A PRACTICAL COMPUTER-ASSISTED LEARNING FRAMEWORK FOR MIXED ABILITY CLASSES

Michael Trimikliniotis, Socrates A. Mylonas

ABSTRACT
Teaching introductory courses in Computer Science and Management Information Systems to students is often hindered by a variety of factors. Computer-assisted learning methods may provide solutions to such problems. Feedback from the students in the development of these methods is of great importance to their overall success.
The essence of the work discussed here, is to create an environment to support computer related courses, addressing the needs of both students and educators. The general aim has been to develop a flexible, simple, modular and consistent system considering not only the administrative framework, but also the actual educational content. This paper will describe the main modules of this system and present examples of their use. The modules have been tested in a number of different computer courses, and feedback from the students was collected in the form of questionnaires. The results of this survey will be presented and discussed in this paper.

KEYWORDS
Computer-assisted learning, web-based learning, multimedia tools, framework, self-assessment.

INTRODUCTION

The revolution in the distribution and sharing of information, which began with the booming of computer technology and the Internet, has influenced the traditional ways of delivering knowledge. New ways of teaching a course have been introduced, but the whole approach to learning has been altered. Terms like Computer Based Teaching-Learning systems (CBTLS) and Virtual Learning Environments (VLE) are widely used these days attempting to replace the classroom environment or simply to add to its functionality and effectiveness [Sandy et al., 2000].

One of the challenges of educators in tertiary education is handling the diversities in a class and providing students the ability to work in an environment that encourages learning and stimulates each student's individual abilities. At the same time, giving support and guidance to those that face difficulties adapting to their new environment appear to be among the objectives of several higher education institutions [Roy, 1996].

Computer-assisted and computer-based learning have been used for a number of years to address some of these issues[Randolph et al., 2000, Brent et al., 2000]. These attempts have not always been successful, because of various factors such as the attitude of participants (students, educators, management, etc.) and the quality of the available material.

Several commercial products have emerged and have been used at various institutions. Feedback on their success is often inconclusive[Randolph et al., 2000], which supports the view that the use of technology alone cannot make a positive impact in the educational framework[Sandry et al., 2000, Tsichritzis, 1999].

The work presented in this paper is primarily concerned with the application of technology for teaching and learning of Computer Science and Management Information Systems courses at college level. Since several of the issues raised are often encountered elsewhere (e.g. in other fields of study or other educational levels, like in secondary education), there is scope for the generalization of this work. The system described is still being developed, with new modules being added. It is a practical system addressing specific needs and has not been designed with a specific pedagogical model in mind.

Motivation

The effort started off as a fragmented attempt by various members of faculty of our department to provide materials and assistance to students enrolling in computer science and information technology courses, by using their individual abilities (system development, programming skills, database management and programming, multimedia development, etc.). It soon became apparent that items developed by one faculty member could be utilized by others. Later on, faculty members of other departments expressed an interest, which raised the need for the formal study of the requirements of such a system [Randolph et al., 2000, Brent et al., 2000].

The nature of the problem

The primary participants of the teaching/learning process are the learners (students, pupils) and the educator (professor, lecturer, teacher), as seen in Fig. 1. These interact by means of a learning environment, a conceptual term, which in traditional education is associated with the educational establishment (school, college, university) and its facilities (classrooms, library, laboratories, policies, assessment methods, etc.).

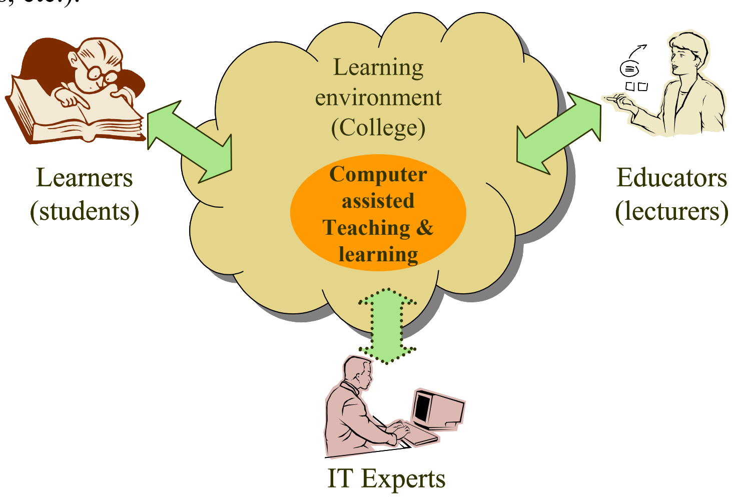

Figure 1. Generalised educational environment model

In a tertiary educational institution, both students and lecturers encounter difficulties that cannot be addressed effectively by 'traditional' instruction[Roccetti, 2001]. A number of issues and needs have been identified, based on informal interaction with students and faculty, and more formal feedback methods, such as class assessment questionnaires.

Students attending our college come from countries with quite different educational and cultural background. These factors are very influential in their way of learning, especially during their first year of study. A number of students wish to learn and become experts in their field of study. Others wish to perform just adequately to graduate, whereas others have a specific aim, usually related to their current employment. Finally, it is evident that their individual expectations play a crucial role in their motivation. Frequently, the use of technology and computer-based multimedia for teaching and learning is in itself a motivational factor, as it makes classes more interesting and exciting.

Lecturers, on the other hand, recognize these difficulties, but often cannot provide easy solutions within traditional classroom-based instruction. The number of examples one can present in class is limited by the time available to cover a given syllabus. Similarly, the number of supervised exercises and projects that may be assigned is sometimes limited by the available time to discuss and correct them. Technology and the availability of several computer-based tools sometimes help, but often the readily available materials do not coincide with the specific course requirements and are of limited use. Lecturers may learn how to use software tools, but often find that many of these, though quite comprehensive, have a steep learning curve before they can be applied even for simple tasks. Lecturers highlight the need for tools that are easy to use and learn, requiring progressive effort to support teaching in innovative ways.

One may actually wonder how a system can satisfy the needs of every individual student, or lecturer, which raises the issue of *adaptation* [Marzo et al., 2000, Tsichritzis, 1999], addressing the need to find out how to match the learning environment to the objectives and skills of the student or teacher, in such a way that does not appear hostile to its users.

The application of technology requires the involvement of technology experts, either directly (e.g. system and network administrators) or indirectly (e.g. software developers). Their role is not considered an integral part of the learning environment (Fig. 1), but in our case they play a vital role as solution providers[Lewis et al., 1999]. System and network administrators are often concerned about the impact of the installation and use of new software on computer and network security, the need for supporting and upgrading software and the provision of assistance and administration services to the users[Brent et al., 1999] (e.g. registering new students to use the available services, removing old students, etc.).
In this manner, the need for learning new skills by the lecturer is reduced, encouraging him/her to use these new tools, whereas students require little or no training, as most are, to an extent, familiar with the web and other Internet technologies.

Proposals

With reference to Fig. 1, computer-assisted teaching and learning may be considered part of the learning environment. In the case of traditional learning its role is limited, whereas in the case of distance learning, its extent may cover almost the entire scope of the learning environment. In our case, the aim is not to replace traditional instruction, but to complement it with new techniques in order to reduce the effect of the problems identified in the previous section. In our attempt to find what the major needs were, the following general requirements were identified (See Fig. 2):

1. A framework to encompass the various tools
2. A variety of course-related generic services for the distribution of materials and information, such as:
    a. A course main page
    b. Posting of on-line announcements
    c. Project and assignment details
    d. Making available materials, examples and notes
    e. Tools to encourage/facilitate communication between the students and the lecturer (e.g. e-mail, on-line message board, etc.)
3. The creation, presentation and distribution of specific course-related materials:
    a. Multimedia files
    b. On-line exercises and quizzes
    c. Self-assessment exercises
    d. On-line submission of projects and exercises for the lecturer to grade (instead of handing over diskettes or sending large files via e-mail).
    e. Announcement of grades to students
4. Administrative services:
    a. For the lecturer to update their course materials, post announcements, contact students, etc.
    b. For the network administrator to install new lecturers, new students and new courses

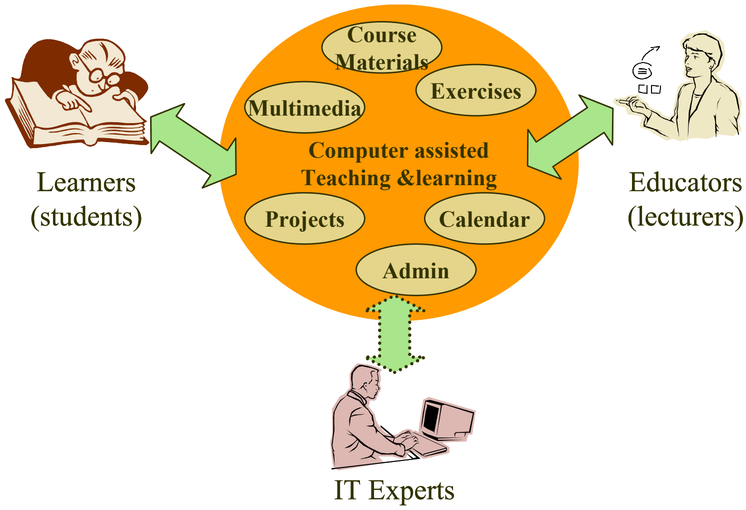

Figure 2. Computer-assisted teaching and learning

Two important observations were made at this stage. The first was the volatility of the requirements, with requests for new services being discussed frequently. The second was the need to separate the general framework and some of the administrative services, which required less maintenance, from the provision of content, which made the overwhelming majority of the requests. For this reason, keeping the system modular and maintaining a unified approach in the interconnection (interfaces) of its various components is essential.

The provision of the framework for this system could have been provided by a number of commercial tools, such as WebCT. We have discussed the experiences of lecturers that tried such systems and discovered that in general, a lot of time was spent in learning how to use the various tools provided. In fact, what would actually suffice (and indeed be preferable to them) is a much simpler and more specific environment, with only the required facilities at hand, that could be used immediately with little or no training. It must be stressed that many commercial packages require the implied complexity and extensive facilities, because they target distance learning, which is different in many respects from the requirements of the system described here.

SYSTEM OVERVIEW

The structure of the proposed implementation is shown on Fig. 3.
To implement the overall framework of such a system, the following additional technical issues had to be considered:

1. The software platform(s) on which to develop the system
2. The programming environment/languages to be used
3. How to maintain the proposed framework simple, yet expandable
4. How to interconnect the various units in the system
5. How to integrate the various blocks (modules) of the system

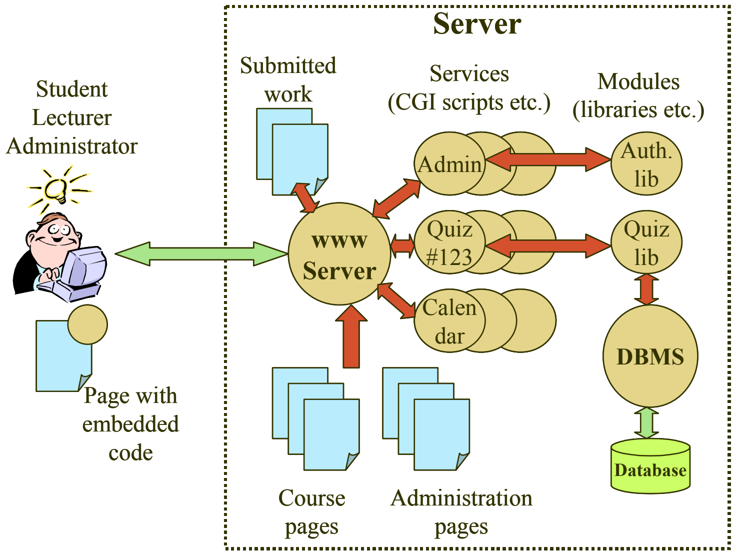

Figure 3. System structure

Server structure
The issue of cost has also been considered. Most commercial solutions cost several thousands of Euros, which may be an inhibiting factor for application in institutions with limited budget or unsure of the benefits of this system. It was therefore decided to develop the solution using freely available software platforms. The prototype implementation is currently running on relatively inexpensive hardware (a Pentium III server with 128MB of RAM, running Linux Red Hat 7.2). The web server making most of these facilities available is Apache 1.3.22[Laurie, et al., 1999].
Dynamic parts of the system, such as questions and answers for quizzes, calendar events, grades, administrative details (e.g. student and lecturer details) are maintained in a database[Trancoso, 2001]. Commercial Database Management Systems (DBMS), such as Oracle could have been used, but to keep in-line with the inexpensive nature of the system, a freeware DBMS was used instead (see Fig. 3).
Some parts of the system consist of normal web pages, (written in HTML), but the majority of the system's modules are in the form of dynamic pages, composed of:
(a) A static *template* page, containing formatting and layout information
(b) An executable *script*, performing the necessary actions to produce the dynamic elements of the page (e.g. randomly select questions in a quiz) merging them with the template and delivering it to the user, processing user input through a form, or both.

Most services provided are implemented as a single script, combined with a number of templates to provide different responses in the case of an error, successful completion of the task and incomplete data. This structure is also implemented in the administrative part of the system.

Some services may combine embedded code, e.g. JavaScript [Flanagan, 1998], which is executed by the computer of the user. For example some self-assessment exercises, after being downloaded require no further interaction with the server. Other services rely on multimedia files.

Simpler tasks, used by more than one service, have been grouped together into modules (libraries). The majority of the modules were written in the Perl 5.0 programming language, which provides good facilities for both interacting with the web and with several DBMS, while being multi-platform[Wall, 2000].

The system has been tested and found to support over sixty computers simultaneously without noticeable performance degradation.

Administrative services

The server must support the work of the administrator who manages the server. The main tasks required are:

- Add/Remove a lecturer (or a student): The lecturer is registered in the system and is given the necessary privileges to maintain, store and remove materials
- Add/Remove a course: This is a quick way to create a website for a course, containing a range of standard elements and assigned to one of the existing lecturers (Fig. 4)
- Install/Remove a system module. This option permits the administrator to add new system modules so that they become available to all courses

Faculty services

Faculty members have a set of administrative tools for each course they have been assigned, through a web page. Authorization is required to gain access to these services, which include:

- Enroll a student to a course
- Post an announcement for a course
- Advertise an assignment for a course
- Upload materials and web pages for a course, including test banks for quizzes
- Update existing course pages
- Publish grades for a given course

Student services

Students generally can view all available material for any course. Additionally they may perform the following tasks if they are enrolled in the course, by logging in:

- Submit an assignment
- Read, search and place questions and answers to the on-line posting list
- Download materials available for the course

EXAMPLES OF SYSTEM COMPONENTS

As already mentioned, the system is expandable. The following are examples of the facilities provided at present.

Course materials

Upon creation of a course, a standard format is created, which includes a Main Page, a Weekly Schedule (which includes links for a web page for each week), Announcements, Assignments, Monthly Calendar, Downloads, and Student Services (Fig. 4). This is available instantly and the lecturer may utilize it without spending any additional time developing its structure. He/she may post announcements, schedule projects, etc. Gradually the lecturer may populate the page with materials, replacing or updating the simple pages available. Through these pages the lecturer may present examples to assist students with difficulties and provide additional materials for those who want a challenge.

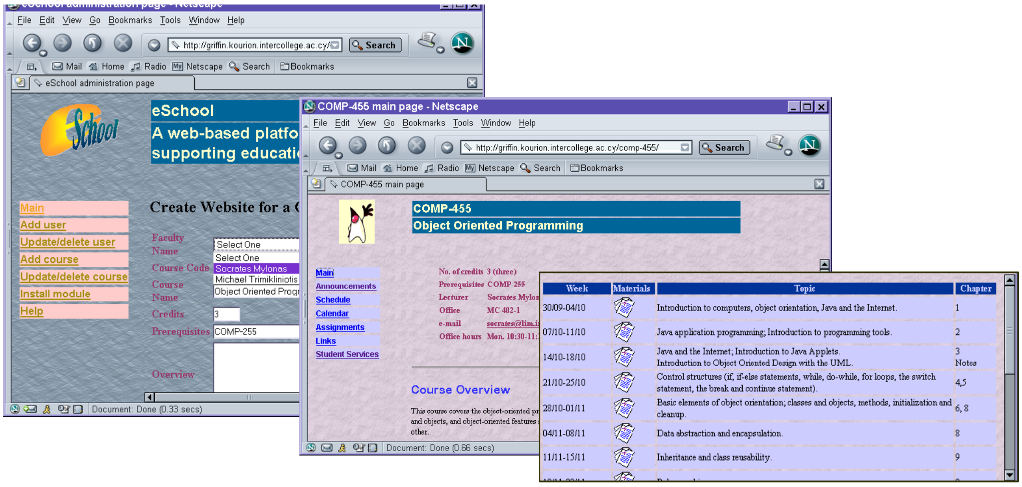

Figure 4. Sample administrative page (create course) and resulting website pages (structure, main and schedule)

Testing and assessment
Two types of assessment tools have been provided, self-assessment exercises and on-line quizzes.

On Line Interactive System for Self Assessment (OLISSA)
This module consists of exercises which students use as many times as they like to test their knowledge. Students attempting exercises from a book, often complain that the book does not provide them with the answer to assess their knowledge and that the lecturer does not provide sufficient homework, implying that feedback and explanatory comments should be preferably provided in real time [Andre et al., 1997, 1998, 1999].

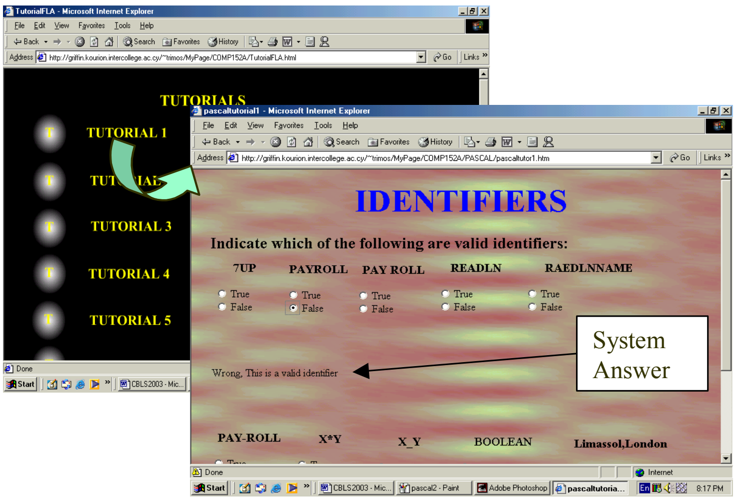

Figure 5. Sample on-line self-assessment exercises for programming courses

Module *OLISSA* has been developed to address this need. It consists of a standard internal sub-module that delivers to students a web-based interactive page with exercises. *OLISSA* is primarily a framework. The visual appearance and mode of the exercise may vary to suit the needs of a particular course or subject. Students access the module through links on the web page of a course (Fig. 5). In general, there are exercises, that the student must attempt and at each answer he/she may receive feedback and comments. Graphical/visual/audio effects are added to improve the interactivity between user and machine as well as on-line notes, connected via hyperlinks for support.

Questions may be either multiple-choice selected from a database, or more specialized exercises, depending on the subject. For instance in Fig. 5, if the student provides an incorrect answer, the correct one is displayed on the screen. In Fig. 6, the student is encouraged to provide five correct answers in a row before advancing to more demanding questions. A progress bar is used to motivate the student, but the correct answer is never revealed.

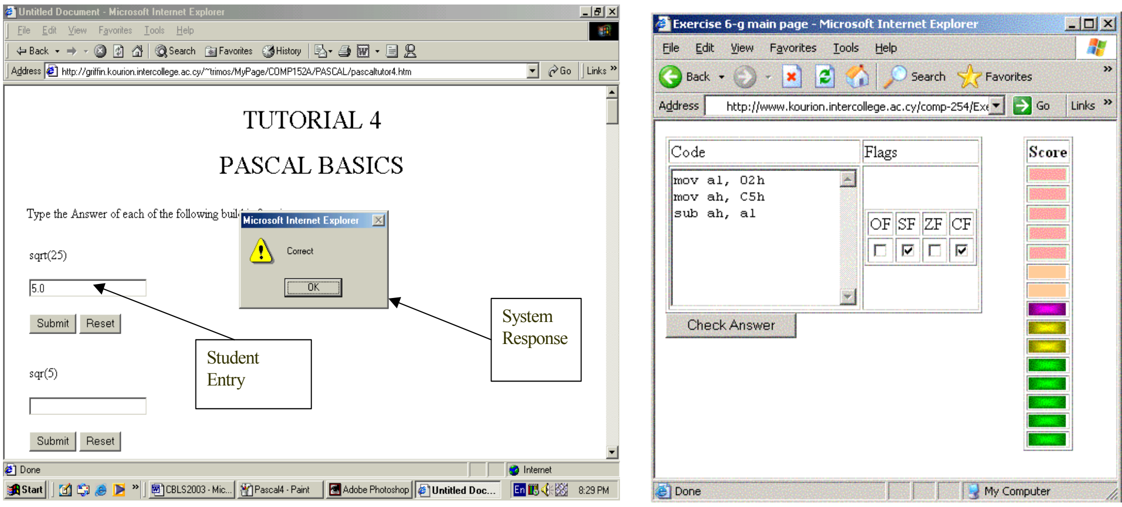

Figure 6. Sample on-line self-assessment exercises for programming courses

Database of Exercises with automatic Assessment (DExA)

This assessment tool is primarily used as an alternative to multiple choice quizzes. Students access a web-based script that returns a set of multiple-choice questions, extracted randomly from a database and merged with a template page and presented to the student. After completing the exercise, the answers are submitted to the server, which automatically assesses their correctness and informs the student and the lecturer of the obtained grade.

Communication tools

With the *announcements* and *assignments* pages (Fig. 7), lecturers may post announcements or assignments and have them immediately available on-line for students to access at all times.

The on-line *submission tool* permits students, especially those taking computer programming courses, to submit their assignments without using physical media (e.g. diskettes), making possible the submission of work without being present at college. Upon the expiration of a deadline, it is no longer possible to make a submission. The on-line *bulletin board* is intended for students to post questions and for answers to be provided, either by the lecturer or by other students. In this way, knowledge could be accumulated so that repetition may be avoided.

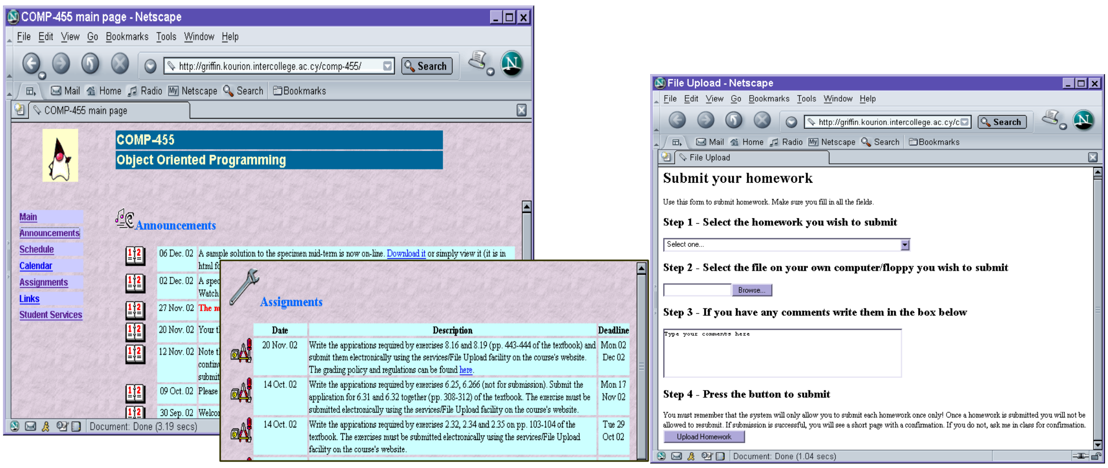

Figure 7. Examples of the available communication tools

Multimedia components, Virtual Movie-Based Learning (VMBL)

This part of the designed system exploits the animation capabilities of computers to help students learn how to use Software Application Packages. Factors influencing student learning include computer phobia, irresponsiveness to traditional teaching methodology (listening to the lecture and then work on a computer, follow written instructions on how to perform a task) and culture (e.g. being hesitant to ask the lecturer for help)[Marzo et al., 2000].

Based on these problems, the idea of a movie-based system to help the students was born. Using inexpensive commercially available software the lecturer could capture the steps/routine needed for the execution of a specific task in a software package (e.g. how to create a relationship between two tables in Access) and store them in a video file (.AVI, .MPEG, or QuickTime). These files can be stored in a folder together with a simple web page to enable easy access to them, by selecting a link to play the corresponding video demonstration (Fig. 8). Students may repeat this process as many times as necessary and then try the exercise themselves. Video and audio files usually have large size, requiring large bandwidth to transmit. Instead of relying on state-of-the-art compression techniques, a simpler approach was used. Students, while in the labs, may use the material *on-line* through the local web server. If they are at home, they may access the *same material* on conventional CDs. Using this system requires only basic training on how to use a web browser and how to insert and access a CD. The example presented in Fig. 8 demonstrates the page used for the teaching of Office packages.

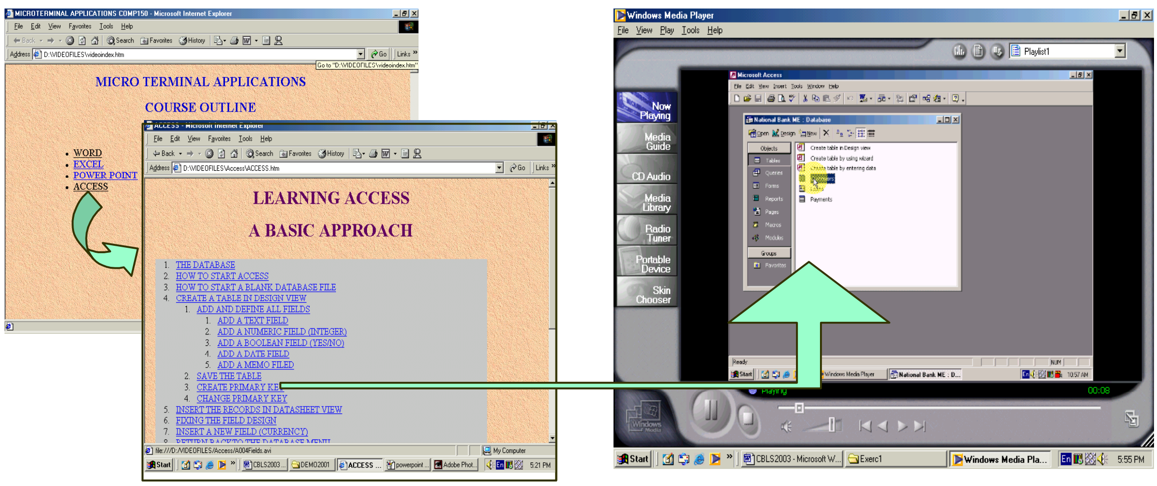

Figure 8. Accessing a VMBL element

VMBL allows classes with students having diverse computer abilities to be formed, allowing students from different majors, and different computer backgrounds to take the same course together.

EVALUATION AND RESULTS

The success of different aspects of the system was investigated for different student groups, over a two-year period. The survey is an on-going process that began in June 2000.

Methodology

Two independent surveys were conducted, one to study the effectiveness of VMBL (Fig. 9) and the other to study the usefulness of the general web-based framework and tools (Fig. 10). Questionnaires were handed out in class. Evaluation by lecturers is still at an early stage and results are not yet available.

Hypothesis

In the case of the VMBL, it is assumed that its use accelerates the learning process, especially of weak students. The grades of the students using VMBL were compared to the grades of students who did not use this technology, including those that took the same course in the past, where this technology was not available.

In the case of the general framework and tools, it is assumed that some of the tested modules motivate students to learn. The objective was to identify which aspects of the system were perceived by students as helpful and whether there was correlation between their preferences and their major, etc.

	PC (PIII)	PC (PIV)	Average
Previous knowledge	2.88	2.85	2.87
Current Knowledge	3.69	3.75	3.72
Standard on-line help	3.50	3.30	3.40
VMBL Usefulness	4.31	4.50	4.41
VMBL Delivery	3.50	3.50	3.50
Speed	2.63	4.00	
Size	1.00	1.60	
Quality	2.88	4.00	
Sample size	20	32	

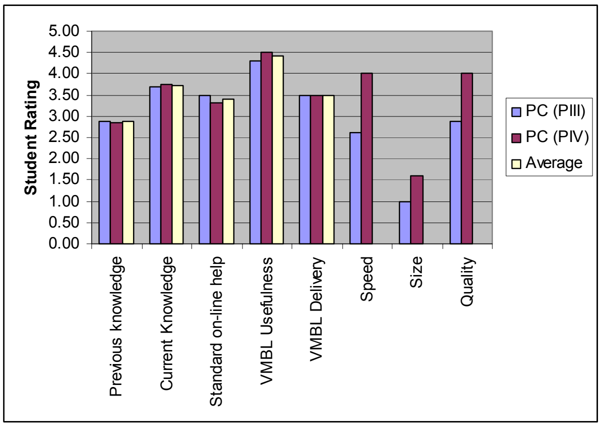

Figure 9. Student survey results on VMBL

Questionnaires
Questionnaires were handed out to students requesting their opinion on various issues related to the tools using a scale from 1 (poor) to 5 (excellent), with an additional option (0) for students who either did not understand or did not want to answer a particular question.
For the VMBL, the questionnaire was divided into three sections. The first investigating the knowledge of the student on the particular computer topic, the second covered educational issues and the third part technical issues (Fig. 9). For the general framework, the questionnaire gathered student views on different aspects of the system and some demographics (year of study, age, major). Additionally, students were prompted to add written comments on each aspect (Fig. 10).

Sampling Method
Due to the limited nature of the available population, specific classes taught by the authors were used. All classes were mixed-ability, multi-culture classes with students coming from different countries, having different majors, and different level of command of the English language (used for instruction in the college). Furthermore, in some classes, especially where VMBL was tested, some students were freshmen, whereas others were junior, or even senior. This diversity of knowledge, language understanding, and age has been thought to be a positive factor for the evaluation of the system. It was not possible, however, to test in a given class all aspects of the system with all possible types of students, since there is a certain bias caused by which students (major, year, etc.) enrol in each class. For the VMBL, there were 52 responses from a total number of 68 students. For the general framework, 45 students responded.

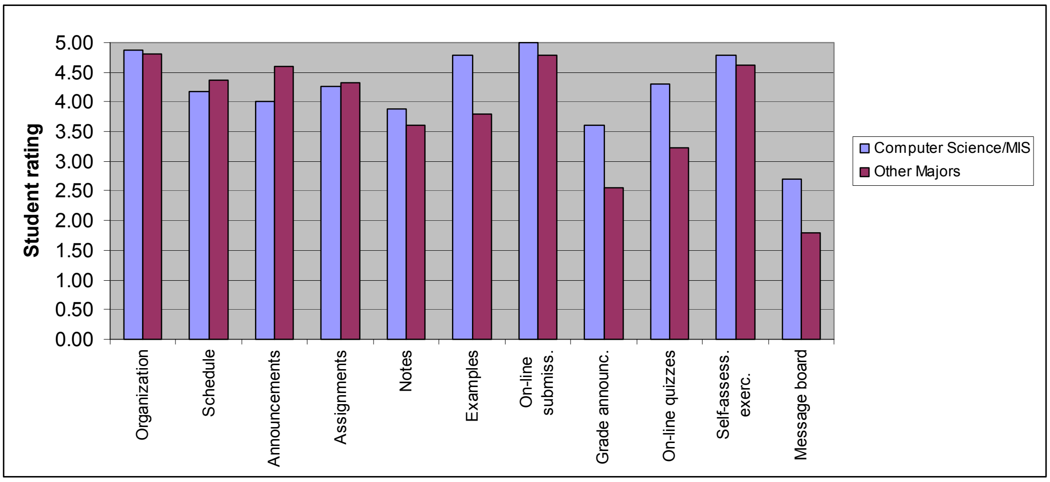

Figure 10. Student survey results for the various system units

Discussion

As seen from the results reported in Fig. 9, students recognised the pedagogical contribution of VMBL and reported an improvement in their knowledge after using the system. This is also evident in the grade comparison among the students using VMBL and those not using it. A considerable decrease in the number of failures and a significant increase in the average grade were noted for students using the system (Fig. 11).

Results were separated into two groups. The first group used slow Pentium III computers and rated the system as slow with low quality, whereas the second group used faster Pentium IV computers and found it more satisfactory. It is concluded, therefore, that most of the technical limitations have been reduced with the improvement of technology.

On the general framework, it was noted that over 85% of students, irrespective of their major, have found the structure of the website(s) easy to use. An average rating 4 to 5 (very good to excellent) was given to the Schedule, Announcements and Assignments pages, for the On-Line Homework Submission tool and the Self-Assessment Exercises. These indicate modules, which are likely to be used by students, irrespective of their area of study. No correlation was found between the student's year of study or age and their preferences. However, a correlation was found between a student's major and their rating to the On-Line Examples, Other Links, Grade Announcement, Quizzes and the Message Board. Computer Science and MIS majors rated these about 20% higher than students with other majors. This may be related to the greater familiarity of these students with technology, which motivates them to explore these aspects beyond the strict class requirements. The Notes page was rated average, perhaps because in all classes tested, students have equivalent printed notes and do not have to rely on the on-line material. A reason why the Bulletin Board was rated poorly is very likely because it contained a limited number of questions and answers and was, hence, of limited practical use.

CONCLUSIONS AND FUTURE WORK

In this paper a framework for the development and use of computer-assisted teaching and learning materials has been presented. A number of software tools to support traditional teaching methods have been developed and evaluated. Although these tools are undergoing further testing and assessment, early results demonstrated that students and educators react positively to their use, especially communication tools, self-assessment methods and animated tutorials.

Students support the use of these tools, because they liberate them from time and space constraints while enabling them to capitalize on those matching their personal learning style and speed.

Grade	Traditional		with VMBL support	
	Number	%	Number	%
A	76	38%	24	35%
B	48	24%	22	32%
C	29	15%	7	10%
D	18	9%	13	19%
F	28	14%	2	3%

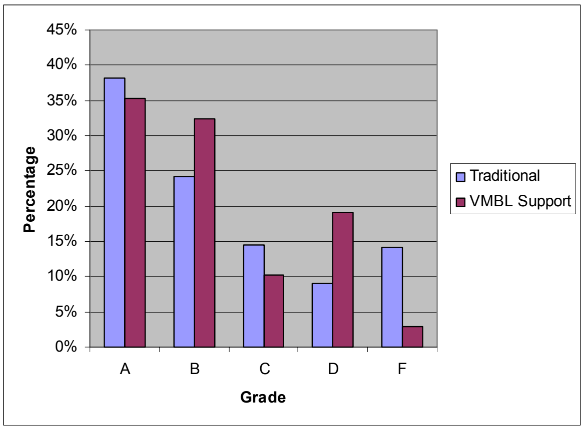

Figure 11. Grade distribution for traditional instruction and with VMBL support

Lecturers recognize the advantages of having on-line tutoring systems to support the learning process of their students. Their concern on the difficulties associated with the development and provision of these materials is not unjustified. It is in this area that IT specialists may offer their expertise. As computer-based learning techniques mature, the demand for content will increase and the cooperation of educators and IT specialists for the provision of modular, generic tools for content provision will become a necessity.

In the near future such tools will include 3D animated agents [Brent et al., 1999] for better face-to-face interaction. These systems possess a degree of intelligence and their aim is to support the learning process of the student [Johnson et al., 1999, Andre et al., 1997, 1998, 1999]. It is worth noting, however, that it was the simplest aspects of the system and the tools that students found most useful. One may conclude that it is not only sophisticated technology that makes an impact in class, but also simple ideas, which can be easily used even by students with little experience. As this paper demonstrated, these on-line tools have a positive impact in student learning, more prominent to students with poor performance, playing a complimentary role to traditional instruction.

REFERENCES

Andre, E.; Rist, T.; & Muller, J. (1997). “Adding Animated Presentation Agents to the Interface”. Applied Artificial Intelligence, ACM, pp 79-86.

Andre, E.; Rist, T.; & Muller, J. (1998). “Guiding the user through Dynamically Generated Hypermedia Presentations with a life-like character”. IUI 98 San Francisco ACM, pp21-28

Andre, E.; Rist, T.; & Muller, J. (1999). “Employing AI methods to control the behaviour of animated interfaced agents”. Applied Artificial Intelligence, 13:415-448.

Brent, D.; Bares, W.; Callaway, C. & Lester, J. (1999) “Student-Sensitive Multimodal Explanation Generation for 3D Learning Environments”. American Assoc. of Artificial Intelligence (www.aaai.org)

Britain, S. & Liber, O. (2000). “A Framework for Pedagogical Evaluation of Virtual learning Environments”. University of Wales – Bangor. http://www.jtap.ac.uk/reports/htm/jtap-041.html

Flanagan, D; (1998). “JavaScript The Definitive Guide, 3/e”. O’ Reilley, ISBN 1 56592 392 8.

Jackson, R., and Fagan, E. (2000) "Collaboration and Learning with Immersive Virtual Reality", CVE (2000), San Francisco, CA USA, ACM September 2000, pp. 83-92.

Johnson, L; Rickel, J. & Lester, J. (1999) "Animated Pedagogical Agents: Face-to-Face Interaction in Interactive Learning Environments". Internat. Journal of Artificial Intelligence in Education, pp. 47-79.

Laurie, B; Laurie, P. (1999). "Apache: The Definitive Guide", O' Reilley, ISBN 1-56592-528 9.

Marzo, J., & Pena, C. (1999). "Learning Style for A WEB-Based Educational System Using Software Agents". http://cbl.leeds.ac.uk/~jas/marzo.html

Roccetti, M. and Salmoni, P. (2001). "A Web-based Synchronised Multimedia System for Distance Education ". SAC, Las Vegas, NV, ACM February 2001, pp 94-98

Roy, M. (1996) "Collaborating across campuses: the virtual university experience ". MCB University Press, Campus-Wide Information Systems, 1996, Vol. 13, No 4, pp.6-10

Trancoso, P. (2001). "eEdu: a Dtatabase-Driven Tool for Online Education Material". Proc., 8th Panhellenic Conference in Informatics, vol. 1, pp. 362-371, Nicosia.

Tsichritzis, D. (1999) "Reengineering the University". Communications of the ACM, June 1999, Vol. 42, No 6, pp. 93-100.

Wall, L; Christiansen, T. and Orwant, T (2000). "Programming Perl, 3/e". O' Reilley, ISBN 0 596 0027 8.

Michael Trimikliniotis, Socrates A Mylonas
Intercollege, Limassol Campus
92 Ayias Phylaxeos Str. P.O.Box 51604, CY-3507
Limassol, Cyprus
Email: trimos@lim.intercollege.ac.cy, socrates@lim.intercollege.ac.c y