

# WWW-based Educational Environments for Software Engineers\*

VINCENT P. WADE, JANE B. GRIMSON and CONOR POWER

*Department of Computer Science, Faculty of Engineering and Systems Sciences, Trinity College Dublin, Ireland. E-mails: {Vincent.Wade, Jane.Grimson, Conor.Power}@cs.tcd.ie*

*Increasing class sizes, expanding curriculum, added time pressure on students and lecturing staff, contention for library resources and study space as well as the cost of educational administration are common problems being experienced within Irish and UK universities today. To alleviate some of these pressures and to improve the overall quality of learning within a university, there is a discernible movement toward enabling and encouraging student-centred learning. Typically, a range of information technology (IT) services are used to support learning within universities, ranging from slide preparation software, through e-mail communication between lecturers and students, to full virtual learning environments. A virtual learning environment has been developed at Trinity College Dublin, which seamlessly integrates courseware delivery, participant interaction and dialogue, access to external systems and educational management systems. This environment, which is based on World-Wide-Web technology, has been applied to the development of a course in database software engineering. The learning outcomes of students who have taken this course have been fully evaluated and compared to those of students taught using traditional teaching methods. The results show that there are tangible educational benefits to be derived, but if these are to be achieved, substantial investment of time and resources into the development of the on-line courseware is required. However, this expenditure can be justified in cases where there are large numbers of students across different degree programmes taking the course and, more particularly, where distance learning is required.*

## INTRODUCTION

THERE IS widespread agreement today amongst engineering educators and professional accreditation bodies that fundamental changes are required in engineering education [1–3]. Specific problems, which are driving this need for change, include the increasing diversity of the student intake, overloaded curricula and the changing nature of the engineering profession itself, which requires good team-working skills and the ability to communicate effectively [4]. In order to address these problems, there is an increasing recognition of the need to move away from teaching, in which students play largely passive roles, towards learning, in which students are actively involved. These problems have prompted the application of advanced technology in an attempt to promote student-centred learning in engineering disciplines as well as to relieve administrative and management burdens. From a pedagogical perspective, the traditional lecture is recognised as not being the most effective means of promoting learning in large classes. There is a need for improved dissemination of the information and a deeper exposure and interaction with the subject material than can be achieved in the traditional classroom environment [5]. From a pedagogical management perspective, traditional academic lectures can result in a high contention for library resources

(especially at project and assessment deadlines) and also leads to difficulties in tracking, monitoring and testing/evaluating student learning in large classes. Applying World-wide-web (WWW) technology with specific reference to best practice in instructional and management design promises greater educational effectiveness and efficiency.

## INFORMATION TECHNOLOGY AND EDUCATION

There have been many experiments in IT-based education in the US and in Europe [6, 7] and many institutions have developed IT-based educational material, including for example, WWW educational sites and multimedia materials [8]. Over the last two years WWW-based educational delivery systems have been introduced to US and European Universities, e.g. TopClass from WBT Systems, WebCT from UBC-Computer Science, and Learning Space from Lotus [9]. These systems offer storage, management and a student-oriented desktop for accessing WWW-based educational content. However the growing capability and use of authoring tools [e.g. 10] should increase the number of WWW educational resources and hence stimulate more extensive use of WWW learning environments.

Four basic principles for successful on-line (networked) learning environments have been identified, namely [11]:

\* Accepted 15 January 1999.

- media richness
- interaction
- timely responsiveness
- organisation of course materials.

New ways of stimulating and motivating students are required which exploit these 'tele-technologies' effectively [12] and early experiments have proved that course composition, presentation and assessment have to be redesigned in order to produce effective courses [13]. Tests carried out between two groups of students participating in a traditional educational course have shown that the students who had access to supplemental multimedia materials (e.g. video, audio and animation) and simulation environments [14], did significantly better than the students who did not have access to such material [15]. Students, especially those on part-time programmes, require flexibility in the timing of course delivery and would like to be able to participate in the course away from the university campus [16]. Courses should not place high demands on hardware and software [17]. Thus course delivery must be configurable to a wide range of computing environments which offer both a uniform interface and the option of personal mobility [18].

Although there is no standard classification for the usage of IT in education, a categorisation can be made based on the type of content presented and the modes in which students interact with the educational material. Eight categories are identified below.

- *Administration information/curriculum dissemination.* This typically includes the dissemination of information for several purposes including advertising the course, course descriptions, curriculum details, timetables, laboratories, assignments, and so on.
- *Student/student and student/tutor dialogue:* This entails the electronic interaction between students or between lecturer(s) and student(s). Examples of systems to support these dialogues are electronic mail, discussion groups, bulletin boards and conferencing (text and multimedia based).
- *Lecture material preparation and dissemination.* This category entails the use of IT in preparing and presenting educational material. Typical examples of IT tools for preparation range from simple applications (e.g. MS PowerPoint, WWW tools) or more sophisticated multimedia tools and WWW authoring tools such as MacroMedia Authorware.
- *Digital libraries and specialised database access.* This category includes the use of databases and other digitally stored material by students. These can be CD-ROM based, network based or on dedicated machine(s). These may/may not be accessible via WWW interfaces. Copyright issues can arise when full texts as opposed to just abstracts are made available electronically.
- *Specialised laboratory software and computer packages.* This is quite a wide category and includes software and hardware used in laboratories within engineering departments. While often not originally intended for educational use, such packages can be employed in the learning of a discipline or domain. They include computer systems for analysis and measurement in engineering laboratories, for example, software engineering tools, and tools for computer-aided design, computer-aided engineering and simulation.
- *Interactive tutorial systems.* This category includes question/answer or problem/solution based systems specifically designed for educational usage (usually self-learning systems).
- *Computer-based educational courses and virtual lectures.* This category includes computer-based training/computer-based learning/computer-based instruction packages as well as WWW-based instructional courses. Such courses may be group or individually oriented and are usually domain (subject) specific. This category therefore generally includes techniques and technologies, which may replace one or more traditional lectures. Virtual lectures are lectures which are supported via IT services, but which are not held in a specific location and time. They can be implemented as a combination of discussion lists, e-mail and bulletin boards, stored video segments and conferencing, and virtual environments.
- *Network-based virtual environments.* This category involves courses, which are entirely supported within a cyber space environment. Such environments offer student/student interaction, contact with lecturers, interactive course material, access to external systems for project completion, automatic submission of projects, and so on.

Although slightly overlapping, these categories illustrate gradually increasing sophistication in the use of IT for educational purposes. A recent survey carried out within Trinity College Dublin collated the usage of each of these categories across the entire University. Figure 1 illustrates IT usage for educational purposes, based on these categories, for eleven of the degree and postgraduate degree courses in the following Engineering departments: Civil, Structural and Environmental Engineering, Manufacturing and Mechanical Engineering, Electronic and Electrical Engineering, Statistics, and Computer Science. The key for categories of IT usage in Fig. 1 are as follows:

- 1 = Administrative and curriculum information dissemination
- 2 = Student/student and lecturer/student dialogue
- 3 = Lecture slide preparation and dissemination
- 4 = Digital libraries, specialised databases
- 5 = Specialised laboratory software and computer packages
- 6 = Interactive tutorial systems

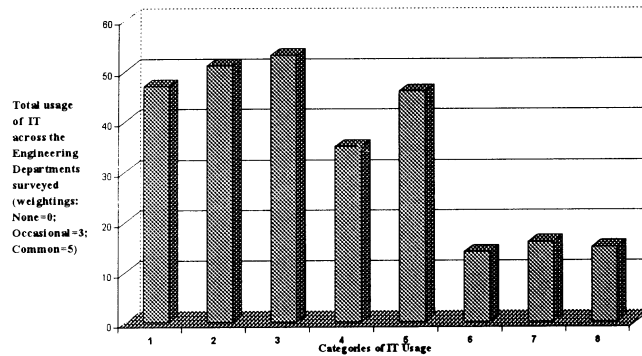


Fig. 1. IT usage for education in engineering faculty.

7 = Computer based learning and virtual lectures  
 8 = Network based virtual learning environments

The survey indicated that the use of IT for educational purposes is quite widespread within the Faculty. However, the degree of penetration of 'advanced' usage of IT, such as network based virtual environments is, not surprisingly, very limited. The remainder of this paper focuses on the development and application of a network based virtual environment for the delivery of a course on (database) software engineering. It describes the objectives of the course/environment, the actual environment developed and assesses the impact of the environment as an educational experience.

#### OBJECTIVES AND CRITERIA FOR ASSESSMENT

The Computer Science department offers a number of courses in database technology as part of the various degree programmes—both full- and part-time—provided by the Faculty. Several of these courses cover similar material with class sizes varying from 35 to well over 100 students. Therefore in order to reduce the time pressure in lectures and the redundancy in preparation/delivery of lectures, to 'improve' interaction with and dissemination of educational material, and provide more in-depth exposure to subject material, a 'self-learning' student-centred course was developed for delivery in a virtual learning environment. Other motivations were to reduce the contention on library resources, automate the process of tracking, monitoring and testing/evaluating student knowledge and to provide a basis for course development across the different degree programmes. The course was designed to be 'self-learning' and focused mainly on the development of practical skills in the application of relational database management systems. It was run over a five-week period and operated in tandem with traditional lectures covering theoretical and other aspects of database technology. The course sought to:

- extend classroom learning paradigm;
- stimulate student 'innovation' and 'discovery'.

Encourage experimentation with the concepts and technology taught within the course;

- increase time flexibility for students taking the course;
- provide persistent access to educational material and provide ease of extension and update of course material while the course is in progress (i.e. 'living' content);
- provide hands-on experience of 'real life' environment;
- support and encourage student usage and feedback;
- provide on-line management of students and course delivery;
- provide an interactive, virtual learning environment which seamlessly integrates pedagogic, administrative and communication paradigms.

These objectives were used as the criteria by which the course and learning environment were later evaluated. The evaluation was conducted with face-to-face interviews with a representative sample of the students taking the course as well as a detailed evaluation using WWW forms.

#### WEB COURSE AND NETWORKED VIRTUAL ENVIRONMENT

The course content comprised text, graphics and animation and was viewable via a JAVA-enabled WWW browser. It was divided into sections, with each section consisting of a small number of modules (a module typically being 1–5 pages). On accessing the course (located via a WWW page), the students authenticated themselves using a unique user ID and password (distributed previously). To view the course, JavaScript was used to launch a separate browser window which had all native browser buttons hidden so as not to distract the user from the main goal of education. A toolbar specially designed for educational use was provided at the bottom of the browser window. From this toolbar the student was able to contact tutors or fellow students, access a relational database, as well as navigate through the educational material.

The modules contained text, graphics and animation to present and illustrate the educational

material, as well as on-line tutorials (form based) which supported automatic submission of completed tutorial answers to tutor(s). Also integrated into the course was an evaluation form, which was automatically submitted and stored for later analysis. Since each student was authenticated, a log of each student's interaction with the educational material was automatically maintained by the system. This log was subsequently analysed in conjunction with tutorial results and evaluation forms as part of the overall evaluation of the effectiveness of the courseware.

The course comprised four different types of information:

1. Administrative (i.e. how to use the course etc.).
2. A database of (self-contained) modules.
3. Indexes or roadmaps of specific routes through various modules.
4. Project assignment specifications and evaluation forms.

The roadmaps were important as the modules could be combined in several ways to satisfy the different requirements for different degree programmes (e.g. business students, computer science students etc.). Each roadmap corresponds to a different 'focus' for the on-line course. Thus the roadmaps provide a means of re-using existing modules with minimal redundancy.

A significant feature of the system was to provide direct access to a commercial relational database management system (RDBMS) via the same interface as the educational course. Thus the toolbar offered an icon which allowed students to issue queries in SQL on live databases. The idea was to deliberately blur the distinction between the educational environment and the 'target' systems. This encouraged students to 'try out' various parts of the course before attempting a larger project. The course provided several project specifications,

which had to be chosen and implemented by the students. The implementation of the projects could be done via the educational interface, providing better pedagogical support for the project implementation.

Figure 2 illustrates the environment showing an example page of the course content as well as the educational toolbar at the bottom of the screen.

## ASSESSMENT OF WEB COURSE AND VIRTUAL ENVIRONMENT

This section discusses the evaluation responses of the students taking the course and discusses the evaluation of the environment and the courseware against its stated objectives. These objectives and evaluations are summarised below.

- *Extend Classroom Learning Paradigm.* The virtual environment and WWW courseware represented a considerable improvement on the original method of teaching which consisted of several tutorials (paper based) and a couple of unsupervised laboratory sessions. It generated much greater usage, understanding and questioning of the course material as the students knew their work was being automatically monitored and assistance was 'virtually' close at hand.
- *Stimulate student innovation discovery and experimentation.* This was achieved (to a degree) via on-line tutorials (with immediate feedback), access to external resources e.g. access to realistic case studies. The blurring of the division between the real-world systems and information and the educational material was particularly important in this regard.
- *Provide greater time flexibility.* Most students appreciated this, especially as the term during

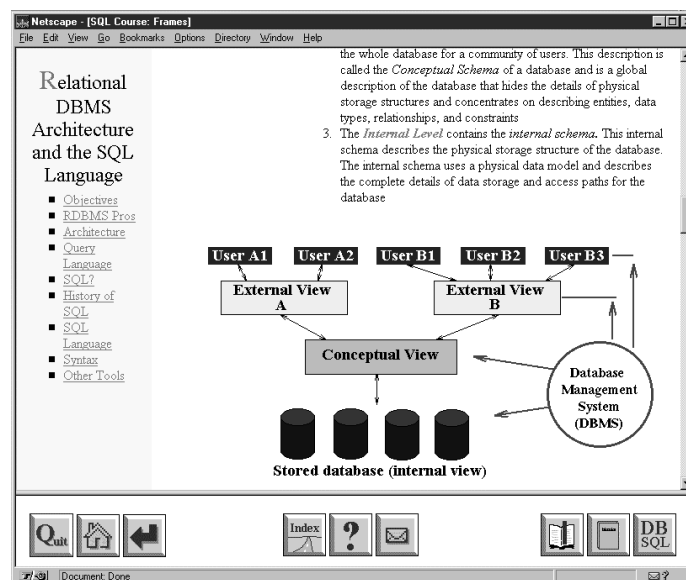


Fig. 2. Virtual student desktop.

which the course was taken was heavily time-tabled with other lecture courses and project deadlines. A few students did, however, indicate a preference for having their day scheduled for them!

- *'Living' course content.* This was surprisingly useful as new information became available during the course execution. Also any 'errors' in the material could be corrected and advertised immediately (unlike the conventional situation which relied on printed handouts). Since the module has been run over a number of consecutive years, incremental extension of the course content has been performed. The effort in achieving this gradual extension has been eased by using standard display templates, icons and feature buttons. The educational course was available (and was used!) 24 hours a day, and was still available in the run up to end of year examinations to assist students in preparation and revision for examinations.
- *Provide hands on experience of 'real-life' environment.* This was achieved by providing access to external systems from within the virtual environment. However, in order to ease the 'jump' from an educational environment interface to a commercial system interface (in this case the commercial system was the Oracle Relational Database Management System [19]), the user was provided with an interactive case study. This case study consisted of a description of an application as well as its implementation in Oracle, together with details of how this implementation was performed. The student could first interact with the case study using a simplified (form-based) interface. Then, when the student wished, full access to the Oracle interface was provided.
- *Support and encourage student usage and feedback.* A major problem with IT-based educational courses is sustaining student motivation. The student can very easily become bored, feel isolated or become disenchanted with the educational experience. It is not enough to rely on self-motivation of the student. In traditional lecture settings, the lecturer may perceive a class's disengagement or disenchantment and may seek to re-attract attention by changing presentation style or may slightly digress to regain attention. Within an IT-based educational system, this degree of flexibility is difficult to encode. Explicit feedback as to student progress was found to be vital in order to sustain student motivation. This encouragement and feedback was implemented using a selection of tutorial style puzzles and on-line comments/feedback. The rule established in the course was that a student's e-mail to a tutor would, at worst, be responded to within 24 hours. Tutorial questions were either automatically corrected and returned to the student in real time, or were stored for tutor correction within two days. The use of tutor-corrected tutorial

questions was not frequent and was used only where the answer necessitated free text response. The frequent use of 'puzzles' and questions was found to help the student establish confidence in his/her comprehension of the material.

- *On-line management of course delivery and students' work.* On-line subscription to the course and account/usage monitoring of the environment and course material were the main administrative systems implemented and greatly reduced the time overhead in concurrently delivering the course to in excess of 100 students. Unfortunately, these systems were not integrated with the college-wide administration systems, although this is technically feasible. Such integration would clearly be desirable, as it would permit automatic registration and enrolment of students.
- *Integrate educational and communication paradigms into a single virtual environment.* By keeping the same 'look and feel' to the educational environment, whether sending notices to tutors or colleagues, communicating with external systems, or interacting with course material, the student is not burdened with multiple program interfaces or interaction paradigms. This was reflected in the student responses and questionnaires which complimented the 'easy to use'/'no manual needed' approach to using the virtual environment. Integration with the on-line monitoring system allowed the tracking of student usage and their correlation of these patterns and the responses to evaluation forms.

## CONCLUSIONS

The evaluation of the course indicated that the students had attained a high competence and confidence in the use of database technology. This was demonstrated in their ability to complete, competently, a medium-sized project using the virtual environment, with little further background reading other than a *Oracle User Manual* (available on-line). Also students demonstrated their comprehension in both written and oral examinations. The satisfaction rating of the course was higher than ratings gathered from the traditional lecture-based approach used in earlier years.

The main problems reported by students using the virtual environment tended to arise from:

- poor quality of terminals (poor screen resolution on older terminals);
- the inappropriateness of an open access terminal room for taking the course (due to noise, poor air quality, lack of ventilation);
- difficulty in sustaining concentration in a room with no supervision and many distractions.

However, students who took the course from on-campus college accommodation or from home, indicated no such problems.

A second problem encountered in developing the

course relates to the use of JavaScript. When the development was first started, JavaScript offered an attractive, multi-platform means of rapidly implementing interactive features for the virtual environment and courseware. However Microsoft's Internet Explorer browser now supports J++ and does not properly support JavaScript whereas Netscape supports JavaScript. This caused problems in accessing the virtual environment via Internet Explorer.

Although the virtual environment and courseware proved very successful, it should be noted that the effort in running such a course is not insignificant. Ensuring the World-Wide-Web server and database server are accessible, dealing quickly with student queries and correction of on-line tutorial submissions is quite time consuming (for approximately 100 students). However, as more and more of the tutorial question corrections are automated, the use of the virtual environment and WWW courseware should be scalable to even larger student numbers with minimal increase in overhead.

The environment and course is now being used by a third degree programme, namely evening Computer Science students, and a fourth degree programme (in Information and Communication Technology) also plan to use the virtual environment and courseware next year. This would bring the total number of students using the course to approximately 250 in any one academic year.

Overall, the effort and time expended in the development of the WWW learning environment and courseware was approximately 2.5 man years and was performed by postgraduate students and an academic staff member. This period included three iterations of design, development, trialling and evaluation. This research formed part of (and as funded by) a larger EU research project, called Prospect, which was investigating the operation and management of broadband (IP over ATM-based) information services across Europe. The iterative design approach to the WWW courseware was vital in achieving a stable, successful student interface. Maintenance of the courseware has been relatively easy, as only minor extensions to the educational resources have been required. Most effort in maintenance is reduced to ensuring student logins for each new student class and minor updates to support (interactive) features.

#### **FUTURE TRENDS AND ISSUES IN IT-BASED LEARNING ENVIRONMENTS**

As has been shown in this paper, there can be significant benefit in using WWW-based educational systems. However, many challenges still

remain. Firstly, there is an overarching need for standardisation in the area of learning technology. The IEEE Learning Technology Standards Committee (LTCS) (P1484) [20] is a formal forum currently attempting to develop standards in several learning technology areas including Architecture and Reference Models, Learner Models, Learning Objects and Metadata. Industrial initiatives are also looking to provide input to this standardisation. One of the most significant recent developments is the Instructional Management Systems (IMS) project which is a US initiative incorporating 600 educational institutions as well as many major industrial partners from across the USA [21]. Much of the work produced in this group, especially in the area of metadata definition of educational resources is being fed into the IEEE standardisation process.

One of the benefits of utilising metadata for learning resources is the enhanced searching and re-purposing that is possible. Currently, the majority of WWW learning content and courseware resides on flat files. Future educational systems will concentrate on the storage, management and re-purposing of finer grains of educational material stored within databases. Retrieval and composition of educational units can therefore be constructed at the time of student access. This allows the customisation of a unit's content based on specific student profile information. A major problem in any on-line or WWW-based course is the need to provide the appropriate level of support and presentation of the content that suits the learner. Such individual customisation of the course content is made possible using such metadata-driven techniques. Early experiments in Trinity College have indicated that by combining fine grained storage of WWW-based content, pedagogical rules for the construction of educational material (based on predefined syllabi) and learner models, customised educational courses can be successfully achieved [21]. Research work is ongoing in the development and standardisation of this metadata and new algorithms to combine this metadata and achieve customised courseware are being developed.

*Acknowledgements*—The authors wish to thank the students on the various degree programmes in Trinity College who so willingly and enthusiastically acted as 'guinea pigs' in the development of the course. Their feedback over the years has played a major role in improving and upgrading the courseware. This research has been supported by the European Commission as part of the PROSPECT Projects funded under the ACTS Programme.

## REFERENCES

1. *ABET Engineering Criteria 2000*, Accreditation Board for Engineering and Technology (ABET), <http://www.abet.ba.md.us/>
2. *Standards and Routes to Registration*, SARTOR, 3rd Edition, Engineering Council (1997).
3. *Changing the Culture: engineering education into the future*, Institution of Engineers of Australia (1996).
4. Reengineering the curriculum for the twenty-first century: a university perspective, *Procs. of Engineering Education—the Future*, Institution of Engineers of Ireland, 12–13 March 1998, Dublin, Ireland.
5. V. Wade, C. Power and R. Riordan, Harnessing WWW/Java technology in teaching a university course, *Proc. 5<sup>th</sup> Annual Conference on Teaching of Computing*, Dublin, August 1997.
6. R. Schank, Learning via multimedia computers, *Communication of the ACM*, May 1993.
7. E. Soloway, Technology in education, *Communication of the ACM*, May 1993.
8. <http://www.utexas.edu/world/lecture/>
9. R. McGreal, Integrated distributed learning environments on the Internet, *Educational Technology Review Journal*, Spring/Summer 1998, No. 9, AACE.
10. <http://www.ilinc.com/index.htm>
11. S. Hiltz, Teaching in a Virtual Classroom, *Proc. ICCAI'95*.
12. K. Tuckey, *Uses of New Technology in Higher Education—Guiding Principles*, Institute for Computer Based Learning, Heriott-Watt University, February 1992.
13. M. Turoff, Designing a virtual classroom, *Proc. ICCAI'95*.
14. <http://wedb.physics.utoronoto.ca/Web/website/appletgallery.html>
15. H. Kaplan, Interactive multimedia and the WWW, *Educom Review*, 32 (1) Jan/Feb 1997.
16. M. Riordan and V. Wade, Tele-educational services in a future open service market, *Proc. ED-MEDIA/ED-TELECOM 1997*, Calgary, June 1997.
17. <http://www.psychologie.uni-trier.de:8000/projects/ELM/elmart.html>
18. S. Trollip, Distance Learning Workshop, Tallaght RTC, 27<sup>th</sup> May 1996.
19. <http://www.oracle.com/>
20. <http://www.edutool.com/ltsa>
21. <http://www.imsproject.org/>
22. B. J. Dillon and V. Wade, Courseware on the Fly, Technical Report, Department of Computer Science, Trinity College, Dublin, 1998.

**Vincent Wade** is a lecturer in the Computer Science Department in Trinity College Dublin and has been researching the area of network based services for over 12 years. He received his BSc from University College Dublin, Ireland and a MSc from Trinity College Dublin, Ireland. He leads the telecommunication information and management research in Trinity College, which includes the areas of Telecommunication based Education Systems, distributed management systems, adaptive hypermedia systems and networked virtual enterprises. He currently leads several EU and industrial research projects in these areas and is author of over forty technical papers in international conference and research journals.

**Professor Jane Grimson** is Dean of Engineering Faculty in Trinity College Dublin, an associate professor in the Computer Science Department in Trinity College Dublin. She also holds the post of President of 'The Institution of Engineering in Ireland'. Professor Grimson's research interests include distributed databases, healthcare informatics, legacy information system integration and migration, and user interface design. In addition to being author of many technical papers published in international journals and conference proceedings, Professor Grimson is author of the student textbook *Distributed Database Systems* (co-authored with Professor Bell).

**Conor Power** is a postgraduate research student working under the supervision of Mr Vincent Wade in Trinity College Dublin. He received his primary degree from Dublin City University and is currently completing his MSc by research.