widespread exchange of ideas about innovative ways to use those resources in teaching and learning.
- MIT *OpenCourseWare* will serve as a common repository of information and channel of intellectual activity that can stimulate educational innovation and cross-disciplinary educational ventures.

<http://web.mit.edu/newsoffice/nr/2001/ocw.html>

Making up-to-date educational content widely available will make possible a new style of national and global collaboration in education. *OpenCourseWare* will make it possible to rapidly disseminate new knowledge and educational content in a wide range of fields. This is particularly appropriate for assisting research universities, where ideas and information need to be disseminated quickly from the laboratory into the educational program before they are published in textbooks.

On the surface the *OpenCourseWare* initiative appears counter-intuitive to market dynamics, by widening access to information and inspiring others to participate. Proponents of the *OpenCourseWare* claim that this initiative will make it possible to focus more closely on the actual process of teaching, on the interactions between faculty and students that are the nub of learning. This initiative has the potential set in motion innovations in teaching. When students are able to acquire course content on the Internet, faculty will be able to pay more attention to the actual process of teaching. *OpenCourseWare* will facilitate faculty to concentrate on using lectures or the laboratory time to enhance learning, with less time devoted to conveying information.

OpenCourseWare is poised to make a major global impact on learning. Whist important, the provision of information (content) of itself does not constitute an educational experience. However, MIT are not providing an education on the Internet only the core materials from the infrastructure that integral to this process of learning. Real education requires interaction, the interaction that is part of American teaching. *OpenCourseWare* will only be as useful as the instructional design that creates an educational experience for students. Over time, the provision of knowledge in a timely and information via the Internet, when coupled with high quality learning experiences, has the potential to the quality of learning around the globe.

Taylor (2001) has identified the phases that have evolved through four generations distance education, beginning with the Correspondence Model based on print technology. Next came the Multi-media Model based on print, audio and video technologies. This was followed by the Telelearning Model, based on applications of telecommunications technologies to provide opportunities for synchronous communication represented by the fourth generation.

As Taylor (2001) observes fifth generation of distance education (The term “Distance Education” is almost a synonym for online learning in the USA.) is already emerging even though many universities are just beginning to implement fourth generation initiatives. Taylor has conceived the fifth generation as a derivation of the fourth generation except that it aims to more strategically incorporate automated features of Web to increase service levels and efficiencies. Taylor argued that the fifth generation has the potential to radically change the cost of providing online educational services (eg. access to institutional processes and online tuition through

... the development and implementation of: automated courseware production systems, automated pedagogical advice systems, and automated business systems, the fifth generation of distance education has the potential to deliver a quantum leap in economies of scale and associated cost-effectiveness. Further, effective implementation of fifth generation distance education technology is likely not only to transform distance education, but also to transform the experience of on campus students (Taylor 2001, p. 4).

As universities beginning to implement fourth generation distance education initiatives, Taylor (2001) argued

that the fifth generation Flexible Learning Model based on online delivery via the Internet is already emerging based on the further exploitation of new technologies. According to Taylor (2001) “The fifth generation of distance education is essentially a derivation of the fourth generation, which aims to capitalize on the features of the Internet and the Web” (p.2). Certain features of the previous four generations of distance education are inherent. Fifth generation Intelligent Flexible Learning Model into Taylor’s conceptual framework. Characteristics of the various models of distance education that are relevant to the quality of teaching and learning (Taylor, 2001) are summarized in Table 1.1, together with indicators of institutional variable costs (Taylor, Kemp & Burgess, 1993).

Models of Distance Education and Associated Delivery Technologies	Characteristics of Delivery Technologies					
	Flexibility			Highly Refined Materials	Advanced Interactive Delivery	Institutional Variable Costs Approaching Zero
	Time	Place	Pace			
FIRST GENERATION -						
The Correspondence Model						
Print	Yes	Yes	Yes	Yes	No	No
SECOND GENERATION -						
The Multi-media Model						
Print	Yes	Yes	Yes	Yes	No	No
Audiotape	Yes	Yes	Yes	Yes	No	No
Videotape	Yes	Yes	Yes	Yes	No	No
Computer-based learning (eg CML/CAL/IMM)	Yes	Yes	Yes	Yes	Yes	No
Interactive video (disk and tape)	Yes	Yes	Yes	Yes	Yes	No
THIRD GENERATION -						
The Telelearning Model						
Audioteleconferencing	No	No	No	No	Yes	No
Videoconferencing	No	No	No	No	Yes	No
Audiographic Communication	No	No	No	Yes	Yes	No
Broadcast TV/Radio and Audioteleconferencing	No	No	No	Yes	Yes	No
FOURTH GENERATION -						

The Flexible Learning Model						
Interactive multimedia (IMM) online	Yes	Yes	Yes	Yes	Yes	Yes
Internet-based access to WWW resources	Yes	Yes	Yes	Yes	Yes	Yes
Computer mediated communication	Yes	Yes	Yes	Yes	Yes	No
FIFTH GENERATION -						
The Intelligent Flexible Learning Model						
Interactive multimedia (IMM) online	Yes	Yes	Yes	Yes	Yes	Yes
Internet-based access to WWW resources	Yes	Yes	Yes	Yes	Yes	Yes
Computer mediated communication, using automated response systems	Yes	Yes	Yes	Yes	Yes	Yes
Campus portal access to institutional processes and resources	Yes	Yes	Yes	Yes	Yes	Yes

Table 1.1. Models of distance education - a conceptual framework (Taylor, 2001)

Prior to the advent of online delivery, variable costs had a tendency to increase or decrease directly with changes in the volume of activity. For example, in second generation distance education delivery, the distribution of packages of instructional materials, such as printed study guides, audiotapes, videotapes and the like varied in proportionate to the number of students.

Taylor (2001) argued that fifth generation distance education has the potential to decrease significantly the costs associated with providing access to online tuition. the Fifth generation of distance education has the potential to deliver a quantum leap in economies of scale and associated cost-effectiveness of automated courseware production systems, pedagogical advice systems, when business systems are effectively implemented. Fifth generation distance education technology has the potential to transform flexibly delivered learning for students both on campus and distance education. The costs and associated benefits education need to be incorporated into any evaluation of fifth generation flexibly delivered education. Flexible delivery, for the purposes of this document, the degree to which learning is 'flexible' depends upon:

- flexible delivery - the range of *course delivery* options available; and,
- the degree of student choice (tailored content, learning style, modularity, etc) built in to the *course design*;

from the learner's point of view, flexible learning is the extent to which they can choose to study what, where, how and when they prefer.

What is online?

Until recently there were no common definitions for describing the different types of “onlineness”. Definitions offered by institutions or individuals offered little basis for comparison or usefulness for articulating strategy and developing subsequent business processes.

The Australian Federal Department of Employment Training and Youth Affairs’s latest version of their survey instrument describes three modes of online - *Web Supplemented*, *Web Dependent* and *Fully Online*. These modes are presented in Table 1. 2.

Mode	Definition
Mode A: Web Supplemented	Participation online is optional for the student. Enrolled students can access information on units of study that is additional to that available in the university’s calendar or handbook. The information may include course descriptions and study guides, examination information, assessment overview, reading lists and other online learning resources. The information is used to supplement traditional forms of delivery.
Mode B: Web Dependent	Participation online for each activity described in (i), (ii) or (iii) below is a compulsory requirement of participation although some face-to-face component is retained (i) students must use the web to interact with the education content necessary for study (ii) students must use the web to communicate with staff and/or other students (iii) students must use the web both to interact with content and to communicate with staff and/or other students
Mode C: Fully Online	There is no face-to-face component. All interactions with staff and students, education content, learning activities, assessment and support services are integrated and delivered online.

Table 1.2. DETYA Online Unit Definitions

The definitions of these modes provide Australian universities with a useful starting point for specifying, articulating and tracking its online strategies will simplify benchmarking with other institutions. A major limitation of these definitions are that they fail to address quality issues (such as teaching and learning approaches) or identify modes that utilise the web other than those complementing face-to-face delivery (ie. Mode B). Alternative strategies to complement the web such as print-based materials, CD-Rom, or teleconferencing (video and audio) need to be recognised past investment and future flexible delivery plans.

Pedagogies used in quality learning materials

The following attributes and assessment framework were developed Edith Cowan University Quality Online Working Group (March, 2001) describe the essential components of quality online learning materials. Edith Cowan University is committed to constructivist approach to learning for both face-to-face as well as online learning (Steffe & Gale, 1995).. The various attributes described in this framework are therefore intended to reflect the constructivist learning principles for resources and learning support as detailed in the University's Teaching and Learning Management Plan

(http://www.ecu.edu.au:80/secretariat/acbdmins/doc/T_Learning_Mngmnt_Plan_V10_rtf.doc)

Learning	Description	Examples
Authentic tasks	The learning activities involve tasks that reflect the way in which the knowledge will be used in real life settings	Problem-based learning activities using real-life contexts; Learning tasks based in workplace settings Tasks are complex and sustained
Opportunities for collaboration	Students collaborate to create products that could not be produced individually	Tasks are set that require students to collaborate meaningfully Peer-evaluation, industry mentors Buddy systems employed to connect learners
Learner-centred environments	There is a focus on student learning rather than teaching	Teacher's role is one of coach and facilitator Inquiry and problem-based learning tasks Activities support and develop students' metacognitive skills
Engaging	Learning environments and tasks challenge learners	Interesting, complex problems and activities rather than decontextualised facts Activities arouse students' curiosity and interests Activities and assessments linked to learners' own experiences
Meaningful assessments	Authentic and integrated assessment is used to evaluate students' achievement	Assessment is integrated with activities rather than separate tests Opportunity to present polished products rather than simple drafts Opportunities exist for students and their teachers to provide comment and feedback on academic endeavour

Table 1.3. Pedagogies used in quality learning materials

Learning	Description	Examples
Accessibility	Resources are organised in ways that make them easily accessed and located.	Resources are separate from learning tasks Intuitive and clear organisational strategies Resources are accessible in a nonlinear format
Currency	The age of resources are appropriate to the subject matter	Resources should where possible be current and based on regular literature reviews by lecturer Seminal works should not, however, be removed on the basis of age Use of primary resources is made wherever possible
Richness	Resources reflect a rich variety of perspectives	Resources should represent a variety of views (including conflicting views) to allow students the opportunity to assess the merit of arguments Resources provide for a range of perspectives Media are used to enrich data sources
Strong use of the media	Media is suitable for the purpose intended	A variety of media is used where appropriate Book on screen approach should be avoided Equally, elaborate multimedia should be avoided when a simple diagram would be suitable
Inclusivity	Materials demonstrate cultural, gender inclusively	Resources include a variety of cultural perspectives where possible Resources avoid gender and culturally exclusive terms Separation of local and generic content

Table 1.4. Resources in quality learning materials

Learning	Description	Examples
Reliable and robust interface	The materials are accurate and error free in their operation	Site is accessed reliably navigation and orientation is seamless Many forms of online support for learners
Clear goals, directions and learning plans	Unit information and expectation of student roles are clear	Students can find information on the website about the unit and its requirements Unit structure makes explicit relationships between learning outcomes, resources, activities and assessments Instructions clearly placed and always available
Appropriate bandwidth demands	The materials are accessible without lengthy delays	Graphics and other elements checked for download times. Delivery formats employ strategies to optimise download times
Equity and accessibility	Unit materials and activities are accessible and available to disabled students	Websites are accessible to disabled students Course requirements and resourcing made explicit to students ahead of the course Students are not hampered by firewalls or geographic sensitive restrictions
Appropriate corporate style	Units adopt a corporate style for websites to ensure minimum quality of presentation of online courses	Layout and presentation should incorporate common elements on the unit homepage reflecting a corporate style (The corporate style should enhance rather than dictate a pedagogical approach) Fonts, resolution etc should conform to the corporate style where possible, but alternatives should be possible when needed

Table 1.5. Delivery strategies in quality learning materials

Edith Cowan University online learning quality checklist

Based on the attributes detailed above, checklists have been developed for use in to evaluating online learning materials to determine areas of strength and weakness. The checklist is not intended to be used to deliver a numeric score that provides a definitive evaluation of the courseware. Rather, this checklist is intended to reflect and to indicate areas of the materials that are pedagogically strong and others that may need further attention. Other aspects of a managerial nature such as password protection, copyright and the like are not considered as these have already adequately dealt with elsewhere. These are seen as matters for others to judge and evaluate.

Pedagogies	never	sometimes	always
Authentic tasks The learning activities involve tasks and contexts that reflect the way in which the knowledge will be used in real life settings			
Opportunities for collaboration The environment encourages and requires students to collaborate to create products that could not be produced individually			
Learner-centred environments There is a focus on activities that provide degrees of freedom, decision-making reflection and self-regulation			
Engaging The learning activities challenge learners and provide some form of encouragement and motivation to support the engagement			
Meaningful assessments Authentic and integrated assessment is used to evaluate students' achievement			
Resources			
Accessibility The resources are organised in ways that make them easily accessed and located			
Currency The age of resources are appropriate to the subject matter			
Richness The resources reflect a rich variety of perspectives			
Strong use of the media The materials use the various media in appropriate ways			
Inclusivity The materials demonstrate cultural and gender inclusivity			
Delivery strategies			
Reliable and robust interface The materials are accurate and error free in their operation across all platforms and browsers			
Clear goals, directions and learning plans Unit information and expectation of student roles are clear			
Appropriate bandwidth demands			

The materials download without lengthy delays			
Equity and accessibility			
The unit materials and activities are considerate of students with visual impairment and physical disabilities			
Appropriate corporate style			
The materials use a style that is compatible with ECU policy and guidelines			

Table 1.6. Online learning quality checklist

(Source: Framework developed BY Edith Cowan University Quality Online Working Group March 2001)

Instructional Design: The Foundation of Online Learning

Without doubt, the quality of the Instructional design is critical in the evaluation of courseware. Quality of instructional design needs to be a pre-eminent consideration in any evaluation of courseware. A significant consideration, which is given scant attention by the literature, is that many learners prefer some form of human interaction in learning tasks, while others do not. Clearly, individual learner preferences will be an important factor in any design and subsequent evaluation considerations for multimedia. Vadas (1989) also notes a significant percentage of students also prefer solitary learning circumstances. Accommodating these individual preferences is possible with learning involving multimedia and needs to be considered by instructional designers as well as those seeking to incorporate multimedia into learning programs. These need to be designed to accommodate individual learning styles.

Traditional use of multimedia has emphasised images and video although speech recognition is already well established within the technology. According to Schank (1994) the future challenge for multimedia design lies in developing programs that actively engage the user. New courseware designs, which do not rigidly structure learner responses, need to be explored to complement this medium of instruction. Successful adoption of multimedia depends on developing quality courseware that takes into account the unique attributes of the technology. Multimedia designs are only just beginning to exploit the capacity of the technology.

Also it is necessary to ensure that non-courseware-based courses involve learning by doing. Multimedia information system design is still a new field and there is no agreed model of how to proceed (Grosky, 1994).

An Online Reality Check

Considerable expectation, hope (and hype) have also surrounded this latest manifestation of educational technology. What is the reality? Scanlon's (1997) review of on-line learning in science highlighted the issue of priorities: some faculty might prefer basic resources (books, adequate hygiene) to Internet connections. Nevertheless, she raised a number of issues for those evaluating on-line learning to consider. Among these were the following:

- reliability of the network
- scalability: small groups of learners, or larger group?
- Bandwidth

- integration, technology and curricular
- pedagogy
- evaluation

Scanlon noted that few studies she reviewed did, “more than describe new and innovative uses of technology” (p. 84), and argued for “more critical accounts” of such endeavours.

Given the large investment in educational technology in universities need to consider seriously whether, as Oppenheimer (1997) claimed they have deluded themselves and that computers, and information technology generally, do not significantly improve learning. Clearly, we need evidence rather than hype, despite the claims of Roblyer, Edwards and Havrilvk (1997) that: “Courseware quality is less troublesome now that it is in the early days of microcomputers when technical soundness frequently caused problems” (p. 116). This may well be true in a technical sense, but is educational courseware contributing to learning in a more significant way than earlier methods? If so, how is this occurring?

Evaluation of educational courseware remains as an important activity in higher education. The need for effective evaluation is as necessary today as it was when computers were first used in higher education.

Conclusion

A number of issues emerge from this review. There are limitations in using the checklist approach to evaluating courseware. McDougall and Squires (1995) have discussed the problems and limitations of the checklist approach to courseware selection, especially because this approach emphasised these technical attributes of packages at the expense of consideration of broader classroom and laboratory environments and activities, learning processes and other educational issues; ie. context was not addressed.

A case is also made for using a checklist developed for assessing aspects of online pedagogical learning materials in higher education. It is argued that, when used in conjunction with context-bound approach this may be a more useful in indicating the pedagogical quality of online learning materials. The checklist proposed is a useful screening device to use before undertaking a context-bound evaluation of courseware.

A ‘checklist’ approach to evaluation can provide useful, if incomplete information about educational courseware. Higher education faculty require additional evaluation data than that provided by checklist evaluation. More effort could be productively devoted to context-bound evaluations. Selecting courseware from a checklist is analogous to purchasing a car. A potential buyer is likely to have a mental checklist of what they want (eg. good fuel economy, price range, air-conditioning, power steering, automatic gears and such) in a car of their choice. A car being considered for purchase may have all the features required in the buyer’s checklist but the final decision is likely to rest on how the car how it performs in relation to the purchasers expectations.

A prospective car buyer is unlikely to make the final decision to purchase the car until they have actually test driven the vehicle. As with motoring magazines, subjective reviews by experts may also sway a prospective educational courseware purchaser. Once the checklist of essential elements (subject matter, level, cost, etc) of courseware is found, a context-bound evaluation may provide the visceral ‘driving’ experience a potential courseware purchaser seeks.

Lessons garnered from evaluating various forms of courseware may be used when evaluating on-line learning material. The conceptual development and application of courseware evaluations has lagged behind marketing initiatives. Courseware evaluation needs to improve considerably to ensure the health of educational computing (Hardin & Patrick, 1998). Hawkrigde (1990) points out that courseware was “only

one element in a complex teaching and learning process” and that faculty “need to understand the context in which it was used” (p. 106). Academics and practitioners are encouraged contribute to the sparse literature reporting context-bound evaluations of educational courseware for higher education, for as Hawkridge asserts” you only understand the potential of courseware when you use it with learners” (p. 106).

References

- Bieler, D. & Stevenson, I. (1998) OVUM REPORT: Internet Market Forecasts: Global Internet Growth 1998 – 2005 December ([http://www.gsmdata.com/repovum2.htm#Key findings](http://www.gsmdata.com/repovum2.htm#Key%20findings))
- Cheung, W. S. (1994). A new role for teachers: Software evaluation In J. Wright and D. Benzie (Eds.), *Exploring a new partnership: Children, teachers and teaching* (pp. 191-199). Elsevier.
- Creer, K. (1996) "Virtual Varsity Rushing Toward Reality", *The Weekend Australian*, 3-4 February: 20.
- Education Department of Western Australia (1984) *Climate*. WESOFT Educational Software.
- Ely, D. P. (1997) Technology is the answer! But what is the question? In R. M. Branch and B. B. Minor (Eds.), *Educational media and technology yearbook* (Vol. 22). Englewood, Colorado: Libraries Unlimited.
- Felix, U. (1997) Integrating multimedia into the curriculum: A case study evaluation. *On-Call*, 11(1), 2-11.
- Fouad, A. F. & Burleson, J. A. (1997) Effectiveness of an endodontic diagnosis computer simulation program. *Journal of Dental Education*, 61(3), 289-295.
- Grundy, S. (1991) A computer adventure game as a worthwhile educational experience. *Interchange*, 22(4), 41-55.
- Hamer, G. A. (1993) *The Use of Technology to Deliver Higher Education in the Workplace*, Department of Employment, Education and Training, Higher Education Division, Occasional
- Hardin L. and Patrick T. B. (1998) Content review of medical educational software assessments. *Medical Teacher*, 20(3) 207-211.
- Hawkridge, D. (1990) Software for schools: British reviews in the late 1980s. In O. Boyd-Barrett and E. Scanlon (Eds.), *Computers and learning* (pp. 88-108). Wokingham: Addison-Wesley.
- Keegan, D. (1986). *The Foundations of Distance Education*. London: Croom Helm.
- Levine, T. (1996) Courseware evaluation. In T. Plomp and D. Ely (Eds.), *International encyclopedia of educational technology* (2nd ed.). Pergamon.
- Lundin, R. (1993) *Overseas Experience in non-traditional modes of delivery in higher education using state-of-the-art technologies*, DEET Higher Education Division Occasional Paper, AGPS, Canberra.
- _____ MIT to make nearly all course materials available free on the World Wide Web (<http://web.mit.edu/newsoffice/nr/2001/ocw.html>)
- McDougall, A.;and Squires, D (1995) A critical examination of the checklist approach in software selection. *Journal of Educational Computing Research*, 12(3), 263-274.
- Oppenheimer, T. (1997) The computer delusion. *The Atlantic Monthly*, 45-62.
- Roblyer, M. D. Edwards, J. and Havriluk, M.A. (1997) Using software tutors and tools: Principles and strategies, *Integrating educational technology into teaching* (pp. 116-126). N. J. : Prentice Hall.
- Rowley, J. and Slack, F. (1997) The evaluation of interface design on CD-ROMS. *Online and CD-ROM Review*, 21(1), 3-14.
- Russell, G. (1994) Implications of context-based software development for education. *Australian Journal of Education*, 38(2), 157-173.
- Scanlon, E. (1997) Learning science on-line. *Studies in Science Education*, 30, 57-92.
- Scanlon, E. , Tosunoglu, C. , Jones, A. , Butcher, P. , Ross, S. , Greenberg, J. , Taylor, J. and Murphy, P. (1998) Learning with computers: Experiences of evaluation. *Computers and Education*,

30(12), 9-14.

- Schank, R. C. (1994) "Active Learning Through Multimedia", *IEEE MultiMedia*, Spring, :69-78.
- Schibeci, R. A. (1985) Educational software: Good, bad and indifferent. *Australian Science Teachers Journal*, 32(3), 23-27.
- Sharp, V. (1993) *Computer education for teachers* (2nd ed.). Madison, Wisconsin: Brown and Benchmark.
- Sharpham, J. (1993) "Managing the Transition to Mass Higher Education in Australia", *Long Range Planning*, April, 26:51-58.
- Sherry, L. (1996). "Issues in Distance Learning". *International Journal of Distance Education*, 1 (4):
- Squires, D. and McDougall, A. (1996) Software evaluation: A situated approach. *Journal of Computer Assisted Learning*, 12, 146-161.
- Taylor, J. C. (2001) *Fifth Generation Distance Education Australia*. Keynote Address presented at the 20th ICDE World Conference, Düsseldorf, Germany, 1-5 April 2001.
- Tergan, S. O. (1998) Checklists for the evaluation of educational software: Critical review and prospects. *Innovations in Education and Training International*, 35(1), 9-20.
- Volpe, R. M., Aquino M. T. B. and Norato, D. Y. J. (1998) Multimedia system based on programmed instruction in medical genetics: construction and evaluation. *International Journal of Medical Informatics*, 50(1-3), 257-259.
- Taylor, JC (1995) 'Distance education technologies: The fourth generation'. *Australian Journal of Educational Technology*, 11, 2, 1-7.
- Taylor, JC, Kemp, JE & Burgess, JV (1993) *Mixed-mode approaches to industry training: Staff attitudes and cost effectiveness*. Report produced for the Department of Employment, Education and Training's Evaluations and Investigations Program, Canberra.
- _____ (2001) Unpublished internal document developed by Edith Cowan University CTLQ Quality Online Working Group, 15 March, 2001.
- Steffe, L.P., & Gale, J. (Eds.). (1995). *Constructivism in Education*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Peter Hosie

Instructional Designer
Resources Development Centre
LIFT (Learning Innovations and Future Technologies)
Edith Cowan University
c/o 12 Osborne Parade
Clarmont, WA 6010
Australia
p.hosie@ecu.edu.au

[Back to contents](#)

Students Building Communities—Funding Schools

StudentDriven, Inc . (a web-based community information and business advertising platform) and **AmericaTakingAction for Kids** (a national school network), in a unique partnership to benefit schools nationwide, have developed and launched 33,000 Internet-based community web sites (<http://www.americatakingaction.org/studentdriven.html>).

Each city or community guide uniquely highlights the city's history, local businesses, hometown based professionals, schools, students, news, local city or county events and interesting people. In addition, the sites host job opportunities for junior/senior high school age students and college age students. Local city editors (students) collect and publish local school and community information, along with sports scores and highlights, special announcements and want ads. The result is real local information that builds and maintains the community interest.

These sites are financially supported by local businesses who link their web sites to the StudentDriven Community for their town. Computer class students and educators are encouraged to embrace the build-out of the school portion of the city site as a class project. Journalism educators, along with marketing class instructors, encourage students to author and edit local news about the school and the community.

Fifty Percent (50%) of the yearly business sponsorship fees are paid to the students and the local schools they attend and/or represent, which will continue year after year to produce extraordinary income.

Student benefits are many: funding for projects, scholarships and endowments, funding that brings the students into contact with potential future employers, funding that helps local business grow, funding that teaches students about commerce. Students will have the opportunity to participate in almost every phase of this project. From learning how to create community web sites that are not typical cookie-cutter sites, experience marketing area businesses, as well as the opportunity to have real copy writing, graphics, and programming skills that will benefit them for many years in the future.

Avis Marie Haynes

Executive Director

AmericaTakingAction for Kids

avis@americatakingaction.com

<http://www.americatakingaction.org>

[*Back to contents*](#)

Visualizing Thinking with Digital Imagery

My research group has been building digital imaging systems to help people think, learn, and do in new ways. We use digital imagery, photographs, video, and drawings, to render tacit thoughts and actions visible,

allowing them to become explicit objects for critique and reflection. We develop tools to analyze and explain "imagery as data" (Smith & Blankinship, 2000) $\frac{3}{4}$ to generate and share hypotheses about image content and to critique and modify these ideas with others.

One strand of this work asks learners to observe, interpret, and develop theories about behaviors and events depicted in imagery. For example, *Animal Landlord*, a video annotation tool designed for high school classrooms, uses nature films as its primary source of data (Smith & Blankinship, 2000; Smith & Reiser, 1997). Using *Animal Landlord*, students describe important actions in nature films, compare these to look for variations, and create qualitative decision trees to explain the evolutionary reasons for behaviors (Figure 1). Students use video as data to generate their own hypotheses about animal behavior rather than listening to descriptive narratives. They use video to create causal models of behavior, and they critique each other's models to develop understandings of evolution and ecology.

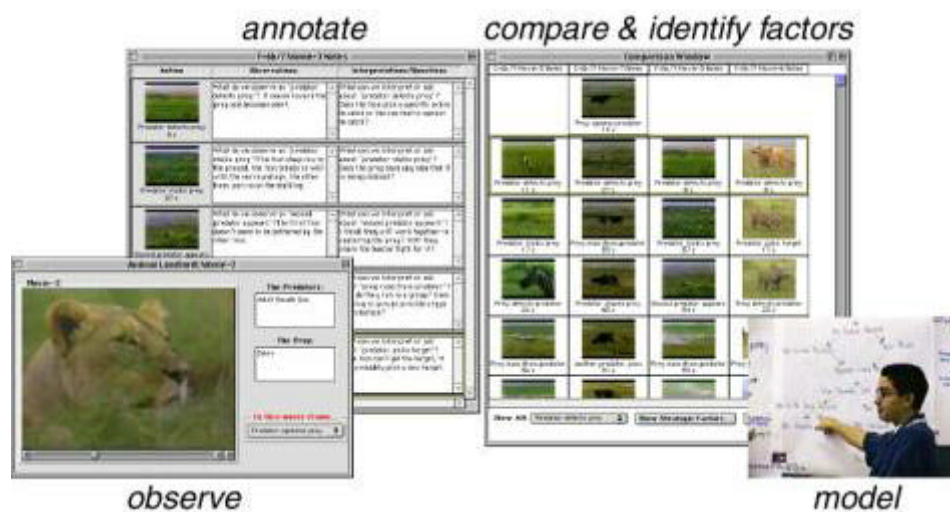


Figure 1. Students generate theories of animal behavior with *Animal Landlord* by a) observing nature films, b) selecting and annotating important scenes, c) comparing filmed events to look for similarities and variations, and, finally, d) constructing qualitative decision trees to explain behavioral patterns.

Animal Landlord engages students in a process that resembles the practices of behavioral ecologists. Similarly, Erik Blankinship's *Image Maps* helps students adopt the practices of expert historians by providing data and analysis tools for generating hypotheses about how and why their communities have changed over time (Smith & Blankinship, 2000; Smith et al., 1999). We began by creating "a time machine," a way for students to photograph today's urban landscape and automatically compare them with images of the past. We augmented a digital camera with a global positioning system (GPS) and a digital compass. Every photograph taken with this camera records an image plus the camera's current latitude, longitude, and orientation. We use these position data to retrieve and present historical images of the same location (Figure 2).

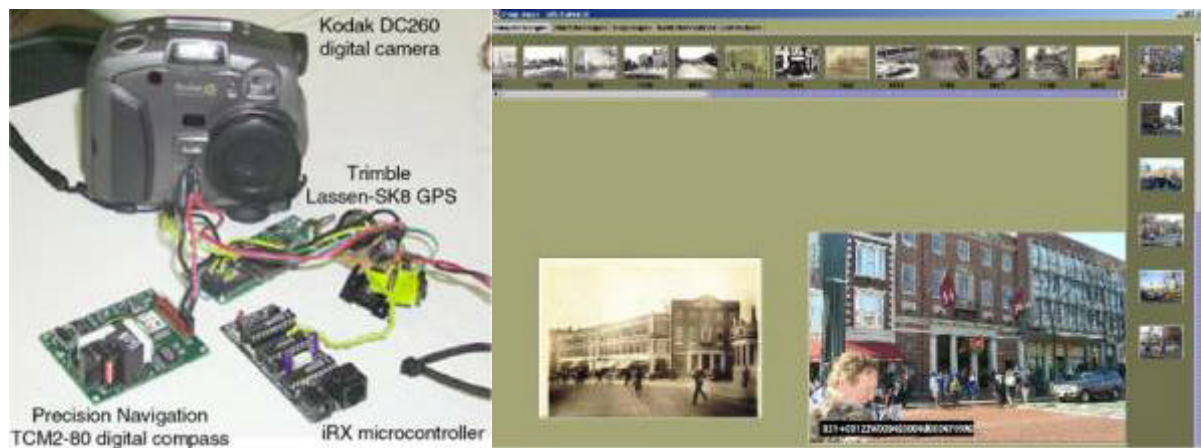


Figure 2. *Image Maps'* location-aware camera allows students to compare photographs of the present with historical images of the same location. Their task is to use these historical records to explain how and why their communities have changed over time.

The main goal is to make a location's history explicit to learners. As they talk with other students, they develop theories about urban planning decisions. We tried to help students understand that history is more than memorizing dates and facts by giving them opportunities to conduct investigations around historical images. For instance, some students question the ways that traffic has flowed through Harvard Square over the last hundred years to explain how the shift from horses to trolleys to automobiles forced the urban landscape to change. Quite often, groups of students will develop different explanations for the same historical changes. This helps them understand that there can be multiple, often competing, accounts of past events.

Animal Landlord and Image Maps use existing imagery to help students understand and adopt expert practices. We have also encouraged learners to generate and analyze images of their own practices, to question beliefs and assumptions about their ways of thinking and doing. For instance, Jonah Peretti's *Teacher's LAB* focuses teachers on curricular enactment (Peretti, 2001) by introducing visual sketches to illustrate classroom activities like writing on chalkboards, breaking into small groups, coming together for whole class discussions (Figure 3). In a sense, he helps teachers reflect on the "pedagogical patterns" that they use to structure classroom learning. By helping teachers diagram their practices, they begin to understand some of the tacit assumptions they make during their daily activities. More so, the work suggests a new form of collaborative professional development, where teachers create, share, and critique images of "best practice".

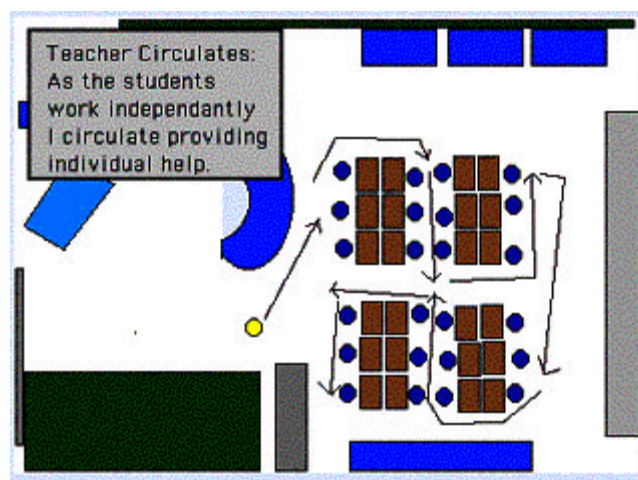


Figure 3. *Teacher's LAB* encourages teachers to diagram and reflect upon basic “pedagogical patterns” like moving through the classroom to assist students.

While *Teacher's LAB* concentrates on teacher practices, Jeana Frost tries to change the health practices of newly diagnosed diabetics in her *Digital Mirror* project (Frost, 2001). While diabetics currently monitor their health by measuring their blood sugar, the resulting quantitative data do not explain *why* they are or are not in danger of diabetes complications. Frost introduced photography into health monitoring, asking patients to capture daily activities, to take pictures when they ate, exercised, felt stressed, and so on. She also created a visualization tool to synchronize these images and blood sugar measurements on a timeline (Figure 4). This allows patients to discover correlations between their behaviors (the photos) and their physiological state (the blood sugar data). The goal is to help diabetics gain an awareness of their behaviors and their impact on long-term health.

In all of these projects, we are focusing on the tacit beliefs and assumptions held by “everyday experts.” For *Animal Landlord* and *Image Maps*, we find that students have strong beliefs about classroom science and history. They are accustomed to films and photographs being accompanied by explanations that provide “the right answers.” Our computer systems challenge these explanations by forcing students to understand that these domains are really about hypothesis generation and testing, collaborating with others to explain and justify theories with supporting evidence. In both cases, this evidence takes the form of digital imagery, and the process of studying the imagery becomes more important than coming up with “the right answers.”

Teacher's LAB and *Digital Mirror* use imagery to make people's personal **acts** into artifact **acts** for critical reflection. Teachers begin to think about their classroom practices, how they use space to organize learning. Diabetics begin to understand that their best intentions are often accompanied by bad habits that compromise their health. These learners use digital imagery to investigate their own ways of doing. In both cases, our preliminary observations suggest that teachers and diabetics are developing new insights into their practices, and future work will examine whether we can establish long-term behavioral changes.

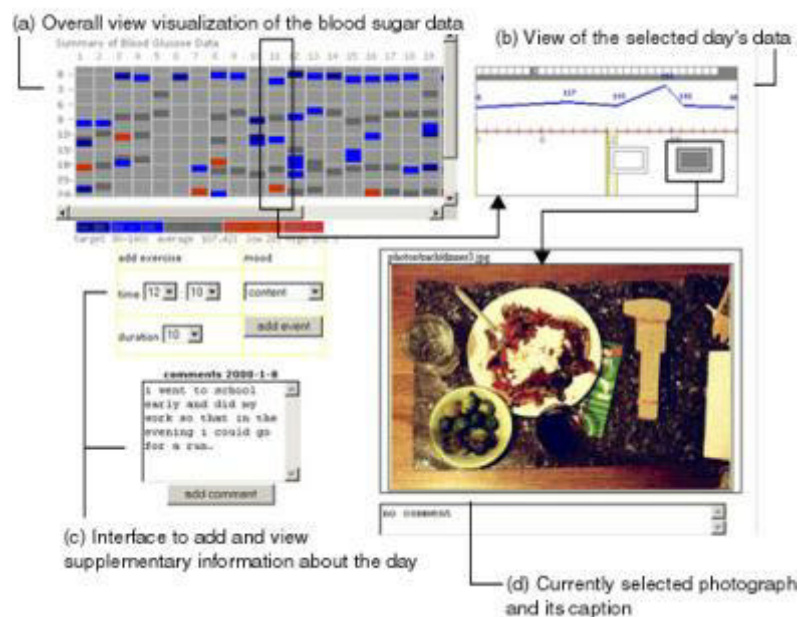


Figure 4. *Digital Mirror*'s visualization shows a) blood sugar levels over time, (b) daily blood sugar and photographs linked to the time they were taken, c) additional patient annotations, and d) the currently selected photograph.

Imagery has a long history as an object of instruction; we are developing systems to promote its use as an object for discussion. We want people to challenge their understandings by observing and interpreting visual data. Some of our work focuses on common imagery, nature films, archival photographs, children's television, etc., trying to help learners understand that there is more to these images than their narrations and captions suggest. Another research strand focuses on capturing personal activity as imagery to help people visualize and critique their own ways of thinking and doing. We continue to explore both areas to understand how the combination of imagery and computation can impact learning and performance.

References

- Frost, J. (2001). *Visualizing Health: Imagery as Data for Changing Personal Practice*. Unpublished MS Thesis, MIT, The Media Laboratory, Cambridge, MA.
- Peretti, J. (2001). *Teacher's LAB: A Platform for Teacher Learning*. Unpublished MS Thesis, MIT, The Media Laboratory, Cambridge, MA.
- Smith, B.K. & Blankinship, E. (2000). Justifying imagery: Multimedia support for learning through explanation. *IBM Systems Journal*, 39(3&4), 749-767.
- Smith, B.K., Blankinship, E., Ashford III, A., Baker, M., & Hirzel, T. (1999). Inquiry with imagery: Historical archive retrieval with digital cameras. In *ACM Multimedia 99 Proceedings* (pp. 405-408), New York: ACM Press.
- Smith, B.K. & Reiser, B.J. (1997). What should a wildebeest say? Interactive nature films for high school classrooms. In *ACM Multimedia 97 Proceedings* (pp. 193-201), New York: ACM Press.

Brian K Smith

MIT Media Laboratory
20 Ames Street, Room E15-305B
Cambridge, MA 02139, USA
bsmith@media.mit.edu

[Back to contents](#)

sitesALIVE! to Launch New Sailing Adventure for Classroom Study

(Ocean Challenge Live! – New York to Melbourne)

sitesALIVE!, an educational website linking classrooms around the world with student-staffed research expeditions, this fall will launch a new program that will track a sailing vessel out to break a long-distance

sailing record set nearly a century and a half ago.

Rich Wilson, founder and president of sitesALIVE! and a noted competitive sailor, will be joined by long-time crew mate Bill Biewenga aboard the 53-foot trimaran *Great American II*, avessel they crewed in 1993 on a previous record-breaking sail from San Francisco to Boston, around treacherous Cape Horn on the tip of South America.

Come September, the two-man crew will set out from New York city harbor in an effort to break the New York to Melbourne, Australia record run of the clipper ship *Mandarin*, which in 1855-56 made the 13,000-mile voyage in 70 days.

“While the trip itself will be a great adventure, its potential as a learning tool is even greater,” says Wilson. “Through our website, www.sitesalive.com students and teachers all over the world will be able to track the voyage, with all its complementary learning aspects, geography, navigation, history, communications, weather and the like.”

Wilson already has demonstrated the innovative learning experiences that can be generated by such a journey. He founded sitesALIVE after his successful 1993 Cape Horn voyage, which lasted 69 days and 20 hours, breaking the record set by the clipper ship *Northern Light* during the California gold rush. (An earlier try at the record ended in 1990, when Wilson and his co-skipper Steve Pettengill had to be rescued in high seas off South America).

During his record-breaking voyage Wilson wrote an 11-part series and periodic updates for students that were published in a dozen newspapers nationwide. In addition, Prodigy produced the first interactive learning adventure based on the voyage. An estimated 300,000 students followed the Cape Horn voyage via the Internet and the newspaper columns that were used in classrooms. When Wilson and co-skipper Biewenga sailed into Boston, they were greeted by more than a thousand of these school children.

Wilson, a former teacher himself, founded the sitesALIVE! parent company, Ocean Challenge in 1991. By the mid-1990s he began establishing partnerships with research organizations like the Center for Coastal Studies, The Center for Marine Resource Studies, and the Center for Rainforest Studies, centers that offer a variety of hands-on student projects around the world.

Currently, sitesALIVE! offers the following programs, which will run concurrently with the new Ocean Challenge Live! – New York to Melbourne program:

- Rainforest Live!--Queensland, Australia
- Class Afloat Live!--Concordia sails the world
- Oceans Live!--Turks and Caicos Islands
- Costa Rica Live!--Atenas, Costa Rica
- Wetlands & Fisheries Live!--Baja, Mexico
- U.S. History Live!--Aboard *Spirit of Massachusetts*
- Math Live!--A program derived from all of the above

Linking up with these various field schools, all of which are accredited, semester-long programs, sitesALIVE! has developed an exciting repertoire of learning adventures designed to motivate classroom students while at the same time providing teachers with vital and immediate learning resources. All of the programs are correlated with the National Standards, and teachers are provided with relevant teacher guides and materials.

For example, for the upcoming Ocean Challenge Live! – New York to Melbourne, teachers will be receiving: A map comparing the progress of the *Great American II* and *Mandarin*; history of the Australian Gold rush;a ship’s log of data radioed from *Great American II*; regular quizzes on various aspects of the voyage;

daily journals and updates; Teachers Guide for class study in relevant topics such as geography, nutrition, history, mathematics, etc.; and Wilson and Biewenga will write essays and provide journals, answer questions, post photos and record audio reports from sea for sitesALIVE!

“We’ll only be a crew of two on this voyage,” says Captain Wilson, “but I’m looking forward to having thousands of students keeping me company on this trip, just as they did the last time. And I can guarantee they’ll enjoy the adventure, learn a great deal, and they’ll never get seasick.”

Cindy Collins

Director of Marketing

sitesALIVE.com

225 Friend St., Fifth Floor

Boston, MA 02114 USA

ccollins@sitesalive.com

<http://www.sitesalive.com>

[*Back to contents*](#)

Technology Based Training in Ancient Egypt

A question

Why is it that so many technology based training projects run over time, run over budget, or fail to deliver what they initially promised? The answer is more complex than it might initially appear. In part it is a consequence of failures in project management during the execution of the project. But we can also see the seeds of that failure in the initial contract negotiations where well-meaning sales staff propose unrealistic prices and timescales for a given level functionality without realising that they cannot be achieved – and well meaning but naive clients accept those proposals. That is not to say that the projects fail: the resulting materials may be quite effective – but the client has not got as much as he or she might have done if the project management had been better. Neither are all organisations failing in this way: those that are better at selling and managing their projects are just more successful than the majority who are not – and that majority bring down the effectiveness of technology based training as a whole.

The evidence to support this unpalatable assertion comes, not only from our experience but also from a recent study carried out in the United Kingdom and published by the UK Department for Education and Employment in 2000 (Seabrook and Grigg, 2000). The skills of Project Management, Client Management, Planning, and Costing and Budgeting were all in the top 6 ranked in importance from a list of 40 TBT development skills. The scores allocated to these skills were so high that it is clear the vast majority of TBT developers and managers recognise they were essential and needed to be carried out to a high quality and degree of expertise. Significantly these same skills also scored highly in questions requesting skills to be improved and skills in short supply. So we have the situation where key skills are in short supply and the

ones that are being used are in need of improvement.

Exercise is good for you

To help address this training need, the authors have developed an exercise designed to help improve skills in managing creative people. It was initially developed for the technology based learning community but has wider application wherever creatives need to be 'managed' so that they can make the greatest contribution to the teams in which they are working.

Project managers sometime despair of the designers in their teams. Managing creatives is regarded as an oxymoron – a contradiction in terms, rather like user-friendly software and empathetic bank managers. They are perceived as having their own agendas, a disregard for budgets and timescales, and a perverse ambition to experiment with new creative ideas without letting their managers in on the secret. For their part the creative members of the team seem to regard project managers as reactionary forces who stifle creativity and need to be out-manoeuvred or avoided.

The exercise concerns the start of a project where the sales team are trying to win some work which is not well understood, either by the client or the contractors. At this stage creative and analytical skills can conflict head-on, information is patchy and good communication between everyone is essential, but unlikely, as relationships of trust have not had time to fully develop. Different people have different ideas (some of them quite erroneous) about how long things should take and how much the client should be charged.

The exercise explores the way that creative team members and managers can work together by placing them in a non-threatening setting where they can explore their contributions to the project without risk to the outcomes or their self-esteem. That setting is Ancient Egypt where Shabaka the Pharaoh is ailing fast and needs a pyramid quite quickly. Participants then takes the roles of members of the royal court, the priesthood, designers, builders and viziers (project managers). Collectively, the designers and builders have to develop a costed proposal with realistic timescales which meets the functional specification produced (perhaps) by the Royal Court.

The training objectives are closely linked to the competences identified in the report described above and generally held to be important for the development of 'good' technology based training.

Needless to say, the teams' plans do not always work.. If they appear to be progressing too well then the facilitators (The Gods) can intervene with, for example, a plague of frogs: it will come as little surprise that the teams can generate plenty of problems for themselves without divine intervention!

We have confirmed that the degree of enthusiasm which participants enter into their roles affects the clarity of the lessons to be learned. But, strangely, the amount of movement is also an important factor. This freedom of movement within and between the client and provider teams enhances communication and respect for one another's skills. The game brings out the different agenda surrounding a project on all sides, and the range of skills needed to create a 'designer pyramid,' all co-ordinated by the key management skills identified earlier. The same important factors come out whether the game is run in the UK or the USA.

But does it work?

The feedback we have received both during the exercises and through questionnaires indicates that the participants can readily draw analogies between the role and events in the exercise and their real life

experiences of developing technology based training. Thus, we are reasonably confident that that exercise has sufficient validity to meet the goals we set. Assessing its impact on the key competences identified by Seabrook and Grigg (2000) is more difficult. Certainly the participants say that they 'enjoyed' the experience and now feel more competent, but the only reliable measure will be to monitor the performance of companies and individuals over a period of months to see whether they really are more competent in working with clients and suppliers to commission and develop high quality, effective learning materials.

Reference

- Seabrook J and Grigg J (2000) *Authoring for CBT and interactive multimedia Report LLTD4*
Published by DfEE and available from Learning Technologies Division (tel: +44 845 60 222 60)

Jan Seabrook and Nick Rushby

Conation Technologies

nick.rushby@conation-technologies.co.uk

jan.seabrook@conation-technologies.co.uk

[Back to contents](#)

Digital Environments: Monitoring Changes to Teaching

Abstract

The pressures, from outside and inside universities to digitize, to computerise, and 'to go online' continue to increase in pace. The rhetoric surrounding such pressures are often argued from a standpoint that positions technology as a 'neutral tool', a tool that will help make work more efficient and effective. The central thesis of this paper is that technology is best understood as fundamentally non-neutral and that various technological applications *privilege* certain ways of doing things, and exclude or restrict other ways. And further, that technology not only influences but can have a transformative effect on what the technology is used for, and on the work practices of those that use it. This paper draws attention to certain practices and changes in work practices in higher education, brought about, in part, or in whole, by the use of the new information and communication technologies (ICTs). In order to draw attention to these changed practices, this paper has been organised under four broad categories, namely: new pedagogical opportunities, changed practices, technology neutrality, and unintended effects.

Introduction

What might help us to rethink the way we use new technology is to develop a model that encourages us to critically question any form of new technology adoption. This paper proposes one such model. The model consists of four elements that need to be mapped onto any decision making process, in reviewing new technology for teaching and learning. The four elements are: new pedagogical opportunities; changed work practices, technology (non)neutrality and unintended consequences of new technology adoption. Each of the four elements applied to educational practice is outlined below.

New pedagogical opportunities

New pedagogical opportunities can be viewed at two different levels. The first level relates to new opportunities to do the same thing we have always done, just adding the technology to it, perhaps making us 'more efficient'. This can be exemplified by viewing activities, which are the same, though 'enhanced' or 'made easier' by the use of new technologies. For example, placing lecture notes and PowerPoint presentations on the Web, providing students with an alternative source to receive lectures notes and PowerPoint presentations, perhaps, in advance of a lecture. There may well be advantages of using new technologies in this way, but if this is all we are doing with the new technology, I would question whether we should bother. Are there real advantages, for example, in transferring notes designed in one medium and placed to another? What is the add-on value for the students and for the lecturer in doing this?

The next level of opportunity in using new technology is enabling us, as teachers and our students as learners, to do something different. Something we would not have been able to do without the use of the new technology. For example, the Web has many useful ready-made resources we can tap into. Rather than developing our own content for the course, we could set students activities to use relevant resources already developed online. One online resource that exemplifies this notion of new and different pedagogical opportunities, is the 'Active World's Universe' – <http://www.activeworlds.com.edu> - which enables students to explore subject environments in simulated settings. Students and teachers from various educational institutions around the world are able to communicate, collaborate and interact with each other to solve various problems and activities within these environments. Research studies exploring the potential of such environments include the following: Dickey, 1999; Pelgrum & Anderson, 1999; Law et al., 2000; Jonassen, 2000.

Changing work practices

Changing work practices refers to the changes in the way we work, how we work, who we work with and what we work on. New technologies place us in a different position to that we have previously been in.

One example of changes in work practices relates to work time. A lecturer may respond to student queries late at night, if she is 'online' and has a relevant response to make. This shift in work time and the speed in which responses can be made, may well change student expectations regarding, for example 'turnaround time': the time taken for lecturers to respond to student queries. These changes in work practices need to be monitored to ensure students, teaching staff and academic managers are aware of these changes as they occur and therefore are able to make changes to operational plans accordingly.

Technology (non)neutrality

Bowers (1988, p. 26) states that a 'widely held view within our culture is that technology is neutral' and is

seen to function best when we are unaware of it, when it is 'transparent' and does not interfere with what we are trying to do. In this paper, I contend that technology is not neutral and hence it has an influential and transformative effect on what it is used for and how it is used. For example, the ease with which some Web courseware tools allow us to transfer our MsWord files into the Web, encouraging us to transfer text designed in one medium into another. A common term for such activity is known as 'shovelware' (Hopper, 1999). It should be noted that recent research into reading (eg Rohonyia, 1999; Lander, 1999) reveals that we read very differently off screen than we do off paper. In addition, Schriver (1997) points out we tend to read less accurately and slower off screen than off printed pages. If we are guided by this research, we need to think critically about what resources we place online and how we expect our students to use the online resources we provide them.

Unintended consequences

The uncritical adoption of technology to educational settings is not a new phenomenon. However, because of the perceived benefits which technology can bring, there is a danger that education becomes captive to a non-problematised use of technologies (Green, Gough & Blackmore, 1996) without considering how it effects and transforms various contexts and situations (eg Kling, 1996). Seldom are there public discussions about the potential problems and consequences of adopting a particular technology or even if there are unintended consequences of adopting a technology which may not be apparent at the time (Tenner, 1996; Burbules & Callister, 1999). Technology is often cited as the way to overcome problems or difficulties faced. However, the unintended consequences are often hard or impossible to predict and are only realised after a passage of time. Too much emphasis on providing materials online can backfire, as Shirley Alexander found (1998). In a presentation on her research, Alexander reported that several students had strongly objected to too much online work. As one student told her: 'I did not come to university to read text off screen!'

Final comments

This paper has reviewed some of the complex and interwoven issues of technological practice and change. The relationship between the use of new technologies, especially ICTs in higher education, academic practices and the varied contexts in which these practices occur, are complex and changing. These complexities can be investigated by examining staff perceptions of their role, the role of technology and its use in the institutions and by the use of the model introduced in this paper. The four broad concepts have been introduced in an attempt to maintain a *problematised* view of new technology adoption.

References

- Alexander, S. & McKenzie, J. (1998). *An Evaluation of Information Technology Projects for University Learning*. Canberra, Australian Government Publishing Service.
- Bowers, C. A. (1988). *The Cultural Dimensions of Educational Computing: Understanding the Non-neutrality of Technology*. New York, Teachers College Press.
- Burbules, N. C. & Callister, T. A. (1999). 'A post-technocratic policy perspective on new information and communication technologies for education' in J. Marshall & M. Peters (eds.), *Educational Policy*. Cheltenham, UK, Edward Elgar.
- Dickey, M. (1999). *3D Virtual Worlds and Learning: An Analysis of the Impact of Design Affordances and Limitations in Active Worlds, blaxxun interactive, and OnLive! Traveler; and*

A Study of the Implementation of Active Worlds for Formal and Informal Education. Doctoral dissertation, Graduate School, Ohio State University, Ohio,
<http://michele.netlogix.net/newside/bioframe.htm> (accessed 5 March, 2001).

- Green, B., Gough, N. & Blackmore, J. (eds.) (1996). *The Australian Educational Researcher*, vol. 23, no. 3, Geelong, Australian Association for Research in Education.
- Hopper, K. (1999). Shovelware, IT Forum, Listserve discussions Shovelware, IT Forum, Listserve discussions, <ITForum@uga.cc.uga.edu> (accessed 2 October 2000).
- Jonassen, D. H. (2000). *Computers As Mindtools for Schools*. Upper Saddle River, New Jersey, Merrill.
- Kling, R. (ed.) (1996). *Computerization and Controversy: Value Conflicts and Social Choices*, 2nd edn, Academic Press, San Diego.
- Lander, D. (1999). Online learning: Ways to make tasks interactive Online learning: Ways to make tasks interactive, <<http://ultibase.rmit.edu.au/Articles/lander2.htm>> (accessed 21 September 2000).
- Law, N., Yuen, H.K., Ki, W.W., Li, S.C., Lee, Y. & Chow, Y. (2000). (Changing Classrooms & Changing Schools: A Study of Good Practices in Using ICT in Hong Kong Schools. Centre for Information Technology in School and Teacher Education, Hong Kong.
- Pelgrum, W. J. & R. E. Anderson (1999). *ICT and the Emerging Paradigm for Life Long Learning: a Worldwide Educational Assessment of Infrastructure, Goals and Practices*. Amsterdam, IEA & Enschede, University of Twente.
- Rohonyi, A. (1999). 'Webwriting, webreading, weblearning' *The New Educational Frontier: Teaching and Learning in a Networked World. Proceedings of the International Council for Open and Distance Education, Nineteenth World Conference on Open Learning and Distance Education*, June 1999, Vienna, (CD-ROM).
- Schriver, K. (1997). *Dynamics in Document Design*, John Wiley, New York.
- Tenner, E. (1996). *Why Things Bite Back: Technology and the Revenge on Unintended Consequences*, Knopf, New York.

Robert Fox

Associate Professor

Centre for Information Technology in School and Teacher Education (CITE)

Faculty of Education, University of Hong Kong

Pokfulam Road, Hong Kong

foxrmk@hkucc.hku.hk

Tel: +852 2859 8014

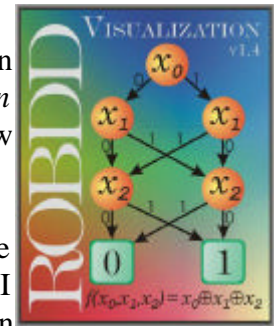
Fax: +852 2517 7194

[Back to contents](#)

Educational Software for Computer Engineering: A Case Study of an Interactive BDD Tool

Abstract

Ordered Binary Decision Diagrams (BDDs) are a data structure for representation and manipulation of Boolean functions often applied in *Very Large Scale Integration* (VLSI) *Computer Aided Design* (CAD). Algorithms have to be developed to allow efficient BDD manipulation and minimization.



In this paper we describe an example of an educational software, i.e. the implementation of a BDD package in JAVA, developed for teaching interactive VLSI CAD courses. A user interface allows to display the construction and minimization process and by this gives further insight into the data structure. New minimization algorithms can easily be evaluated and their performance can be analyzed. This helps in understanding the basic principles of BDDs. The interactive tool allows for a fast understanding of the algorithms flow and can be directly used in on-line learning and tele-education.

Introduction

Decision Diagrams (DDs) are often used in VLSI CAD systems for efficient representation and manipulation of Boolean functions. The most popular data structure *are reduced ordered Binary Decision Diagrams* (BDDs) [Bry86,DB98], also called ROBDDs. They are part of almost all courses on function representation in VLSI CAD education. BDDs are very sensitive to the variable ordering, i.e. the size of a BDD (measured in the number of nodes) may vary from linear

to exponential. Finding the optimal variable ordering is an NP-hard problem and the best known algorithm has runtime exponential in the number of variables. This is the reason why many authors presented heuristics for finding good variable orderings from circuit descriptions in the last few years. The most promising methods for BDD minimization are based on dynamic variable ordering [FMK91], i.e. improving graph size using exchanges of neighboring variables. The best results measured in the number of nodes of the resulting BDD were obtained using sifting [Rud93,PS95]. For getting a better insight into the minimization process, it is very helpful to have some visualization of the minimization process. This is the reason, why several authors of BDD packages provide (graphic) outputs of the BDDs [BRB90,Lon93,Som97,Hoe99]. But none of these packages allow any user interaction during visualization.

In this paper we describe the implementation of a BDD package in JAVA. The BDD package has an interactive interface that allows to display the construction and minimization process and by this gives further insight in the data structure. Almost every formula can be entered and immediately translated into a corresponding BDD. Users can easily change the global variable ordering in a given BDD simply by selecting a certain variable with the mouse and exchange its position with adjacent variables. The effects on the BDD can be seen immediately. This allows the user to manually optimize the BDD if the automatic minimization algorithm returns unsatisfying results. The view area can also be changed using the mouse in order to zoom into very large structures. New minimization algorithms can easily be evaluated and their performance can be analyzed. The tool can be directly applied in on-line learning and tele-education. The user gets “hands-on” experience while working with the software.

To make the paper self-contained, a brief definition of BDDs is provided. The next section will shortly describe the main aspects of BDDs and the impact of the variable ordering followed by a short chapter on implementation of the package in JAVA (for more details see [DR01]). Finally the key features of the *Graphical User Interface* (GUI) are briefly discussed.

Preliminaries

Binary Decision Diagrams

Each Boolean function can be represented by a *Binary Decision Diagram* (BDD) [Bry86], i.e. a directed acyclic graph where a Shannon decomposition is carried out in each node. In the following, only reduced, ordered BDDs are considered and for brevity these graphs are called BDDs. (For more details see [DB98].)

We briefly consider the typical synthesis operation on BDDs: The synthesis of two BDDs is carried out by performing a recursive call on sub-graphs. We refer to this recursive splitting algorithm in the following as the *If-Then-Else* (ITE) algorithm, since based on an if-condition one of two branches is selected. Based on ITE all binary synthesis operations, like AND and OR, can be carried out.

Dynamic Minimization

The basic operation of dynamic variable ordering is the exchange of adjacent variables [FMK91,Rud93]. The exchange is performed very quickly since only edges have to be redirected within these levels. Thus, the size is optimized without a complete reconstruction of the BDD. Only local transformations for the two levels are performed. This is due to the observation that BDDs are a canonical form. The exchange of two variables do not change the sub-BDDs of other levels.

The sifting algorithm [Rud93] successively considers all variables of a given BDD. When a variable is chosen, the goal is to find the best position of the variable, assuming that the relative order of all other variables remains the same. In a first step, the order in which the variables are considered is determined. This is done by sorting the levels according to their size with largest level first. To find the best position, the variable is moved across the whole BDD. In [Rud93], this is done in three steps:

1. The variable is exchanged with its successor variable until it is the last variable in the ordering.
2. The variable is exchanged with its predecessor until it is the topmost variable.
3. The variable is moved back to the closest position which has led to the minimal size of the BDD.

BDD Implementation in JAVA

In this section we describe the main components of our implementation. The core of the BDD package is the underlying graph structure.

Storing Nodes

For each node the two outgoing edges and the label at the node have to be stored. Furthermore, we support additional information to allow simple handling of the graphical output, like e.g. the x- and y-coordinates when drawing the graph. For the constructor of the node class we get:

```
public node(int vi) {  
  G = null;  
  H = null;
```

```

v = vi;
last_x = 0;
last_y = 0;
version = 0;
selected=false;
counted=false;
}

```

In the same way a node can be created by assigning the successors **g** and **h** already during the call, i.e. **public node(int vi, node g, node h)**. The class also provides a set of operators for efficiently traversing the BDD and determining information, like the number of nodes. Based on this simple setup the recursive algorithm ITE described in the previous section can be directly applied.

Synthesis of Nodes

The ITE algorithm operates on three nodes F, G and H given as parameters and merges them together returning a result node R. The function represented by R has to be in a strong canonical form. Thus, additional checks have to be added to maintain this form. The fact that ITE can be used to perform all binary operations simplifies the implementation of all synthesis operations; the following example shows the synthesis of nodes using the basic Boolean functions AND, OR and NOT:

```

public node AND(node a, node b) {
    return ITE(a, b, ZERO);
}
public node OR(node a, node b) {
    return ITE(a, ONE, b);
}
public node NOT(node a) {
    return ITE(a, ZERO, ONE);
}

```

These class methods are automatically invoked during parsing of the given user formula.

Parsing the Formula

Parsing of the user formula is implemented using the parser generator CUP ("Constructor of Useful Parsers") by Hudson [Hud98]. Using a simple parser-specification file CUP generates JAVA source-code which can be adapted to the given context. In general, these specification files contain the grammar for the parser.

Drawing the Graph

Structure Analysis

Before we can draw the BDD data structure in a panel for user interaction, a complete analysis must be performed. Starting from the graph's root we descent recursively in the BDD to determine the correct x- and y-coordinates for each node. This analysis is only possible because each node is unique in a given BDD.

Displaying the BDD

The actual displaying (in the following called 'painting' – derived from JAVA's *paint()*-methods) of the BDD is done by a special method in class node. Seen in a larger context, the *paint()*-method of the BDD's panel is responsible for a correct call sequence of the *structure_analysis*- and *paint_node*-method. By modifying *orig_x* and *orig_y* the graph can be moved easily, e.g. by dragging it with the mouse.

More details on the implementation can be found in [DR01].

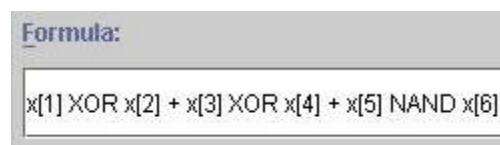
GUI - Graphical User Interface

The visualization of BDDs is an important topic, since this gives some feedback how well the minimization techniques, like sifting, are doing. For this, most BDD packages presented so far [BRB90,Lon93,Som97,Hoe99] have an output option to display the graph already build. But none of them allow any user interaction or visualization of the BDD manipulation, i.e. minimization and graph construction.

The key advantage of this BDD package implementation is the ability to change the global variable ordering. This can be achieved in different ways. Another important improvement is the visualization during automatic minimization using sifting heuristic and during BDD construction.

In the following we describe the *Graphical User Interface* (GUI) of our JAVA based BDD package (see also Figure 1 and Figure 2 for GUI screenshots). The key features are:

- **Trace mode:** During BDD construction using the ITE operator each operand F, G and H is displayed in a window. By this, the construction can be observed step by step. This is very helpful when getting familiar with BDDs and also to debug programs.
- **Automatic minimization:** Sifting heuristic can be applied to the current BDD to minimize the number of nodes used. A *slideshow-mode* is available to watch the BDD after each sifting-step. Animation delay can be adjusted.
- **Setting the variable ordering:** Using the variable ordering window a (partial) ordering can be set and the BDD is directly modified accordingly. Algorithms can be tested easily this way (see also Figure 3 Setting the global variable ordering).
- **Interactively change variable ordering:** Use the mouse to select a specific variable and transpose it up and down the current ordering while watching immediate reconstruction of the BDD accordingly.
- **Verbose mode:** In each step a protocol is reporting the current action. This simplifies the understanding of the algorithms.
- **Simple input format:** A simple format allows to input formulas in a readable form, like $x[1]*x[2]+x[3]$ stands for $x_1 \text{ AND } x_2 \text{ OR } x_3$. An example formula in the GUI looks as follows:



- **Resizing the BDD:** Use the mouse to easily move (click and drag) and resize the BDD to fit your current window size or to zoom into the data structure.
- **Swap high/low edges:** Low and high sons (THEN and ELSE branches) can be swapped, i.e. left son

is *low* son and right son is *high* son or vice versa.

- **Keep configuration:** When finishing the program the current configuration can be stored in a file. Next time the program starts it may restore the saved values. The program supports several configuration files.

Conclusions

In this paper we presented a visualization of a JAVA based implementation of a BDD package. We enhanced the recursive ITE-algorithm to accept different variable orderings which could be changed interactively. Additionally the sifting algorithm was implemented to allow automatic minimization of the BDD by finding a good variable ordering. An optional step-mode allowed a user to watch the BDD synthesis algorithm. The tool is used for teaching BDD algorithms and provides much more insight than the “classical” presentations, since the user can “play around” with various parameters and options can see the development of the graph in the construction process. JAVA has been used as a programming language to also allow web-based training. It is focus of current work to establish a complete VLSI CAD course based on the principles shown here.

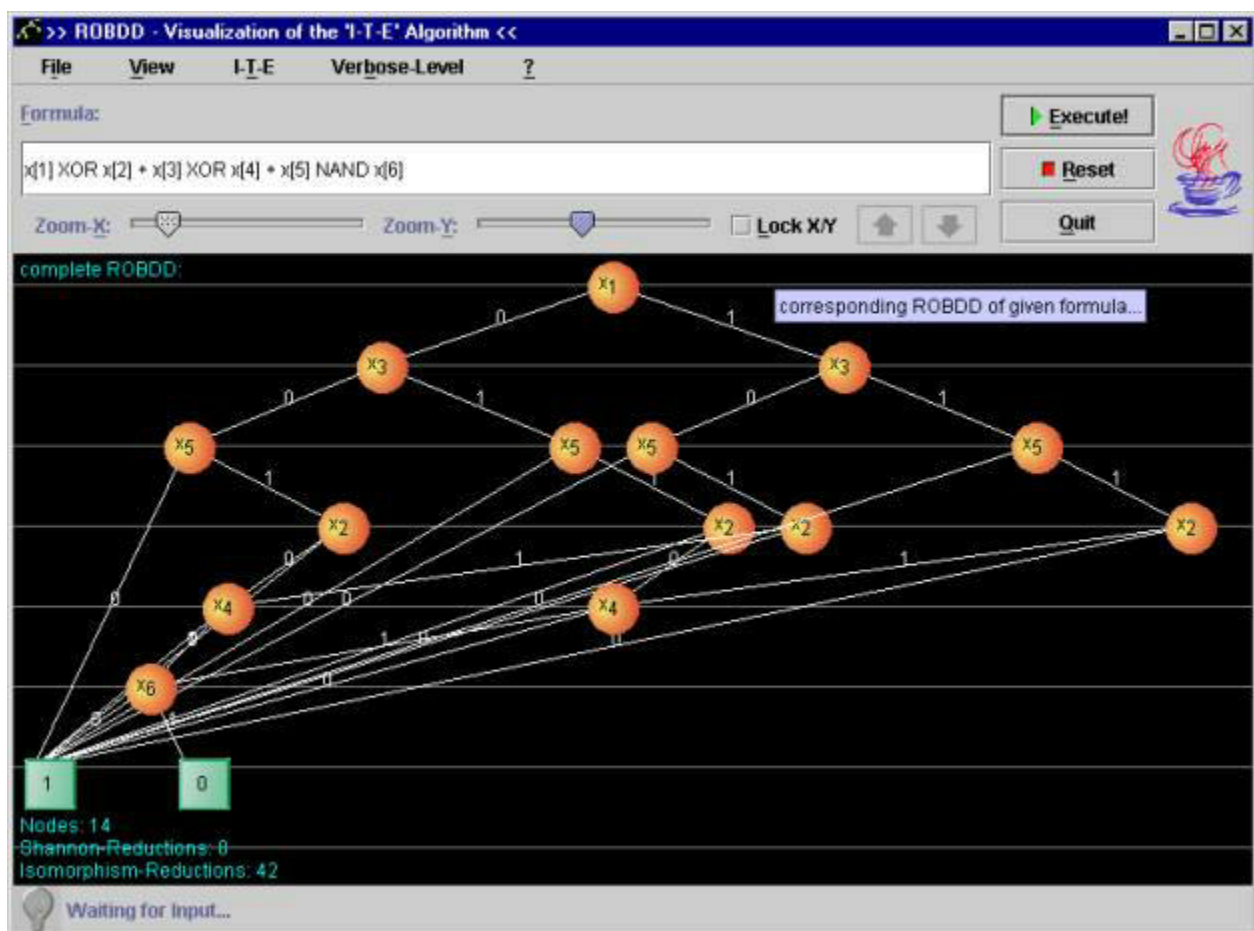


Figure 1. Screenshot before sifting

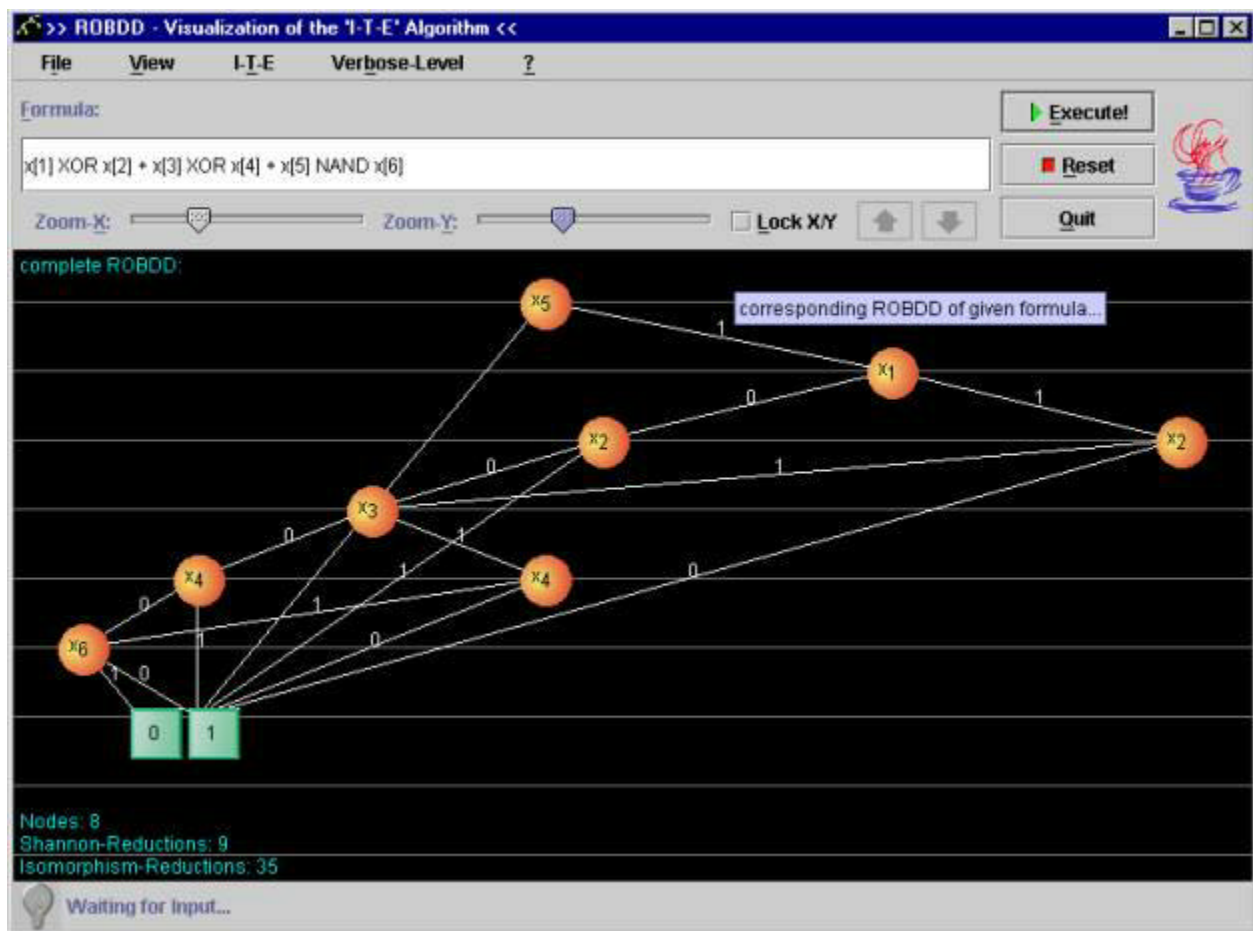


Figure 2. Screenshot after sifting

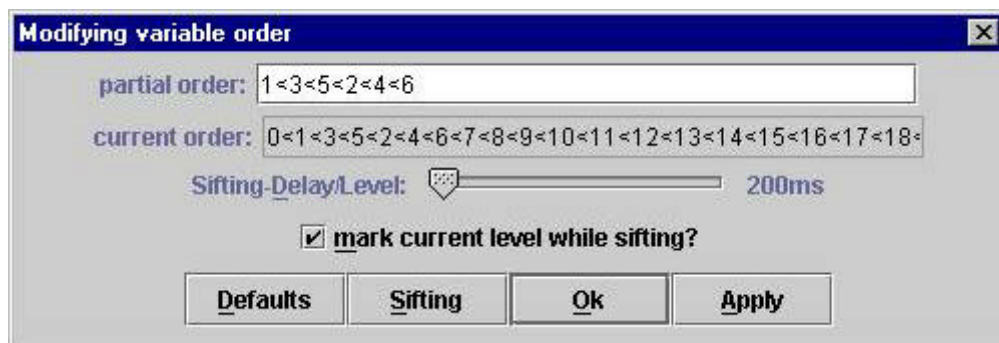


Figure 3. Setting the global variable ordering

Acknowledgment

The author likes to thank Jochen Römmeler for his help with the implementation of the methods discussed in this paper.

References

- [BRB90] K.S. Brace, R.L. Rudell, and R.E. Bryant. Efficient implementation of a BDD package. In *Design Automation Conf.*, pages 40-45, 1990.
- [Bry86] R.E. Bryant. Graph - based algorithms for Boolean function manipulation. *IEEE Trans. on Comp.*, 35(8):677-691, 1986.
- [DB98] R. Drechsler and B. Becker. *Binary Decision Diagrams - Theory and Implementation*. Kluwer Academic Publishers, 1998.
- [DR01] R. Drechsler and J. Römmeler. Implementation and Visualization of a BDD Package in JAVA, Internal Report, University of Freiburg, Germany, 2001.
- [FMK91] M. Fujita, Y. Matsunaga, and T. Kakuda. On variable ordering of binary decision diagrams for the application of multi-level synthesis. In *European Conf. on Design Automation*, pages 50-54, 1991.
- [Hoe99] S. Höreth. TUDD: Darmstadt University Decision Diagram Package Release 0.8b. University of Darmstadt, Germany, 1999.
- [Hud98] S.E. Hudson. Constructor of Useful Parsers, <http://www.cs.princeton.edu/~newlineappel/modern/java/CUP/>. 1998.
- [Lon93] D.E. Long. Long-Package Sun Release 4.1 Overview of C Library Functions. 1993.
- [PS95] S. Panda and F. Somenzi. Who are the variables in your neighborhood. In *Int'l Conf. on CAD*, pages 74-77, 1995.
- [Rud93] R. Rudell. Dynamic variable ordering for ordered binary decision diagrams. In *Int'l Conf. on CAD*, pages 42-47, 1993.
- [Som97] F. Somenzi. CUDD: CU Decision Diagram Package Release 2.1.2. University of Colorado at Boulder, 1997.

Rolf Drechsler

Institute of Computer Science
Albert-Ludwigs-University
79110 Freiburg, Germany
drechsle@informatik.uni-freiburg.de

[Back to contents](#)

Chaucer in Cyberspace: Linking the Medieval and the Modern Worlds through Technology

At the University of Colorado at Colorado Springs, English 395, Chaucer, examines the major works of Geoffrey Chaucer, consummate poet of the English Middle Ages and arguably the greatest poet of the English language, with a special focus on the *Canterbury Tales*. Works are read in the original Middle English; a short introduction to the language precedes the study of the poetry. The primary objectives of the course are to help students to better understand, judge, and appreciate medieval literature in general and the works of Geoffrey Chaucer in particular. A secondary goal is to help students to appreciate the achievements and "alterity" of the Middle Ages. The course is offered once per year and normally enrolls thirty-five to forty students.

I have taught this course every spring for almost thirty years. This spring, however, with the assistance of a Teaching with Technology Partnership Grant from this campus's Teaching and Learning Center, I redesigned the course and added a significant dose of technology. This report relates the history and results of my redesigned Chaucer class.

The primary goals of the course redesign were threefold. First, I wanted to enrich this course for my students with a greater appreciation for the achievements (e.g., music, art, architecture, manuscript illuminations, philosophy, etc.) of the Middle Ages (many websites exist which make this possible) and, thus, place Chaucer more completely in his milieu and establish a better appreciation for his genius. This is something I always try to do through lecture; but constraints of time and resources place limitations on what is possible without neglecting the literature itself. Next, I wanted to introduce my students (and, in a certain sense, myself) to the research resources of the World Wide Web and to require them to develop some discernment about what can be found there. Also, very importantly, I wanted my students to create a *florilegium* (a "bouquet" or collection) of primary, secondary, and lecture quotations which were particularly meaningful to them; this, I believed, would force them to read and listen more attentively, to be immersed more fully in the discourse of Chaucer's medieval world. Furthermore, this *florilegium* was to be a group effort, thus forcing the students to form a discourse community outside of class time. Finally, I wanted to weave together the best of the medieval university experience (it was, after all, the medieval period that established universities) with modern technology.

With the assistance of my technology advisor, as provided by the grant, I created for my class a website, replicating in significant ways both the *Canterbury Tales* and the medieval university experience, which was the locus for all these goals.

- A. The first subpage of the site was a Prologue or introduction, which, modeled on the "General Prologue" of the *Canterbury Tales*, set forth the goals of the class (including a reading list) and of the redesign and required the students to introduce themselves, again in the spirit of the "General Prologue," to one another and to me.
- B. The second subpage of the website was a *Compilatio* (medieval Latin for compilation or collection). On this subpage I provided the students with a link to the Chaucer website at Harvard University, an invaluable source of information on the poet and his work. Students were then required to augment the subpage with links resulting from their own Internet searches and with evaluations of the links they had established. Directions on how to establish links and guidelines for the evaluation of websites were provided for the students. Links were registered on a first-come/first-served basis; and no duplications were allowed. Here again, it should be noted, the technology mirrored the *Canterbury Tales* since the poem itself is a *compilatio* of tales intended to be shared with and evaluated by fellow travelers. This part of the project enriched the course with multimedia experiences both during and outside of class.
- C. The third subpage of the Chaucer website was a *Florilegium* or collection of quotations from Chaucer's work. Students were required to quote passages, which had special meaning to them or which they regarded as important to an appreciation of Chaucer, from the works we were reading and briefly to explain their choices. Students were also encouraged to comment on one another's

selections. Again, no duplications were allowed.

- D. The final subpage of the website, based on the *disputationes* of medieval universities, required students to respond thoughtfully to different prompts which addressed crucial aspects of Chaucer's artistry and to comment on one another's observations through a threaded discussion forum. "In medieval universities the masters or professors had two primary duties: to give lectures (*lectiones*) and to conduct disputations (*disputationes*)."*Lectiones* were synonymous with teaching; "the master commented on and interpreted a text or subject, chapter by chapter or topic by topic, weaving his own thoughts into the presentation or *glossa*." With *disputationes*, the master selected a subject or topic to be disputed, opened the discussion, and presided. Participants could argue for or against the proposition. "After the last objection and answer, the master gave his final exposition, proposing his solution, called the *determinatio*, usually given the morning of the first school day...following the dispute."

These features of medieval teaching and learning are still at the heart of the modern university experience. *Lectiones* take care of themselves (with work); but *disputationes* are less easily sustained in a course, largely because of time constraints. This is where technology helped reflect the medieval university: *disputationes* in the Chaucer class were conducted electronically, thus uniting the best of both the medieval and the modern worlds and, as planned, building a strong community of learners and a sense of camaraderie outside class time.

Near the end of the semester, the students were given a survey to evaluate the technology used in the class, to make suggestions for improvement in the future, to rate the success or failure of the experiment, and to comment on their responses. Space does not allow a detailed report on the survey. Overwhelmingly, however, the students responded positively. For example, they strongly agreed that the *Compilatio* made them more aware of Chaucer resources on the Web and taught them to distinguish better the quality of Chaucer resources available there. Equally strongly, they reported that their contributions to the *Florilegium* and the opportunity to comment on the contributions of their fellow students made them closer readers of Chaucer's works and increased their appreciation of his artistry. Similarly, they insisted that their own contributions to the *Disputatio* and the requirement to respond to their classmates' comments assisted their learning. One student summed up rather nicely the prevailing sentiments of the class: "I think all three [the *Compilatio*, the *Florilegium*, and the *Disputatio*] were effective learning tools, but they were also a fun and interesting way to be more involved in the class.... Keep using the technology."

The last exercise the students had on the website was to select a student pilgrim whose contributions to the website were of "best sentence and moost solaas"; once again, this exercise mimicked the plan of the *Canterbury Tales* which turns the storytelling of the poem into a contest and calls for the pilgrims to select a winner according to the criteria listed above. Far from fostering a sense of rivalry, this feature sealed the sense of camaraderie which had been created during the semester—and led to a great deal of humor and fun.

One final comment. When I proposed my grant, I insisted that I wasn't interested in whistles for the sake of bells and whistles or technology for the sake of technology but, rather, that I would judge the experiment a success or failure on the basis of its effectiveness in producing a better Chaucer class. I am now satisfied that it did that.

Professor Thomas J. Napierkowski

Department of English

University of Colorado at Colorado Springs

1420 Austin Bluffs Parkway, P.O. Box 7150

Colorado Springs, CO 80933-7150 USA

tnapierk@mail.uccs.edu

Tel: +1 719 262 4007

Fax: +1 719 262 4557 (719)

[*Back to contents*](#)

Conference announcements

International Conference on Distance Education in Virtual Space

19-21 September 2001

Yalta, Crimea, Ukraine

Further information:

Victor Cauk, Organizing Committee Secretary

Information Marketing Department

Kharkiv State Technical University of Radioelectronics

14, Lenin av., 61166, Kharkiv, Ukraine

imd@kture.kharkov.ua

Tel: +38 0572 40 95 15

[*Back to contents*](#)

