McGill

The Design of

Asynchronous Learning Environment

Junren Zhu

Supervised by

Professor Gerald Ratzer

School of Computer Science

McGill University, Montreal

March 10, 2001

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements of the degree of Master of Science

0-612-70535-8

Abstract

Asynchronous Learning Environment(ALE) has the capability of providing learning to people anywhere and at any time for both to secure degree and to engage in continuing education throughout their lifetimes. The advance of communications and information technology will make students choose to purchase and enroll in open market, widely available networked courses regardless of institutional affiliation.

Research results have found that success factors for asynchronous learning include whether students felt part of the online learning group, immediate feedback from instructors, automatic self-test, and indicating student's performance and progress in the course. These findings present basic requirement for the design of ALE. This paper explores all aspects of Asynchronous Learning Environment, including the architecture of ALE and complete database design. The modules of ALE include multimedia presentation, identity verification, intelligent agent, automatic test marking, computer conference, chat & whiteboard, and learning scheduling assistance . The purpose of this research is to make ALE a better way of education than traditional education. A database is designed to fully support these ALE functions.

Guidelines of designing ALE are provided with implementation examples of intelligent agents that providing automatic reminders and learning progress report. Conclusion and further works are discussed at the end of the paper.

The design described in this paper is intended for use by engineering courses. But it can be used by courses of other disciplines without much modification.

Résumé

L'Environnement d'Apprentissage Asynchrone (ALE) a la capacité de pourvoir d'apprendre les gens de n'importe où et n'importe quand, à haute sécurité de s'engager dans une éducation continue pendant leurs temps libres. Le développent de la technologie des communications et de l'information fera les étudiants d'acquérir et de s'engager dans une marche ouverte sauvagement disponible des cours en réseaux sans regarder l'affiliation de l'institution.

Les résultats de recherche ont trouve que les factures de succès pour l'apprentissage asynchrone incluent si les étudiants sentent qu'ils font part d'un group d'apprentissage en ligne, qu'ils ont une réaction rapide de leurs instructeurs, qu'ils peuvent s'auto-examiner automatiquement et qu'ils ont des indications sur leurs performances et leur progression pour le cours.

Ces résultats présentent les conditions fondamentales pour la conception d'un ALE. Cette approche explore tous les aspects sur L'Environnement d'Apprentissage Asynchrone, incluant l'architecture de l'ALE et la conception complète de la basé des données. Les modules de l'ALE incluent la présentation multimédia, la vérification d'identité, l'agent intelligent, le pointage automatique aux épreuves, les conférences en ligne, les discutions et l'assistance de l'apprentissage planifiée. Le but de cette recherche est de faire l'ALE une meilleure modalité d'enseignement que l'enseignement traditionnel. Pour appuyer totalement ces fonctions d'ALE on a conçu une base de données.

Les lignes directrices pour la conception d'ALE sont pourvues avec des exemples implémentes des agents intelligents qui ont des mémentos automatiques et des rapports d'apprentissage progressif. A la fin du travail on a discute les conclusions et les prochains travaux à suivre.

La conception décrite dans ce travail a l'intention d'être utilisée pour les courses d'ingénieur, mais elle peut être utilisée pour des courses des autres disciplines sans faire des modifications majeures.

Content

Chapter 1 Introduction

Rapidly evolving information technologies are dramatically changing the way we collect, manipulate, and transmit knowledge. They have increased vastly our capacity to know and to do things. They allow us to exchange information, to communicate, and to collaborate free from the constraints of space and time. Needless to say, the implications of this technology for knowledge-intensive organizations such as universities are profound indeed. The university might be better regarded as a "knowledge server," providing knowledge services (i.e., creating, preserving, transmitting, or applying knowledge) in whatever form needed by contemporary society.

Providing learning to off-campus students was first observed in correspondence courses and in the use of TV broadcasting. But neither of these methods allows adequate and rapid interactions among learners and instructors. Technology-supported methods of supporting group discussions coupled with the widespread availability of personal computers and the Internet has made it possible to provide high-quality interactive distributed group learning experiences.

1.1 The Definition of Asynchronous Learning Environment (ALE)

ALE is an environment for anytime-anywhere learning. ALE combines self-study with substantial, rapid, asynchronous interactivity with instructors and other students.

Learners who need to study part time in order to advance their skills or to secure degrees may well be the group that is most impacted by ALE.

Based on the Internet revolution, ALE provides the means for offering education to students on campus, near campus, and far from campus.

Information technology-enabled communication and dissemination capabilities are likely to impact the economics of engineering education, access to engineering education, the roles of faculty and students, timing and methods of education, the development of student's technical and communication skills, and the ways that traditional engineering students and lifelong learners interact with industry and alumni.

ALE extends learning beyond the boundaries in which graduates will work. The possibility of allying engineering education closely with any industry located anywhere is a vision that can be rapidly realized via ALE, but it is very difficult using traditional learning scenarios. Further more, the use of ALE to improve student's abilities for working together in a distributed networked environment will surely prepare students for a future world in which most work will be accompanied using networked technologies.

It is becoming clear that there is likely to be competition among educational suppliers to provide the best education to remote learners. These learners will have the luxury of choosing the very best education at the lowest cost. Institutions of higher education will have difficulty conforming to these two constraints. Very significant collaboration should result from the freedom of choice that remote learners are likely to have. That is, schools will need to collaborate due to the fierce competition that ALE is likely to generate. While this prognosis may well be unsettling to colleges of engineering accustomed to "business as usual", it is likely that the quality of engineering education will improve in a collaborative and competitive environment.

ALE will provide the possibility for students to take a course from any university based only on the quality and cost of that course. New administration styles will emerge to manage the students at any location from all over the world. The competitions will take place between courses instead of universities. This is a vision resulted from:

a) the global nature of our society;

b) the ease with which information technology—computers, telecommunications, and multimedia—enables the rapid exchange of information; and

c) the degree to which informal collaboration (networking) among individuals and institutions are replacing more formal social structures, such as corporations, universities, and governments.

ALE also presents the issue of "scaling up". There might be a significant demand for some courses because all students have the freedom of taking any course they like. ALE reduces the cost of education in the tasks of grading, student reminding, and online assistance. The automation of test/assignment grading, assistance to common questions, and personal reminding and reporting will significantly reduce the work load of instructors, thus increase the scale of courses.

As one example of this phenomenon, it is clear that many students are already moving rapidly to embrace Net-based learning and are taking increasing control of their own education. Although enrolled in traditional academic programs and participating in time-tested pedagogy such as lecture courses, homework assignments, and laboratory experiments, when unleashed many students approach learning in very different ways when they work on their own. They use the Net to become "open learners," accessing world-wide resources and Net-based communities of utility to their learning objectives.

1.2 Learner–Centered vs. Teacher-Centered

Teacher-centered education is a professor teaching a class of students, who respond by reading assigned texts, writing papers, solving problems or performing experiments, and taking examinations. All students learn the same course at almost the same pace. Learner-centered education is a student learning courses at his own pace. He can read assigned texts, writing papers, performing experiments and even taking examinations at his favorite time.

It could well be that faculty members of the 21st century university will find it necessary to set aside their roles as teachers and instead become designers of learning experiences, processes, and environments. Further, tomorrow's faculty may have to discard the present style of solitary learning experiences in which students tend to learn primarily on their own through reading, writing, and problem solving. Instead, they may be asked to develop collective learning experiences in which students work together and learn together with the faculty member becoming more of a consultant or a coach than a teacher.

Network capabilities will permit creation of engaging interactive on-line courses. Courses can be created for Internet delivery that provide high levels of interactivity and are scaleable to large numbers of students. A key ingredient in on-line engineering education is to provide interactivity and a feeling (for the student) of being individually tutored.

1.3 Distance Education vs. Conventional Education

There are intensive discussions on which is better (distance education vs. conventional education). In my point of view, Asynchronous Learning Environment will combine their advantages rather than compare them and choose one. Actually, most courses in McGill university have their web pages for dissemination of course materials and information such as course syllabus and notes, bulletin board, assignments etc., even they are taught in a traditional way. ALE will provide more functions such as individual performance and progress evaluation and reminders for assignments and personal advancement, in addition to online tutorials and prompt reply of course questions. There will be no doubt that ALE become the dominant education method in the future.

Traditional class meetings accomplish functions such as:

- dissemination to the students of the course materials that they are to learn during the course, with clear signs on which parts are most important or most difficult. If the instructor feels that the students have difficulty understanding some material, he has the option to explain it in another way or by more examples.
- communication to all students in the class answers to questions of common interest, with direct feedback of the extent of acceptance of the answers.
- administration of examinations: all students in a class take the same exam at the same time, with the same amount of time. The students can ask questions about the test if they have difficulty understanding the exam.
- collection of student work: students' work can be collected in the class or deposited at a designated place.
- motivation of the learners, perhaps through interaction with the instructor and other students. The instructor can encourage the students with his/her teaching experience. The students who learn well can help those who have difficulty understanding course materials.

It has been proved[4] that Internet and world wide web can be used to accomplish all of these functions without recreating a classroom environment for the distance learners. For example, both asynchronous and synchronous tools, such as computer conferencing, white board, online chat, can be used to provide motivation to the learners. Distance learners, as well as on-campus learners, seem to view on-site, on-demand, highly interactive learning opportunities as the ideal. And ALE is the only choice for those who want to obtain education but are limited to go to a traditional class.

We believe that traditional teaching and learning experiences in engineering education can be mapped to network-based learning. For example, lecture-based learning can be duplicated in on-line presentations, discussions can be conducted asynchronously, and laboratories can be offered on-line using either simulated laboratories or remotely controlled laboratories.

Lectures and laboratories are the traditional way of delivering engineering education. These two methods are efficient, yet neither can be easily scaled to teach large numbers of students. Large lecture classes found in many institutions are disliked by students and delivery over television has proved to be typically dull. Interaction between the student and instructor is usually minor in either the large lecture or televised class. Laboratory learning is traditionally achieved in hands-on small-group learning experiences.

The current engineering education lecture/laboratory model will likely continue, and economic pressures will probably dictate how quickly network-learning strategies are adopted. If network-enabled learning methods are shown to decrease costs and to increase access, learning, and student satisfaction, then rapid acceptance of the network-based teaching and learning paradigm may be expected.

1.4 Advantages of Asynchronous Learning

Asynchronous learning has the following advantages over conventional education[4]:

- The ability to educate remote learners without regard to time and place.
- University liaison

  ALE will make it possible to eliminate the boundaries of universities. Professors from different universities may cooperate to teach a course. This will leverage the quality of teaching and reduce the work load of each professor. Students will have a broad choices to take courses from different universities. There will be competitions between courses as well as universities.
- University & industry collaboration

With ALE, it will be possible and easier for students to learn from knowledgeable industry elite as well as university professors. It will also be easier for students to learn practical knowledge from industries.

- Individualized learning environment.
  Each student will have his own pace of learning, progress report, his assignment or test schedule.
- The instructor will have better view of the students' learning progress by immediate feedback from students.
- The reducing of the cost of routine educational tasks such as grading, student reminding and online assistance.
- Due to the nature of time shifting and the use of network-based support tools (e.g., computer conferencing), there is qualitative evidence that scale-up is possible. In this case, "scale-up" means being able to increase the student/faculty ratio while maintaining quality education.

ALE makes possible a wide range of capabilities that will make the educational experience of future learners quite different from today's campus-centered experiences. Time on campus for traditional learners may also be reduced and average costs for education may be decreased. In addition, it is likely that new technologies will provide new styles of learning experiences (e.g. using visualization and simulation tools) that will be accessible via the Internet.

1.5 The Impact of ALE on Traditional Education

In the past, most colleges and universities served local or regional populations. While there was competition among institutions for students, faculty, and resources, the extent to which institutions controlled the awarding of degrees and credentials, led to tightly controlled competitive markets.

Today, universities are facing new competitive forces. As the need for advanced education becomes more intense, some institutions are moving far beyond their traditional geographical areas to compete for students and resources. There are hundreds of colleges and universities that increasingly view themselves as competing in a national or even international marketplace. Even within regions such as local communities, colleges and universities that used to enjoy a geographical monopoly now find that other institutions are establishing beachheads through extension services, distance learning, or even branch campuses. Furthermore, with advances in communications, transportation, and global commerce, several universities in the United States and abroad are increasingly viewing themselves as international institutions, competing in a global marketplace.

Many in the academy would undoubtedly view with derision or alarm the depiction of the higher education enterprise as an "industry," operating in a highly competitive, increasingly deregulated global marketplace.

A case in point: The Big Ten universities in the US (actually there are twelve, including the University of Chicago and Penn State University) are already merging many of their activities, such as their libraries and their federal relations activities. They are exploring ways to allow students at one institution to take courses—or even degree programs—from another institution in the alliance in a transparent and convenient way. They are even working together to position themselves to provide educational services on a global scale.

This paper focuses on the structure and database support of ALE. After a literature review in chapter 2, an architecture of ALE is presented in chapter 3. In Chapter 4, I designed the database supporting ALE. The implementation methods and two examples are presented in chapter 5. Chapter 6 gives the conclusion and further work.

Chapter 2 Research Overview

There are many challenges facing ALE. Researchers are working hard to make ALE as well as or better than traditional education. For example, graduates of traditional engineering programs experienced long hours of laboratory experiments , lectures and endless problem solving exercises. Students without access to a campus-based education were hard pressed to study engineering in a similar way. But some researches are undergoing to let ALE students to do engineering experiments on-line. Although not all the experiments can be done on-line, different views about the experiments can be obtained by ALE students and costs of experiments can be reduced. This chapter is an overview of the effort in making ALE benefit the asynchronous learners as well as the on-campus learners.

2.1 ALE Development

In distributed learning environments where there is the potential for losing the cohesiveness and spontaneity of the classroom experience, it is essential to understand how to improve the online learning experience. The instant availability of a human tutor online would be ideal. However, cost and availability are limiting factors in supplying continuously attentive human tutors. Bourne[16] describes a series of experiments designed to evaluate ways by which online courses (i.e., ALE courses) can be improved by the introduction of autonomous intelligent software agents. Intelligent Agents (IAs), termed "knowbots" can perform the duties of online facilitators for routine tasks. Checking computer code, responding to simple questions, reminding learners about the need to turn in assignments and potentially even grading essays [3] are among the types of things that intelligent agents can accomplish.

Raineri [11] developed CyberProf - an interactive, World Wide Web-based teaching system developed at the University of Illinois. Using CyberProf, instructors can create on-line lecture notes that include equations, animations, and graphics; write online interactive homework problems; communicate with students using CyberProf's Web-based conferencing system; survey students in order to receive feedback on course material; and record student grades in CyberProf's on-line gradebook. Students can review lecture notes 24 hours a day; complete homework problems on the Web and receive immediate intelligent feedback to their answers; use the conferencing system to post questions to their instructor, teaching assistants, and peers; and review their grades in the class at any time.

Barker[9] developed CHARLIE, a new asynchronous learning system for engineering technology students at Buffalo State College. It contains many fully-electronic and fully automated features including: individualized homework with graphics, course agenda, course syllabus, tutorials, lecture summaries, access to the Internet. In this paper, CHARLIE is compared to other asynchronous learning systems: Mallard, CyberProf, CAPA, APMS, in their features of user interfaces, functions, cost, client's computer requirements, ease of creation etc.

Mallard [31] is Web based and provides tutorials, practice exercises and quizzes (or homework) for several classes of problems such as multiple-choice, arithmetic, and boolean- expression evaluations. Intelligent grading programs assess the correctness of a response and also determine why an answer is incorrect. Mallard tutorials may contain sound, video, or links to interactive exercises. Mallard requires a UNIX or Linux operating system and a browser such as Netscape 3.0 or later. Security includes four levels of access: student, teaching assistant, developer, and administrator. Mallard strives to make the development of course materials as easy as possible. Mallard's security is based on Netscape's encryption strategy and also on log files which monitor student use.

CyberProf [32] appears similar to Mallard in some ways, but a distinguishing feature may be its research potential in terms of learning how students learn. CyberProf requires a UNIX or Linux operating system, Pearl, C, LaTex, and other utility programs. Student exercises are developed and managed using a suite of utilities: Problem Set Editor, HTML Editor, Grading Software, Survey Manipulating Utilities, Index Maker, WebPlot, and Webdraw. Since Webdraw and Webplot include Java, users are required to use a

Java-compatible browser. CyberProf evaluates symbolic questions that require solutions including units. CyberProf's grading package will evaluate a student's graphical response.

CAPA[33] is a software tool to implement a Computer Assisted Personalized Approach for homework assignments, quizzes, and examinations. CAPA permits randomly distributed variables, exercises requiring graphical solution, and feedback on questions requiring derivations. CAPA permits many types of quantitative and qualitative problems and questions, including: multiple choice with N-correct out of M-choices, fill in the blank, ranking types, and matching responses.

Kortemeyer[19] discussed collaboration of a large number of instructors to provide content of the highest quality . They have constructed Lecture*Onlin*e , a system that provides a homogeneous navigational interface into which web-pages (from any location on the WWW) and individualized homework problems from different sources can be compiled in a way transparent to the student. Ten faculty members at three universities and one community college have been able to achieve the beginnings of the type of collaboration.

Kumar[23] developed a Virtual Learning Environment (VLE), an integrated university environment where students can apply for admission over the internet, enroll in the classes offered by VLE after admission, access a complete course, take tests, and interact with the professors as well as classmates. The VLE has tools for administrators, professors and students for performing their duties. VLE supports the goals of a virtual university. The VLE implements specific modules for routine administrative functions of a university environment. VLE provides for student admission, course registration, grade management, and administrative report generation among other functions. It implements all the functions for a student to take the course over the VLE. The VLE allows students to view lectures and access the study material for every topic of the course. The students are given tests at specified times, and answers to the questions are transmitted to the professor automatically. The communication hub of the VLE has an e-mail facility, chat facility, and a multimedia teleconferencing system. In addition, it has all the tools and instructions for a professor to develop the material for a course to be offered through the VLE. It also includes all the day-to-day procedures that a professor must follow once the course is ready for delivery to the students.

Chandler [29] discussed salient features of a virtual classroom model that can take advantage of a worldwide networking infrastructure of WWW to conduct education electronically. A conceptual overview of this model, called VCOIN, is presented in detail. VCOIN provides a platform for a student-oriented learning approach as opposed to the conventional teacher-centered classroom system. The paper discusses how a student can exploit the flexibility of the VCOIN system to determine his /her plan for learning. However, VCOIN provides several unique features for retaining the instructor's ability to oversee student activities, which is a major advantage in a conventional classroom. Algorithmic details are provided for the following: a) A combination of centralized and FCFS floor control for regulation of audio/video communications of student groups. b) A centralized Mouse Detect feature for monitoring and controlling student activities at a remote place. The practicality of the VCOIN model was successfully validated during pilot runs. Several screen captures are provided to illustrate the performance of the prototype during these pilot runs using multiple computer labs.

Lockledge [8] analyzed ALE from software engineering point of view. He presented user requirement analysis, the Internet Learning Environment Architecture, and a general consideration on the database design including the following factors:

- a student may only be eligible to take an exam once.
- time limit on the online test.
- evaluating a test with multiple choices is simple. But in case of essay answers the system can email these to an instructor to be evaluated.

Some synchronous applications can also be used for users to exchange information. These applications include: Internet Relay Chat(IRC), electronic white board, and audio and video conferencing. They permit better communication among instructors and students when face to face communication is impossible. Microsoft NetMeeting provides all these applications[38]. Video and audio conferencing can let users share ideas, information, and applications using video or audio, or send and receive video images and audio to each other. Electronic whiteboard lets users collaborate in real time with others using graphic information. The users can review, create, and update graphic information, manipulate contents on the whiteboard with the mouse, use different-colored pointers to easily differentiate participants' comments, and save the white board contents for future reference.

Most features for efficient asynchronous learning have been explored by researchers[4, 5,6,9]. But integrating all these features in one ALE and in a platform-independent design still presents great challenges. Moreover, with the development of Internet and practical use of ALE, more useful features will be added to the ALE design.

2.2 Engineering experiments over Internet

Engineering experiments can be provided via Internet. This offers a valuable component to remote engineering instruction that can not be replaced by simulation software packages. The cost of experiments can also be reduced by sharing expensive instruments and students can have better understanding or better views of the experiments they are performing.

Saliah [12] presents a generic interactive virtual and remote engineering laboratory. The laboratory, accessible over heterogeneous computer networks(LAN/WAN), is a collaborative learning environment where teacher and students can interact and perform laboratory experiments. It is possible, from the videoconference and remote command (A1), across the Internet network, the GPIB telecommunication bridge (protocol IEEE 488.2)from site B, to manipulate remotely the articulated robot arm.

Hong Shen [10] designed an interactive on-line laboratory for remote education called Automated Internet Measurement Laboratory(AIM-Lab), which utilizes the Internet and the World Wide Web. The system chosen for AIM-Lab is based on the client/server paradigm. The server, written in Microsoft Visual C++, includes three main components.

The driver interface layer(DIL) manages the tasks and communicates with the instrument driver. It sends the commands to and receives data from the instrument driver, which uses the HPIB IEEE 488.2 standard protocol to drive the instruments. The other two components are a TCP/IP server socket, which communicates with the client side over the Internet, and a GUI interface for the instructor.

Benetazzo[6] argued that, for a preliminary familiarization and experimentation with instrumentation and measurement procedures, the use of virtual equipment is often considered more than sufficient from the didactic point of view, while the hand-on approach with real instrumentation and measurement system still remains necessary to complete and refine the student's practical expertise.

In [7] an internet-based laboratory for an introductory digital signal processing (DSP) class is presented. The laboratory consists of five exercises covering topics like discrete- and continuous-time convolution, Z-transform, pole-zero plots, filter design, and fast Fourier transform (FFT). The software includes web pages which describe the laboratory exercises and supporting Java programs. The Java programs enable visualization of complicated mathematical concepts and allow students to design active simulations of DSP systems. This virtual laboratory has the advantage that students can perform the exercises from remote locations. The paper presents the different programs which were developed for this laboratory and describes several exercises that involve interactive DSP simulations in a user-friendly object-oriented environment. The software has been used at Arizona State University for a senior/graduate level digital signal processing (DSP) course. The paper presents experiences gained by using this innovative software in a class setting and provides information compiled from student evaluations.

2.3 Automatic Testing

Automatic grading of assignments and tests can greatly reduce the work load of instructors. Multiple choice exams can be easily put online and student identities verified by a variety of means. Alternatively, exams can be given in a proctored environment online and automatically graded and entered into a database. There have been recent advances in automating the grading of essays as well as symbolic answers. Automatic grading is more important to asynchronous learning students because they can have self test at any time to check their mastering of course materials.

Tinco [20] developed the QUIZIT system at Virginia Polytechnic Institute & State University to support adaptive as well as standard testing, alone with automatic grading and record keeping, using the WWW as a delivery vehicle. In connection with their Interactive Learning with a Digital Library in Computer Science project, they have applied QUIZIT to help a master's level course as well as a large freshman level service course. There are two run-time modules in QUIZIT: the Grader module and the Monitor module. They are composed of CGI programs(written in C) that interface the WWW pages generated by the compiler and the database engine used to record state information for each student. In addition, these programs are responsible for grading and providing feedback to students after each quiz.

Maron [26] describes a system in which the computer algebra system Maple running on a remote server is used to check the correctness of symbolic answers submitted via the WWW. The goal is to have a computer grade the kinds of symbolic algebraic, trigonometric, and calculus expressions that arise naturally in engineering and scientific applications., rather than just numerical or multiple -choice options. A database keeps extensive, detailed records of students performance.

2.4 Applications of ALE

Although the complete ALE is still under research, some forms of ALE are being used in universities.

Mantey [25] developed the design of a "mirrored-pair" of classrooms, to support distance learning at University of California, Santa Cruz, for the delivery of graduate courses to a site 30 miles from campus, a site now connected to campus by an ATM link. They have also considered requirements for support of remote or "distributed" faculty "electronic office hours", as interactive sessions over a computer network. Another dimension in this digital media in the educational setting is the support for capture and storage of lectures in a digital multimedia library, which offers significant potential value for improved learning and increased faculty productivity.

In distance learning it is more difficult to teach skills than facts. This is particularly so for software analysis and design, which is normally performed as a synchronized group activity. Douglas [17]described a computer-based simulation approach to learning software design skills, from a distance. The approach allows asynchronous practice in object-oriented design using the CRC-card method. The CRC-card method requires groups of designers to role-play the components of an object-oriented design. The approach is intended as a precursor to collaborative design over the Internet.

The Department of Electric and Computer Engineering, University of Florida, has developed an online M.S. degree program that utilizes synchronized streaming video and audio as well as asynchronous learning networks to facilitate interactivity [21].

Latchman [15] presents a model for using information technology to enhance the learning experience for conventional on-campus students, as well as for those students whose circumstance require that they be asynchronous in time or space. The approach emphasizes a solution which allows students to attend the class in real time via the Internet, or to access asynchronously digitally stored video materials with hyperlinks to online training resources at any time. The proposed solution permits interaction in real time and asynchronously among students and between students and instructor, which is a key for effective learning. The hybrid synchronous and asynchronous learning environment in University of Florida will revolutionize the way of learning as well as teaching in the near future. By opening new horizons to working engineers, offering them a chance to improve and enhance their knowledge, they are building one more bridge between the academic community and industry---"virtual universities".

Marion Hagler[18] discussed how to present mathematical equations on the web. With the recent release and widespread support of XML (extensible markup language) and the development of MathML, Web pages not only can display mathematics and equations in TeX-like fashion, but, beyond that, retain the meaning of the equations so that they can be opened and processed by a variety of mathematical software applications. The Web thus can expand the scope of its inherent intense interactivity to include equations and mathematics, as well as text and multimedia.

Starrett [5] presents the experience of five faculty from four universities who are co-developing and team teaching courses using the World Wide Web (WWW). In particular, two novel power engineering courses, Flexible Control of Transmission Systems and Flexible Control of Distribution Systems, introduce the student to the application of power electronic-based controllers, known specifically as Flexible AC Transmission systems (FACTs) controllers, to enhance the performance of the transmission and distribution systems in an electric power system. In order to leverage faculty expertise in the various topics included in each course, a unique partnership was formed among five faculty from the following universities: University of Arkansas, Kansas State University, University of Missouri-Rolla and Purdue University. In order to achieve the most effective presentation, in the presence of geographical, cost, time and technology constraints, the co-developers decided to pursue various distance learning strategies to link the four universities together in a live interactive virtual classroom utilizing data-conferencing software and the WWW. They discussed the use of live audio and video via an integrated service digital network (ISDN) approach, and data-conferencing software in conjunction with telephone conferencing. Each of these alternative methodologies are examined in detail and are compared and contrasted from a cost and pedagogical point of view based on their experiences.

Some of the research work presented above were supported by Alfred P. Sloan Foundation in the programs of Education and Careers in Science and Technology. So it's necessary to understand the goals and efforts of this foundation, as mentioned in the speech of the program director, A. Frank Mayadas, on the 'Learning Outside the Classroom', the Sloan program in asynchronous learning, anytime, anywhere, online[37]:

> The goal here is to make available quality higher education and training anytime and anywhere for anyone who is motivated to seek it. Grants have gone to institutions of higher education to encourage their use of Asynchronous Learning Networks (ALNs), which make possible electronic access at any time to remote learning resources such as instructors, fellow students, text, and software. Sixty institutions, ranging from elite universities to community colleges have directly received grants from the Sloan Foundation, constitute the Sloan ALN Consortium. Through academic year 1999-2000, consortium members will have enrolled well over 100,000 students, and recorded over 3000 faculty years of ALN teaching experience. They are committed, today, to providing over 50 degree programs. All these members are growing at a rapid pace. Where reasonable comparisons are available, we find that learning effectiveness and completion rates in ALNs are about the same as for traditional classrooms.

Chapter 3 The Architecture of ALE

Successful distance education depends on good course material presentation, effective and adequate interaction between students and the instructor, prompt help and feedback from the instructor[4]. The Asynchronous Learning Environment (ALE) should be a network-enabled learning system that are engaging, robust, scaleable and can be easily created and maintained by engineering educators. The design of ALE should fulfill all the needs of its users.

3.1 ALE User Requirement

There are three primary types of users of the ALE: the student, the instructor, and the administrator. Each user has his own requirement which must eventually be met by the system. These requirements are listed below.

Student requirements

- Study at anywhere and any time.
- Course schedule help: Students might need help in deciding his/her schedule for a new course. Some sample schedules should be provided to help the student set up his/her personal course schedule. It's important for the student to follow the course schedule strictly.
- Find out where they are in their course of study. It is related to the student's personal course schedule.
- Get feedback on their performance, by taking self-test or assignments.
- Check their course marks and final grade at any time.
- Correspond with instructors and administrators efficiently.
- Find out what help is available on a specific topic.
- Find relevant material on a topic.

Instructor Requirements

- Use the easiest tool to put course material and test on Web, and put answers to test in the database for automatic marking.
- Correspond with students and administrators.
- Find out how effective their materials are in preparing students for the evaluation.
- Determine students performance and send necessary feedback.
- Find material on a related topic.

Administrative requirements

- Assess what classes should be offered based on student need.
- Assign instructors to classes.
- Correspond with students and instructors.
- Determine how a group of students is performing.
- Create optimal schedule for personnel and resources.

3.2. The ALE Architecture

Based on the requirement analysis of ALE, we present an architecture of ALE in Fig. 1. The modules are explained below.

1. Web-based learning material, including

- Multimedia : Multimedia can be included in on-line courses. Audio recordings can be delivered as downloadable clips or made instantly available using programs such as RealAudio. Likewise, video can be provided using RealVideo. The showing of PowerPoint slides with accompanying audio can be achieved using ActiveX plug-ins for web browsers or by converting PowerPoint slides to HTML pages. Interactive simulations and learning modules can be implemented on the Web using various types of software, including Visual Basic, Java , Power Builder and others.
- Animation
- Lectures on demand
- Industry information (practical technology)

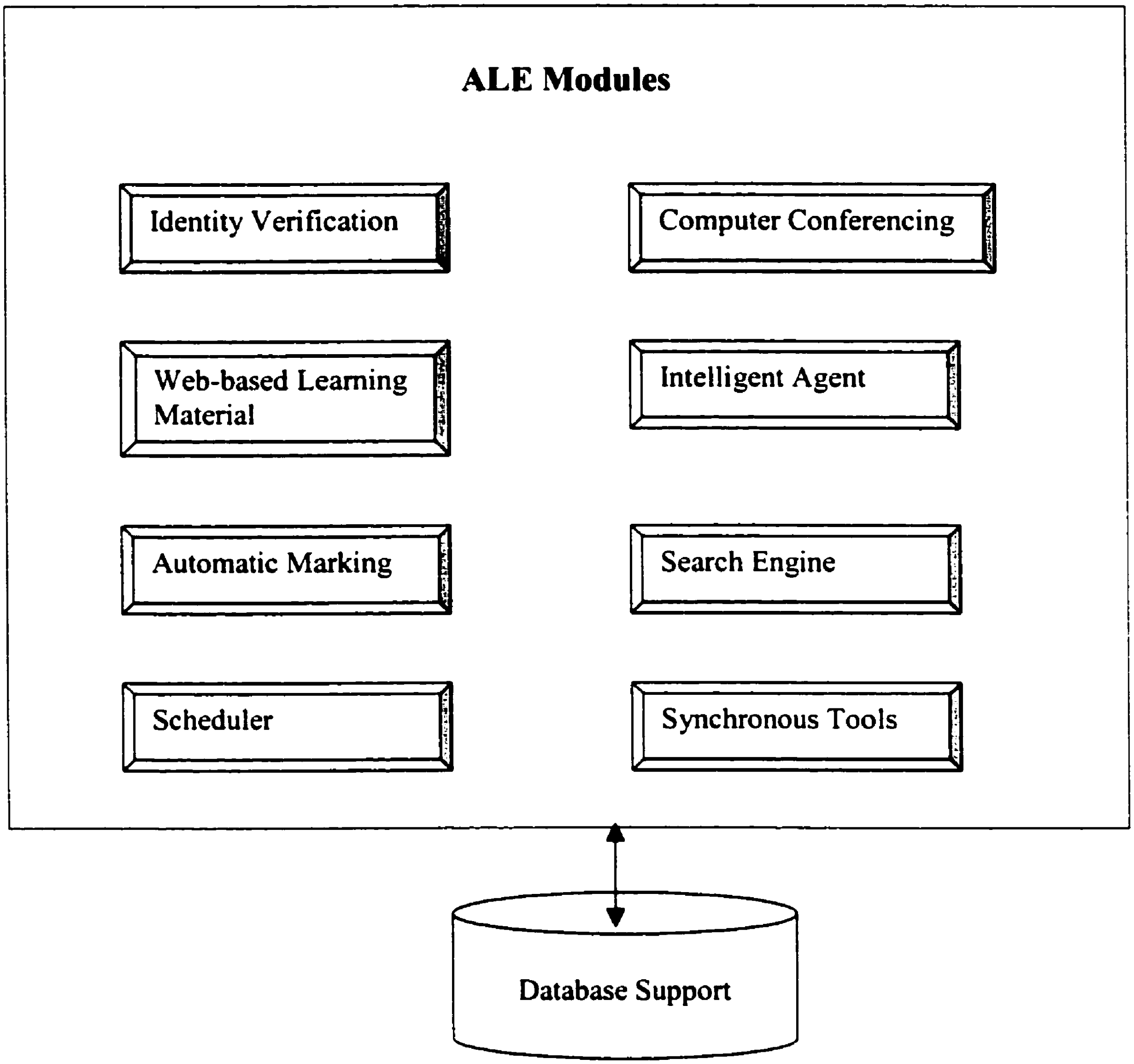

Fig. 3.1. The Architecture of ALE

The instructor doesn't have to use all these formats to present the course material. It depends on the best way to teach the course.

Some applications should be used to handle file formats that are not native to the Web. These applications will include, but will not be limited to, an animation player, a Virtual Reality Modeling Language(VRML) viewer, and an application which allows scripts to be written to control external applications from the browser. The other applications included to enhance the Web are: Email, Usenet News, a chat facility, and an electronic white board. The electronic white board lets users collaborate in real time with others using graphic information. The users can review, create, and update graphic information, manipulate contents on the whiteboard with the mouse, use different-colored pointers to easily differentiate participants' comments, and save the white board contents for future reference. A good example of white board is shown by NetMeeting [38].

3.3 Computer Conferencing

Computer conferencing provides the connectivity between people and among people. The better the use of that connectivity, the better the learning that will take place.

There are many ways to provide online connectivity, including e-mail, listservs, newsgroups, and computer conferencing. E-mail permits one-to-one communication, listservs, one-to-many, newsgroups, many-to-many, and computer conferences, many-to-many. All these communication modalities are used in online courses today.

Newsgroups and computer conferencing have many of the same features, but newsgroups are typically constrained to provide news and normally do not support very robust communication. In contrast, computer conferencing provides additional features that assist in organizing asynchronous conversations.

Several excellent conferencing tools are available. Probably the best known are Lotus Notes [34], First Class [35] and WebNotes [36]. Lotus Notes is a full featured system that is integrated with the Web. FirstClass provides both synchronous (i.e. chat) and asynchronous capabilities.

As an example of the capabilities of computer conferencing, consider Fig.2. On the left side of the screen are items that are typically called forums. One can divide a conference into a collection of multiple forums, each with threads of conversation. Forums might, for example, be used for discussions about each week's assignment in a class and/or to provide help for each week's work in a traditional class. Messages are posted to a thread and replies to messages and replies to replies are easy to generate. Most conferencing systems permit the embedding of multimedia and links to other sites in messages. One of the best capabilities that we have seen is the ability for users to request notification when a message is posted in a particular thread. For example, an instructor might want to be notified by email each time a student posts a message requesting help.

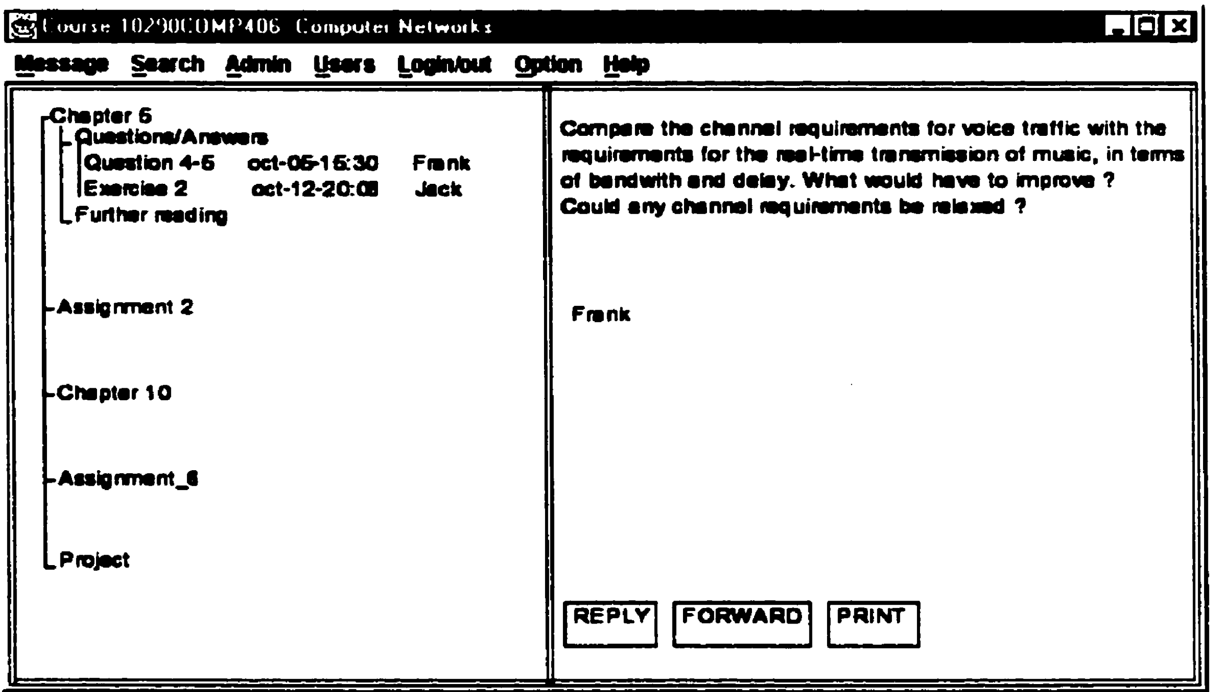

Fig. 3.2. An example of Computer Conferencing

3.4 Automatic Marking

Automated testing system allows students and those outside to evaluate their understanding of course content, receiving instant feedback to aid learning, and yielding a lasting electronic record for instructors and project evaluation groups to interpret. It also provides assistance with record keeping to faculty.

For interactive learning, the long turnaround time in student evaluation and grading has been a continuing concern. Although electronic submission and grading of assignments by e-mail speeds up the process considerably when compared to traditional paper submission, there is also a significant increase in the number of messages between instructors and students which can lead to queues and delays.

Automating part of the process of authoring and submitting exams is an efficient and low-cost alternative to achieve faster feedback in these cases. If used with moderation, and combined with other forms of evaluation, online quizzes can provide yet another means of assessing student performance while significantly reducing instructors' overload.

Most existing computerized test systems are part of computer based training(CBT) and multimedia authoring packages like Authorware, ToolBook, These systems normally come with graphical authoring interfaces, and course database support. Some of them also have networking capabilities, and provide extensive record keeping. Nevertheless, each of these systems has its own different "standards" (i.e., platforms, authoring languages, internal data representation). Further, they are usually very expensive to use on a large scale (e.g., in a university environment), and development costs also are very high. Because they all use different formats, test libraries cannot be easily integrated or accessed, reducing reuse and portability.

With the popularity and growth of the World Wide Web for delivering courseware, several CBT variations, often called web-based training (WBT), have begun to appear. Most of them use HTML extensions as authoring languages, and focus more on presentational than on descriptive syntax. In addition, database and record keeping support often is lacking.

The Authoring Module

Instructors writes quizzes using a customized language. HTML markup is allowed inside textual elements, like question descriptions or general instructions. A special compiler translates test package into HTML pages and secure answer databases which will be used later by the run-time module to instantly grade quiz upon student submission.

The Grader

The grader module is responsible for grading and providing feedback such as score, questions missed etc., to students.

The Monitor Module

At any time, an instructor can browse the database information to monitor any student activity. A complete analysis of this information is extremely important during the assessment process, since many different paths and possibilities may arise because of the test's adaptability and/or networking problems (i.e., time may or may not be so significant, since server connections can be very unstable).

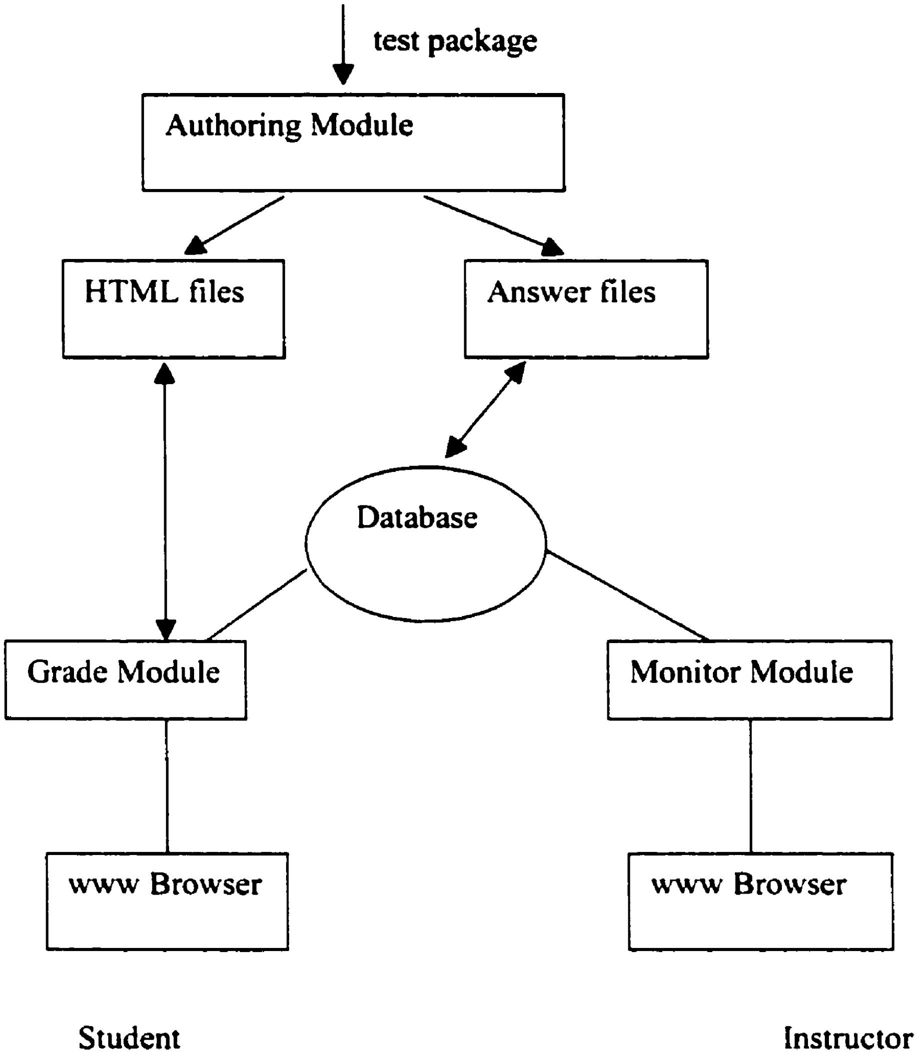

Fig. 3.3 Automatic test system architecture

3.5. Intelligent Agent

An intelligent agent is any program that can be considered by the user to be acting as an assistant or helper.

Checking computer code, responding to simple questions, reminding learners about the need to turn in assignments and potentially even grading essays are among the types of things that intelligent agents can accomplish.

There are many challenges facing ALN. In distributed learning environments where there is the potential for losing the cohesiveness and spontaneity of the classroom experience, it is essential to understand how to improve the online learning experience so that it approaches and perhaps even exceeds more traditional instructional methods. The instant availability of a human tutor online would be ideal. However, providing this capability is no more realistic because cost and availability are limiting factors.

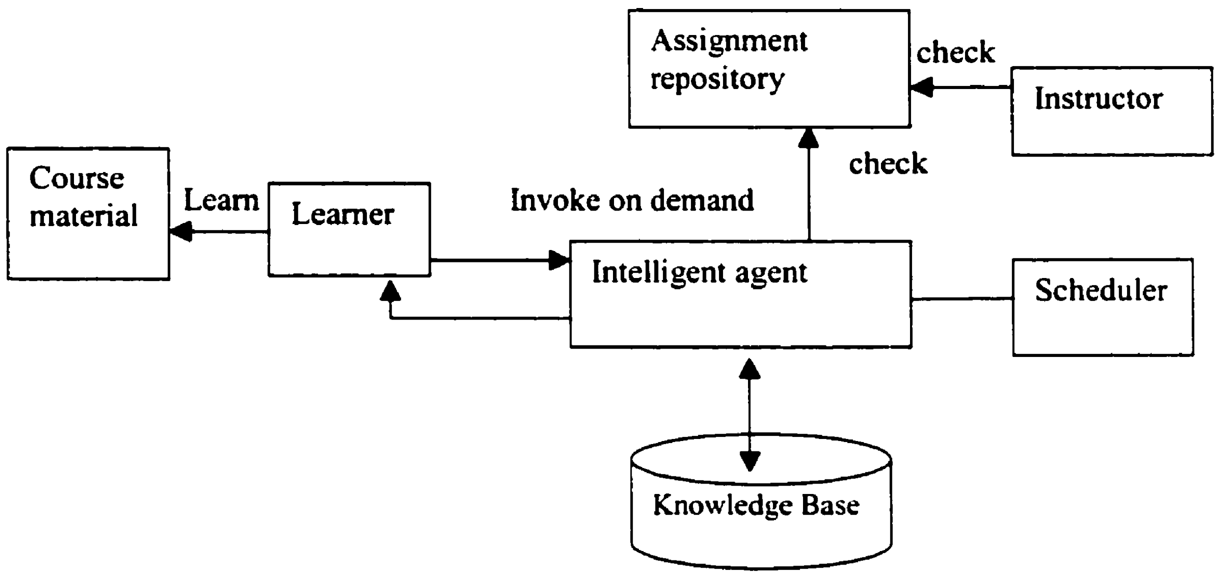

Fig. 3.4. Intelligent agent

Often students simply want questions answered and would be happy with any type of effective immediate feedback – human or machine.

Intelligent agents can be used as a motivational tool. Features of intelligent agents such as encouraging e-mail, immediate feedback, and reminders helped motivate the distant learners to complete the assignments and the workshop. Specifically, the immediate feedback that the on-demand intelligent agents provided after checking the student's assignment helped motivate the learners to stay focused on completing the assignments. Explicit directions on how to fix problems in an assignment were found by learners to be useful feature. Intelligent agents can be a strong motivational tool.

Immediate feedback, including presenting learners with possible solutions, helped the learners to quickly solve their problems. Other than reporting the assignment checking status, immediate feedback also provided other assistance about where to find information in the learning materials and where to seek further help. In these cases, intelligent agents helped learners reduce the time required to find answers to their problems.

3.6. Scheduler

It's important for distant learners to follow a course schedule strictly. The scheduler is a program to help students to determine their individual learning schedule.

At the beginning of the course, the instructor should provide some sample schedules for students to learn the course. The student can use scheduler to choose from one of the sample schedules and modify it slightly, and then save his schedule in the database. His schedule will be used to measure the student's progress and remind him if necessary. An example interface of scheduler program is shown in Fig. 3.5.

Some checks should be performed before letting the student submit his/her schedule. These checks include:

- the start and end date are not allowed to be blank;
- the students should be advised to learn some parts in a designated order for logic learning;
- the learning interval for any part should not be too small or too large;

- overlapping of learning intervals for the parts are not allowed;

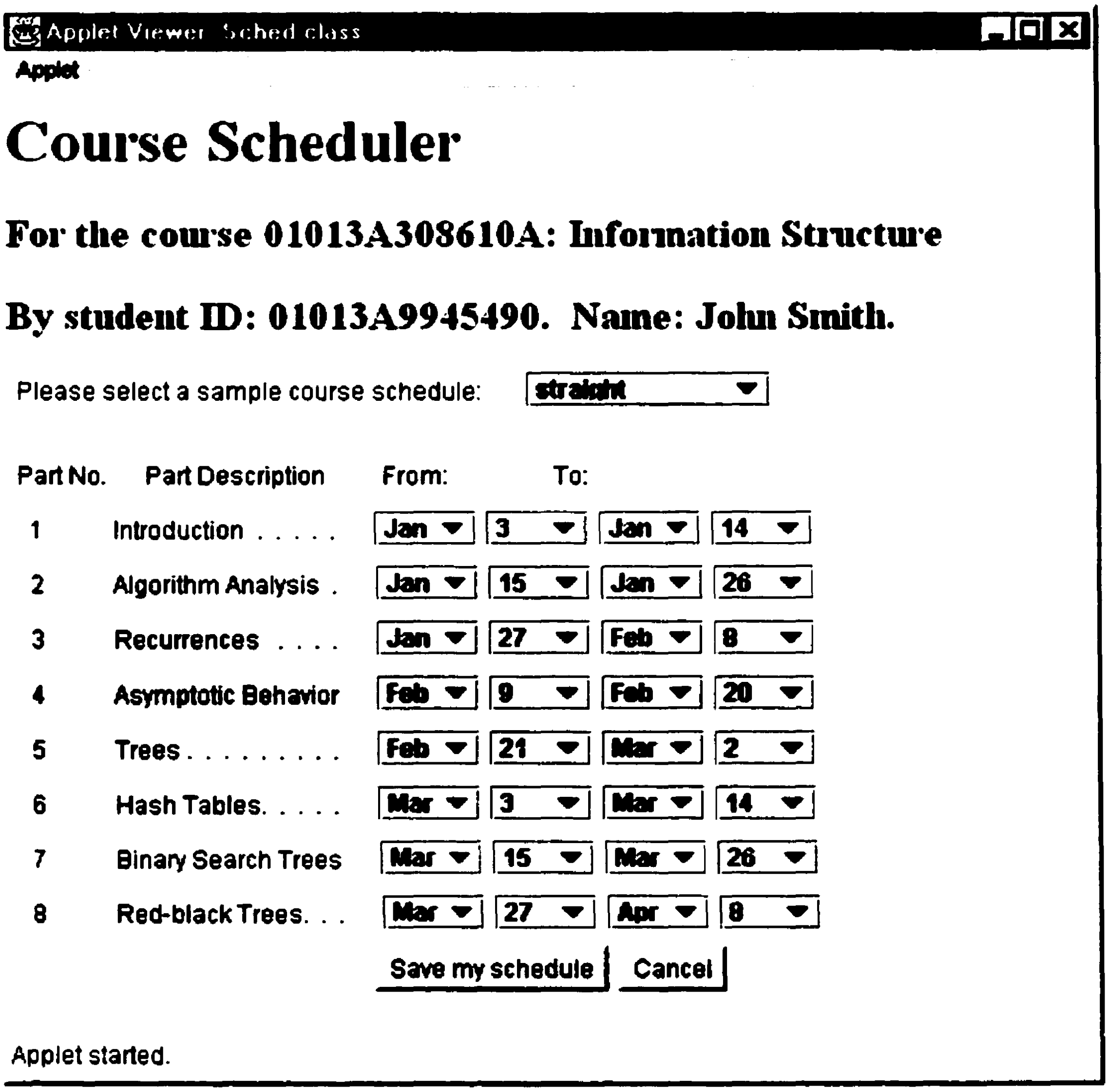

Fig. 3.5 A sample scheduler

Chapter 4 The Design of ALE database

4.1 Scenarios of Database Support

The database should be designed to support all the modules in the ALE architecture(Fig.1). Before we design the database, it's a good idea to have some scenarios to see what kind of activities the database should support. These scenarios are also used to check the correctness of our design of ALE database.

Scenery 1: Student determines his schedule for a course

The instructor should provide some sample schedules for student to learn the course. The student can choose from one of the sample schedules and modify it slightly with the help of a scheduler program, and then save his schedule in the database. His schedule will be used to measure the student's progress and remind him if necessary. It's important for distant learners to follow their course schedules strictly.

Scenery 2: Student takes self test at any time in the middle of a course

The student wants to take a self-test after learned another chapter. He goes to the course web site , finds the right chapter, and clicks on self-test button. The program run as a Java applet, and it picks some questions randomly from the quiz database. After the student finishes the test, the program automatically evaluates the test and gives feedback of performance to the student.

Scenery 3: Student takes a formal test

The student might have to go to a designated lab at the designated time, use the right computer with an designated IP address. The time that the student takes the test is recorded automatically and the student has a time limit for taking the test. After the time limit, the program automatically submits the test and close.

Scenery 4: Student checks his grade for all courses

The student goes to administrative web site, clicks on the "grade report" button, enters his user name (or ID) and password, all his course grades are listed.

Scenery 5: Administration office print a student's transcript

The administrative personnel goes to a secure administrative web site, enters his user name and password, clicks on "transcript" button, enters student's ID and the student's transcript is listed. He clicks on "print" button to print it.

Scenery 6: The instructor check student's performance

The instructor goes to his course Web site, logs in as instructor, clicks on "student performance" button and enters the student ID, the student's progress, all test results are listed.

Scenery 7: The instructor put a formal test

The instructor goes to his course Web site, logs in as instructor, clicks on "Test"->"new Test" buttons. A program will run to let the instructor to enter or paste the questions one by one, and he can also enter the answers to the questions for the system to automatically evaluate the test. Then the instructor sets some configurations such as:

- The test is IP-sensitive or not, that means if the student has to use a designated computer or not. If it is, he can either manually enter the IP address for each student (the instructor might be not preferred this option) or let system randomly designate an available IP address to a student.
- The test will be evaluated automatically or not. If it is not, who is going to evaluate the test.
- The start time (should be an interval) and time limit on the test.

4.2 The Design of ALE Database

The Relational data model is used to design this database to provide full-scale support to learners, instructors and administrators. The requirements for ALE from these three different groups of users are listed in section 3.1. The ER diagram of ALE database is shown in Fig. 5.1. Two aspects of the diagram are worth mentioning: dealing with the many-to-many relationships and normal forms of relations.

dealing with the many-to-many relationships

There are problems with many-to-many relationships. First: the relational model cannot handle many-to-many relationships directly. it is limited to one-to-one and one-to-many relationships. This means that all the many-to-many relationships must be replaced with a collection of one-to-many relationships in order to use a relational DBMS. Second, there is nowhere to store the relationship data in a many-to-many relationship. For example, consider the relationship between the entity 'Student' and 'Course'. A student can take many courses; and a course is taken by many students. But there is nowhere we can store the information about a student's performance in a course ! We therefore must have some entity to represent the relationship between the two.

Entities that exists to represent the relationship between two other entities are known as composite entities. The 'Course_Report', 'Course_Taken' and 'Learn_Schedule' entities are composite entities that resolve many-to-many relationships. These entities also give more flexibility to database query.

Normal forms of relations

With the ER diagram of the database, we'll check the relations against the theoretical rules for good design.

If we place the relations in third normal form(3NF), then we have good reason to believe that we have avoided most of the problems common to bad relational designs. The first three normal forms are:

1NF: The data are stored in a two-dimensional table with no repeating groups. (A repeating group is an attribute that has more than one value in each row.)

2NF: The relation is in first normal form and all non-key attributes are functionally dependent on the entire primary key. (A functional dependency is a one-way relationship between two attributes such that at any given time, for each unique value of attribute A, only one value of attribute B is associated with it through the relation.

3NF: The relation is in second normal form and there are no transitive dependencies. A transitive dependency exists when the following functional dependency pattern exists:

A -> B and B -> C therefore A -> C

After checking according to these normal form definitions, we know that the relations in our design are in the third normal form, thus free from the problems common to bad relational designs.

It's interesting to notice that several new things might appear in the near future.

1. As students are not restricted to learn from one university, and an university might provide education to learners all over the world, new administrative agents or "Asynchronous Learning Centers" (ALCs) might appear to administrate distant learners. They will provide course information of famous universities all over the world and administrate the database.
2. Independent instructors might appear. They are usually good experienced instructor and they are connected to one or more ALC for the resource.

When enter data to the database, the course number should be universal, because students can take courses from different universities. One solution is use university code + course number as a universal course number.

Student number should be universal too, because they might come from different countries. Each course has a scale restriction. But the ALC has to administrate vast scope of students over years. Even the social security number cannot distinguish all the students. One solution is to use a number that is similar to IP address for computers in the world.

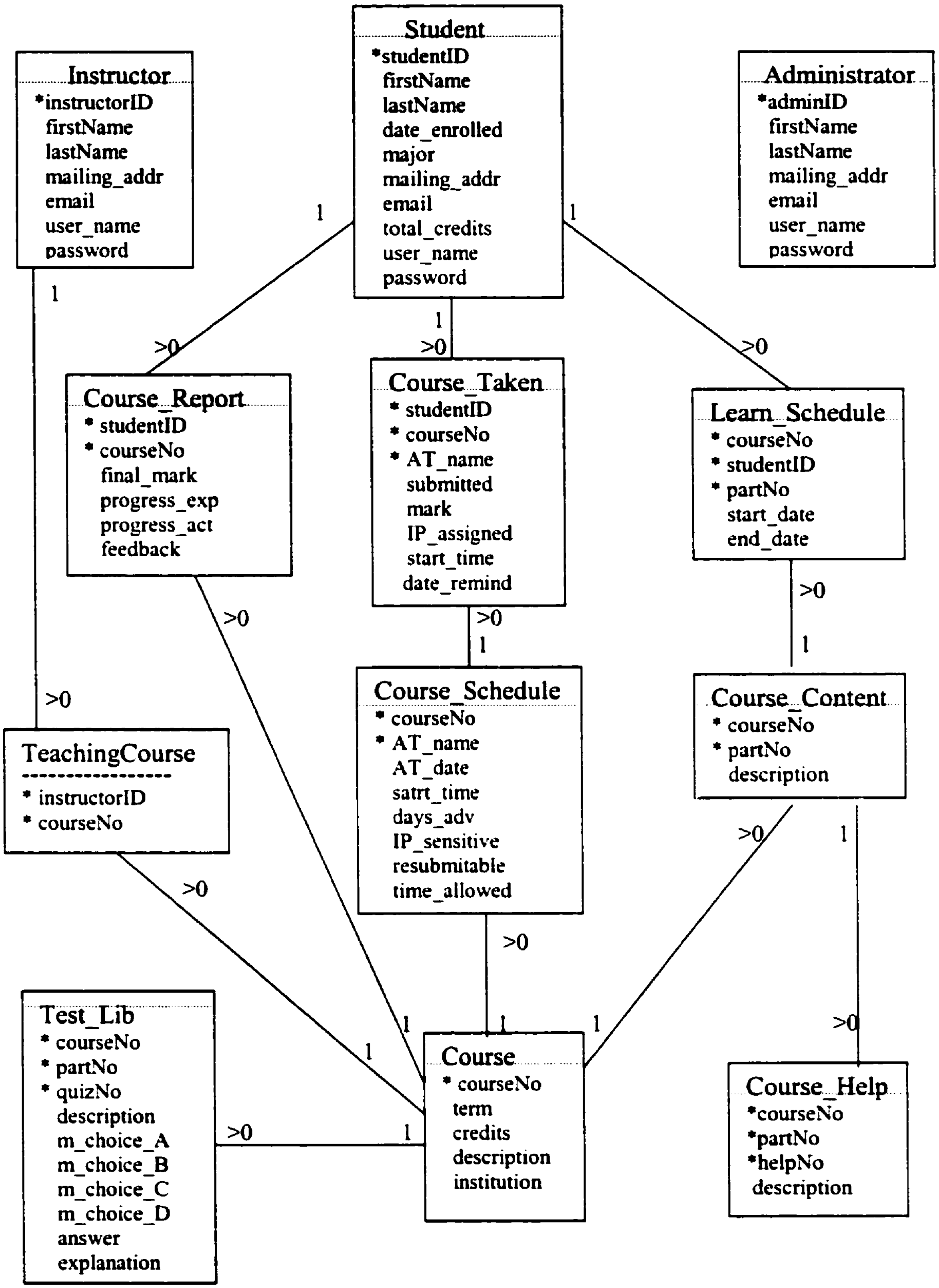

Fig. 4.1. ALE database ER diagram

4.3 The Implementation

An ER diagram in which all many-to-many relationships have been translated into one-to-many relationships through the introduction of composite entities can be translated directly into a set of relations. There are some rules to translate the ER diagram into relations. These rules are:

- Create one table for each entity.
- For each entity that is only at the "one" end of one or more relationships, and not at the "many" end of any relationship, create a single-column primary key, using an arbitrary unique number if no natural primary key is available.
- For each entity that is at the "many" end of one or more relationships, including the primary key of each parent entity (those at the "one" end of the relationships) in the table as foreign keys.
- If an entity at the "many" end of one or more relationships has a natural primary key (for example, an order or invoice number), use that single column as the primary key. Otherwise, concatenate the primary key of its parent or parents with any other column or columns needed for uniqueness to form the table's primary key.

Following these guidelines, we end up with the following tables for the ALE database:

Student (studentID, firstName, lastName, date_enrolled, major, mailing_addr, email, total_credits, user_name, password)

Instructor (instructorID, firstName, lastName, mailing_addr, email, user_name, password)

TeachingCourse (instructorID, courseNo)

Administrator (adminID, firstName, lastName, mailing_addr, email user_name, password)

Course (courseNo, term, credits, description, institution)

Course_Report (studentID, courseNo, final_mark, progress_exp, progress_act, feedback)

Course_Taken (studentID, courseNo, AT_name, submitted, mark, IP_assigned, start_time, date_remind)

Course_Schedule (courseNo, AT_name, AT_date, satrt_time, days_adv, IP_sensitive resubmitable, time_allowed)

Course_Content (courseNo, partNo, description)

Course_Help (courseNo, partNo, helpNo, description)

Learn_Schedule (courseNo, studentID, partNo, start_date, end_date)

Test_Lib (courseNo, partNo, quizNo, description, m_choice_A, m_choice_B, m_choice_C, m_choice_D, answer, explanation)

Chapter 5 ALE Implementation

The considerations in ALE implementation include the client/server model, client and server side functions, database access and programming language . We recommend the three-tier client/server model and that Java be used for both client and server side applications. The reasons are explained in section 5.1 and section 5.2. Two examples are also given to implement two modules in ALE using Java and JDBC.

5.1 Why Java ?

Java is a simple, object-oriented, network-savvy, interpreted, robust, secure, architecture neutral, portable, high-performance, multithreaded, dynamic language.

Java provides facilities for building sophisticated, object-oriented distributed systems using Remote Method Invocation(RMI). Java also supplies the facilities for working with World Wide Web browsers and with legacy systems such as FTP and Telnet. Starting from version 1.2, Java also offers an implementation of the Common Object Request Broker Architecture(CORBA) standard.

Interpreted, portable, and Architecture-neutral

Rather than being compiled into the native code for a given architecture, Java source code is compiled into a compact intermediate code that has no direct correspondence to any given item of hardware. This intermediate code is executed under the auspices of an interpreter known as the Java Virtual Machine(JVM). The JVM is a relatively straightforward program that can be easily ported from one machine architecture to another with minimal change, It is this fact that makes the portability of Java code and a large part of its appeal.

Java's portability also comes from the inclusion of a number of standard library facilities that all Java environments are required to provide. Developers do not have to worry about the existence or quality of a given entity.

Robust and secure

Java was originally intended for the creation of embedded code. Java has no pointers and handles memory allocation and de-allocation "behind the scenes". These two features remove many possible errors. Other language features, such as exception handling, bounded arrays and build-in string type, also make it easy to create well-behaved code.

Security is an area where Java engineers have devoted a lot of time and effort.

- The bytecode verifier ensures that bytecode is structured correctly and is not likely to induce problems at run time. These tests are required because it is assumed that bytecode maybe designed to be malicious or that an uncaught compiler error may manifest itself and thus cause problems.
- A large number of tests are carried out, including: ensuring that no instructions are likely to index beyond the limits of an array, that the run-time stack is always correctly formatted (the number of stack pops are equal to the number of pushes) and that types are always used in appropriate ways (no attempt is being made to use an integer as a reference to an object, for example).
- A number of security policies embedded within the virtual machine are enforced as well. For example, a downloaded applet cannot access the host machine's file systems; it cannot connect to an arbitrary server on the internet, and it cannot execute a native command in the host system.
- The combination of JVM and security policies is commonly called the "sandbox".
- A digital signature can provide a means whereby the author and the integrity of the applet may be established. Any change made to the applet(including attempts to change the signature itself) will result in a different value for the signature. A valid digital signature indicates that the applet's author is trusted, and the code is unaltered from the time of signing.

High-performance

Java is an interpreted language and as such forces a performance penalty when compared with C or C++ that are traditionally compiled prior to execution. Java provides two mechanisms to obviate this problem: JIT and "HotSpot" technology.

A JIT(Just-In-Time) boosts execution speed by converting Java bytecode to native code "on the fly", which can then be directly executed instead of being interpreted as bytecode by the Java Virtual Machine.

Sun's new HotSpot Virtual Machine technology uses a process called "Adaptive Optimization": As an application runs, the JVM detects the frequently used "hot spots", and immediately uses this information to optimize those critical portions of the code.

Multithreaded

The presence of threads as a part of the language definition allows the application designer to more closely model a real-world problem.

5.2 The Client/Server Model

Client/server computing has become the enterprise computing paradigm of the 90s. For nearly three decades, enterprise computing was dominated by the mainframe-with-attached-terminals configuration. Client/server architecture offers following advantages over mainframe computing:

- Distributed processing – The computing load is distributed among the client and several servers. It allows the client easy access to centralized corporate data on several different servers.
- Specialized front/back ends – the server warehousing large amount of data under tight central control, the client visually displaying and processing small extracted data according to individual preferences.
- Centralized administration, security, backups –These expensive but essential functions work best when they are under central control. The client/server model ensures that data is appropriately safeguarded.
- Capacity – Because the client/server model splits the work across the client and the server, a given server can handle more clients, deferring costly upgrades of mainframe hardware.
- Scalability – It's easier to add capacity to a client/server environment than to a mainframe.
- Geographical separation – Client systems ease telecommuting by providing a virtual presence at the central site, as well as independent local processing. Large companies can be spread over the world and yet still have branch offices on line.

In the ALE implementation, the three-tier client/server model is usually adopted, as illustrated in Fig. 5.1. The first tier is typically a large mainframe working as a database server, and the middle tier is usually the application server or web server. The third tier is

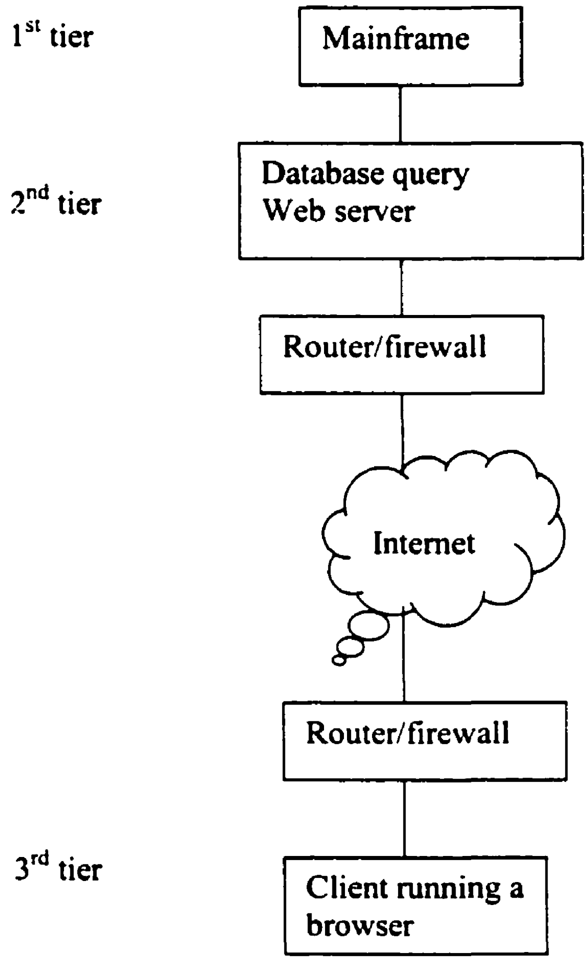

Fig 5.1. Three-tier client/server model

solely a GUI. It simply displays screens to the user, captures the input, and passes it to the middle tier. The middle tier knows how to talk SQL to the database server. It understands and implements the ALE functions.

The three-tier client/server model has the following advantages:

- It frees clients from dependencies on the exact implementation of the corporate database. This allows the database vendor to be changed (e.g. from Sybase to Oracle) without affecting client software at all. The database could in reality be several physical databases, each contributing a portion of the data. A three-tier system allows client/server to be easily deployed, using existing system and infrastructure.

- It allows the "business logic" to be concentrated in one place. If the ALE's functions change, the software is solely restricted to the middle tier. There are many fewer servers in the middle layer than there are clients, so updating software is much cheaper. The middle tier is independent of the client GUIs.
- The system can be implemented more efficiently. If the middle tier can batch teleprocessing requests from many clients to the database, it can greatly improve overall system throughput.
- Scalability. The top and middle tiers need not be a single system, but could be implemented by multiple servers for logistic, security, or performance reasons. The multiple sources will be hidden from the GUI layer.

The Java Servlet Model

The default functionality of a web server is to accept a request for an existing document and return the document to the requesting browser, as illustrated in Fig. 5.2. Server-side include tags allow the server to scan the document before returning it to the requesting browser and replace the appropriate tags with some computed value, such as the current time, the time the document was last modified or the type of browser that is requesting the document. Server-side includes are limited in capability.

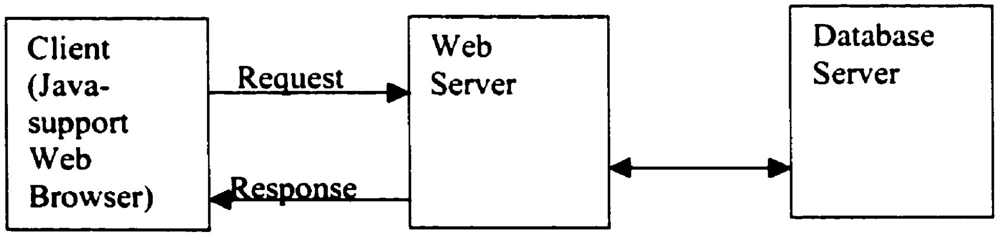

Fig. 5.2 Client and Web Server

The traditional approach to the creation of interactive web pages has been the use of the Common Gateway Interface (CGI). The server executes CGI programs whenever a request is made to access the content of that CGI. There are two disadvantages to this approach;

first, it is very difficult to incorporate the output of a CGI program into a web page. The CGI program itself creates the whole web page, so headers, links, and the rest of the page have to be written into the program. This makes reuse of a given CGI in more than one web site difficult, as the actual CGI program has to be rewritten to match the look and feel of each web site.

The second main disadvantage of CGI programs is that there is no continuity -- every request starts a new invocation of the CGI program, which terminates when the program has successfully returned the information. This is a very big performance hit when the output of the CGI depends on information found in a remote database. Each invocation of the CGI program has to create a new connection to the database, authenticate itself with the database, and make the transaction request.

The Java Servlet model incorporates a good mix from both server-side includes and CGI programs - Servlets incorporate all the favorable features of the Common Gateway Interface but can be embedded in a web page, like server-side includes. Servlets are server-side Java applications that respond to a web page request. The ability of Servlets to incorporate themselves into a web page allows them to be easily reused within the context of many web sites. To facilitate this reuse they take parameters, passed in the form of HTML tags, that specify what each instance of the Servlet is to do.

5.3 Security Issues

Authentication can be accomplished through the Basic Authentication protocol, a WWW standard supported by all browsers. Passwords are checked using Crypt, the same scheme as most Unix systems. For added security, the server may be run over Secure Sockets, which encrypt the passwords while they are being sent from the requesting browser to the server.

Netscape's SSL

SSL is Netscape's Secure Socket Layer – the software encryption behind the Netsite secure server and the Netscape Navigator browser. It is a security protocol that prevents eavesdropping, tampering, or message forgery over the Internet.

SSL is a layer above TCP/IP and below the application layers, so you can transparently add SSL to anything that uses sockets: telnet, ftp, mosaic, httpd. You need it at both the client and server end. It uses RSA data encryption and relies on signed digital certificates from a trusted third party.

If a URL starts with httpd:// instead of http://, it is coming from a secure server. Netscape wants to make it a standard, and they developed an ANSI C library reference implementation called SSLRef, which they submitted to the IETF. If you use Netscape Navigator 3.0 or later as your browser, you automatically have access to SSL when you browse secure sites.

5.4 Implementation of Automatic Reminder

A program is always running on the server to check if any student needs a reminder for his assignment/test/project. The instructor decide how many days in advance a student should be reminded about his/her submission of assignments. The student will receive a reminder email once and only once per day in that period. After the student has submitted his/her assignment, there should not be any reminder email any more.

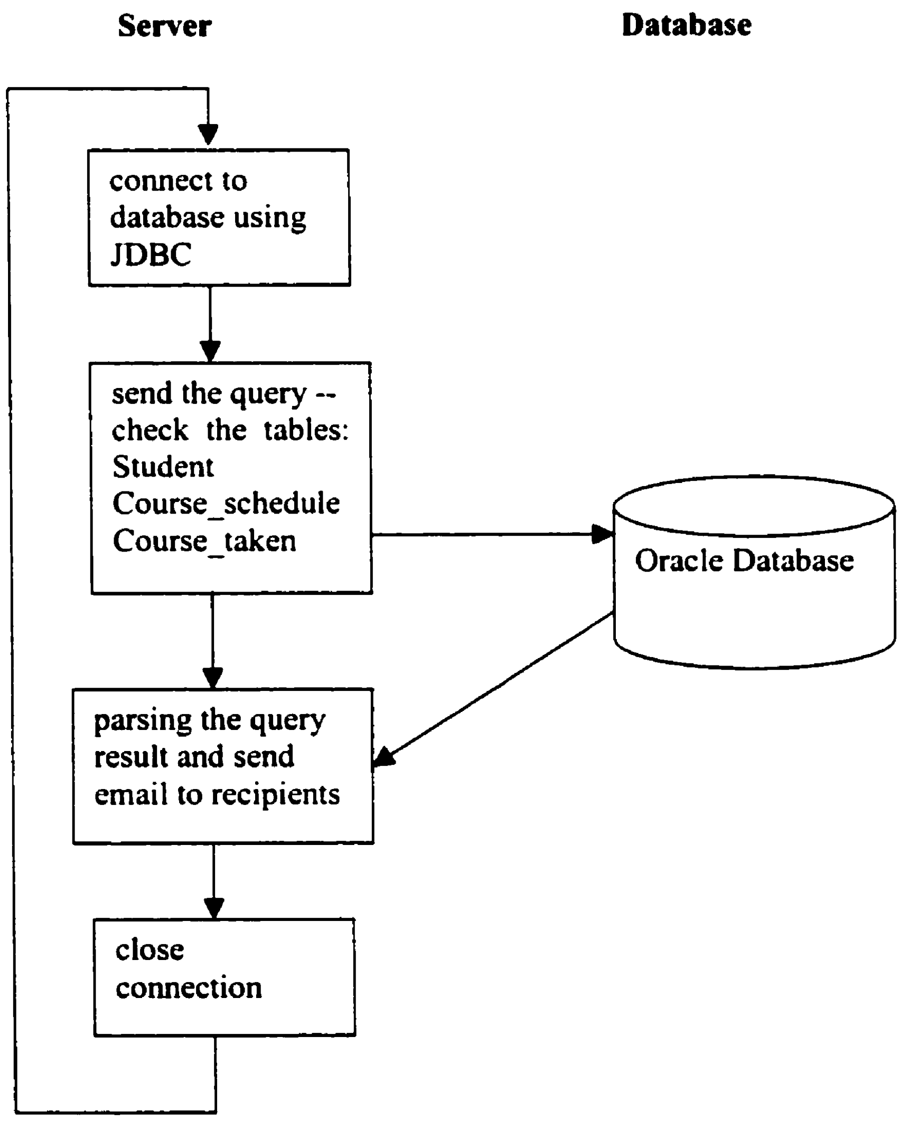

Fig. 5.2 Reminder implementation

This task should be scheduled to perform at the beginning of each day. It is performed exactly once per day unless otherwise specified.

Oracle provides two categories of JDBC drivers:

- JDBC Thin for Java applets and applications
- JDBC OCI for Java applications

JDBC Thin

Oracle's JDBC Thin driver is a Type 4 driver that uses Java sockets to connect directly to Oracle. It provides its own implementation of a TCP/IP version of Oracle's Net8. Because it is written entirely in Java, this driver is platform-independent.

The Thin driver does not require Oracle software on the client side. It connects to any Oracle database of version 8.0.4 and higher. The driver requires a TCP/IP listener on the server side.

JDBC OCI

Oracle's JDBC OCI drivers are Type 2 JDBC drivers. They provide an implementation of the JDBC interfaces that uses the OCI (Oracle Call Interface) to interact with an Oracle database. This driver can access Oracle8.0.4 and higher servers.

Because they use native methods, they are platform-specific. The supported platforms are:

Solaris: version 2.5 and above.

Windows: 95 and NT 3.51 and above.

The JDBC OCI driver requires an Oracle 8.0.4 client installation including Net8 and all other dependent files.

Since the JDBC Thin driver can be used in applets that do not have an Oracle installation, it's impossible to use a TNSNAMES entry to identify the database being connected. The host name, TCP/IP port and Oracle SID of the database being connected must be explicitly listed.

The query which we used to find the list of students being reminded for their submitting of assignments is:

```
select course_taken.studentID, course_taken.courseNo,
course_taken.AT_name, course_schedule.AT_date FROM
course_taken, course_schedule
```

```
where submitted='n' and date_remind !=sysdate and
(course_schedule.AT_date-sysdate)>=0 and
(course_schedule.AT_date-
sysdate)<=course_schedule.days_adv;
```

The parameter `course_schedule.days_adv` will let the instructor to decide when to remind the students.

5.5 Implementation of Progress Report

When a student wants to check his progress in a course, he only needs to enter the part number of the course which he is learning, the server will give him a progress report based on his learning schedule stored in the database.

A Java applet is used in the Web page for student to enter the part number he/she is learning and to display the progress report in both graphical and descriptive formats, as illustrated in Fig.5.3. The advantages of using a Java applet instead of forms and HTML pages are:

- The student can check their progress very quickly. Instead of displaying the progress in different pages, the applet show the result in the same page.
- Different users can check their progress in the same page. Thus it can also be used in a public area.
- It's easy to use Java graphic classes to show student's progress visually.

The information (student ID, part number he/she is learning etc.) are transmitted to the Web server and executed by a Servlet program. The Servlet will use this information and make a connection to the Oracle database, then send the query and get back the expected progress of the student. This result is then sent back to the client site to be displayed at the student's browser.

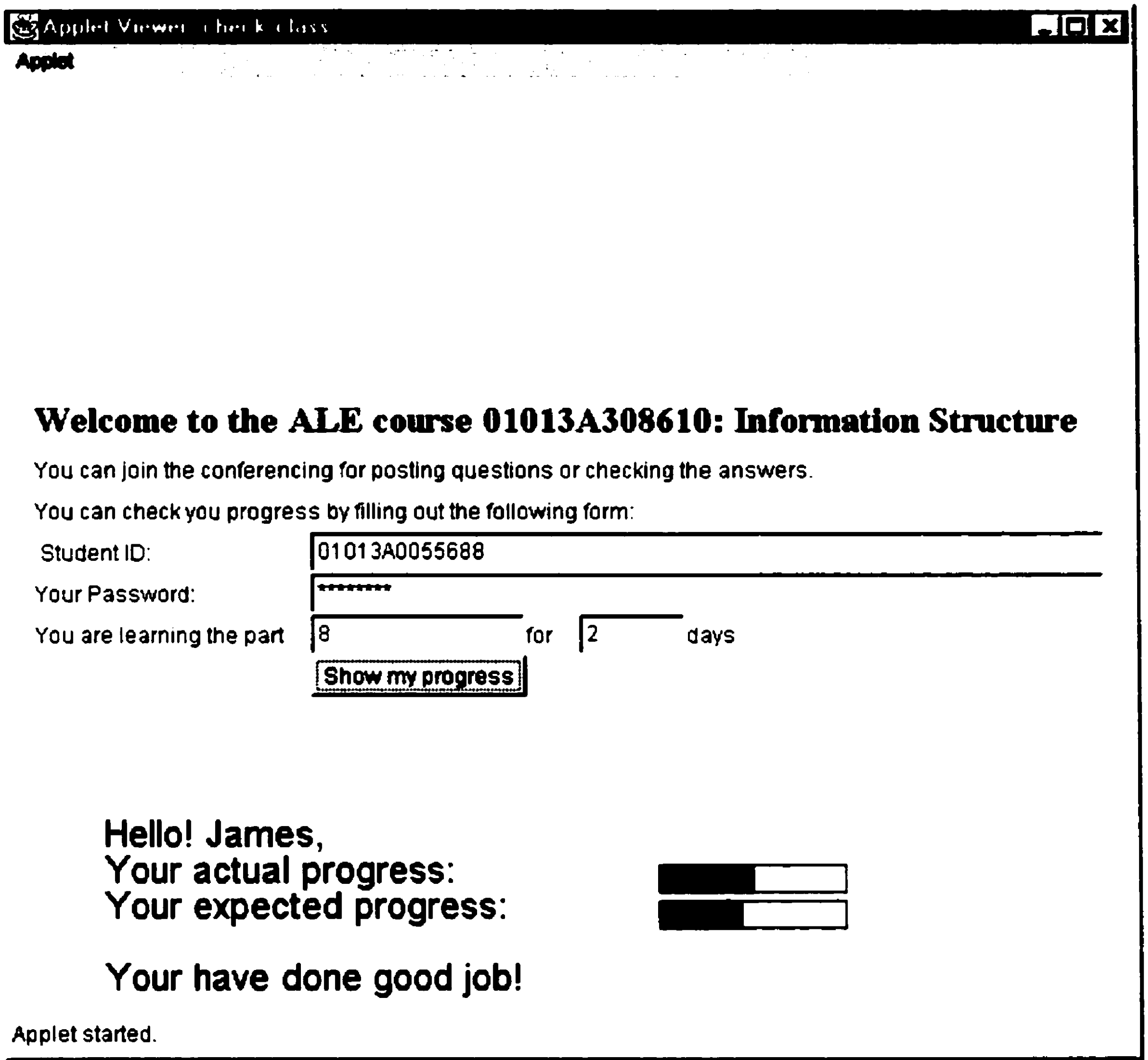

Fig. 5.3 Applet for checking progress

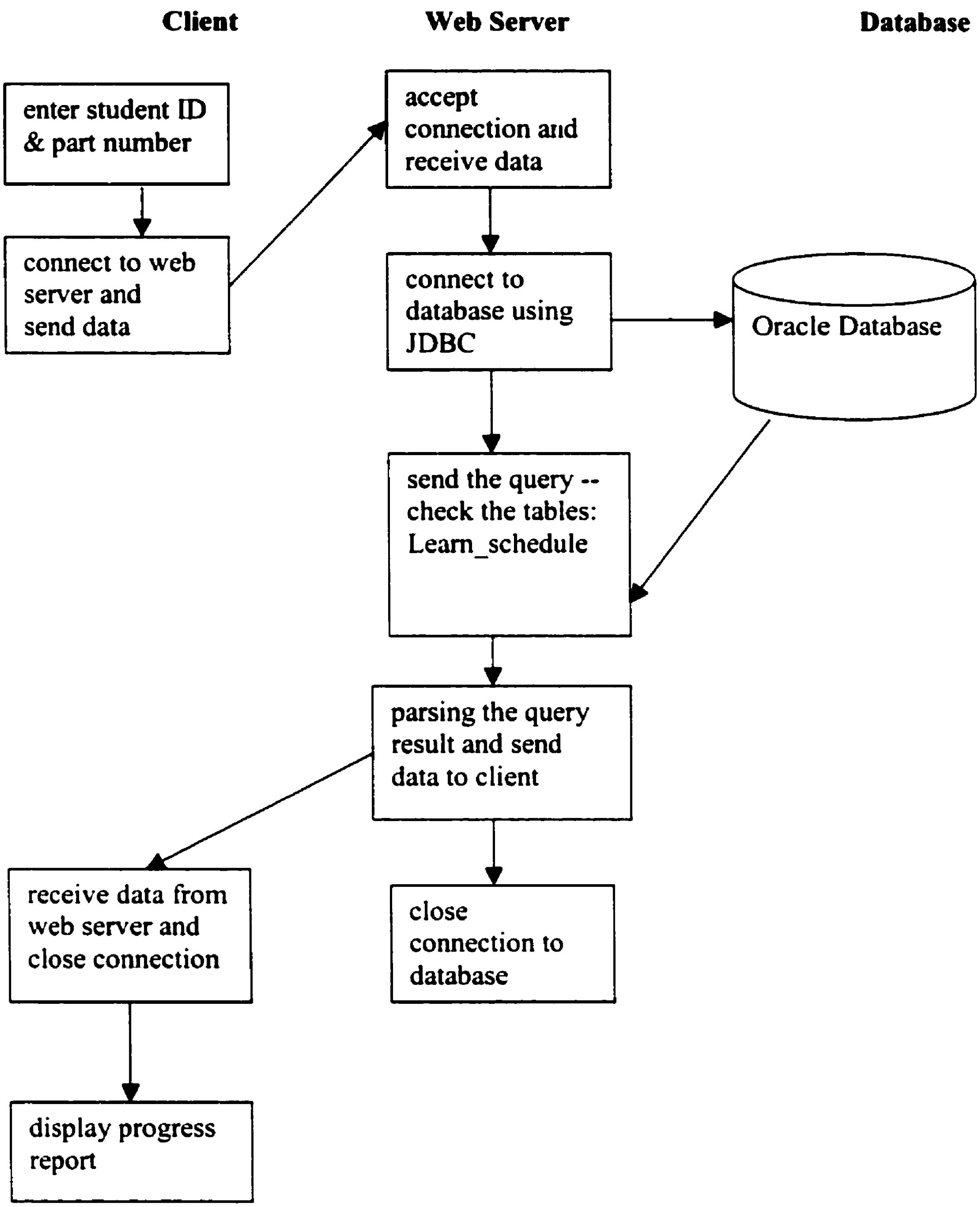

Fig. 5.4 progress report implementation

Chapter 6 Conclusion and Further Work

Asynchronous Learning Environment will assist learners in honing problem – solving skills by providing engaging, learner-centered challenges both for individuals and groups of learners. Both hard skills and soft skills can be improved by online learning. Perhaps most importantly, ALE will help students learn to work in the online milieu of the twenty-first century, in which the individuals of the world will become increasing multinational and workers can contribute to and be employed by a company no matter where they might be in the world.

The structure and design of ALE are discussed in this paper, and some implementation methods and examples are presented. But the some aspects of design of ALE are still under research[4]. Integration of all modules to form a fully functional ALE still needs great efforts. This process can be accelerated by the emergence of independent ALE agencies and their funding on ALE construction. These agencies will cooperate with famous universities and independent professors to serve ALE learners all over the world.

Acknowledgements

The author gratefully acknowledge the supervising of Professor Gerald Ratzer, who spent much effort and time in helping me choose thesis topic, reading my reports and providing good suggestions and helpful feedback.

Reference:

1. A Browser-based System to Support & Deliver DE
   Wendy Doube , 1998 Frontiers in Education Conference, Tempe, Arizona , November 4-7, 1998, pp. 479 – 484.
2. A Framework and Reusable Tools for Developing Interactive Course Web Sites
   Henri Naccache, Timothy Lindquist, Susan Urban, 1998 Frontiers in Education Conference, Tempe, Arizona , November 4-7, 1998.
3. A Model for On-Line Learning Networks in Engineering Education
   J. R. Bourne , A. J. Brodersen, J. O. Campbell, M. M. Dawant, R. G. Shiavi
   Journal of Asynchronous Learning Networks, Vol. 1, Issue 1 - March 1997
4. Asynchronous Learning Networks: An Information-Technology-Based Infrastructure for Engineering Education
   John R. Bourne, A. Frank Mayadas, J. Olin Campbell, Proceedings of IEEE, Vol. 88, NO. 1, January 2000
5. A Virtual Classroom Via Dataconferencing: A Multi-Institutional Experience
   Shelli Starrett Anil Pahwa, etc. 1998 Frontiers in Education Conference, Tempe, Arizona , November 4-7, 1998.
6. A Web-based distributed virtual educational laboratory
   Luigino Benetazzo, Matteo Bertocco, Franco Ferraris, Alessandro Ferrero, Vincenzo Piuri, etc.
   IEEE Transactions on Instrumentation & Measurement, vol.49, no.2, April 2000, pp.349-56. Publisher: IEEE, USA.
7. An Internet-based Computer Laboratory for DSP Courses
   Axel Clausen and Andreas Spanias, 1998 Frontiers in Education Conference, Tempe, Arizona , November 4-7, 1998.
8. An Internet Based Learning Environment
   Jeffey Lockledge, Donald Geister, Lance Mitchell, Phil Go, FIE'96 Proceedings, 1996 ASEE/IEEE Frontiers in Education Conference, Salt Lake City, Utah.
9. CHARLIE: A Computer-Managed Homework, Assignment and Response, Learning and Instruction Environment
   D. Steven Barker, 1997 IEEE Frontiers in Education Conference, pp. 1503, Pittsburgh, Pennsylvania

10. Conducting laboratory Experiments over the Internet
Hong Shen, Zheng Xu, etc. 1999 Frontiers in Education Conference, San Juan, Puerto Rico, 1999.
11. CyberProf™ - An Intelligent Human-Computer Interface for Interactive Instruction on the World Wide Web. Deanna M. Raineri, Bradley G. Mehrtens, Alfred W. Hübler
Journal of Asynchronous Learning Networks, Volume 1, Issue 2 - August 1997
12. Design of a Generic, Interactive, Virtual and Remote Electrical Engineering Laboratory
H. H. Saliah 29th IEEE Frontiers in Education Conference, pp. 12c6-18, November 10-13, 1999, San Juan, Puerto Rico
13. Effective Electronic Marking for On-line Assessment
Mike Joy , ITiCSE '98 Dublin, Ireland, 1998
14. Implementing Novel Instructional Techniques Via the Web
Chaya Gurwitz, 1998 Frontiers in Education Conference, Tempe, Arizona , November 4-7, 1998.
15. Infomation Technology Enhanced Learning in Distance and Conventional Education
H. A. Latchman, Ch. Salzmann, Denis Gillet, Hicham Bouzekri, 29th IEEE Frontiers in Education Conference, November 10-13, 1999, San Juan, Puerto Rico.
16. Intelligent Agents for Online Learning
Choonhapong Thaiupathump, John Bourne, J. Olin Campbell,
Journal of Asynchronous Learning Networks , Volume 3, Issue 2 - November 1999
17. Learning object-oriented software design at a distance.
FIE'99 Frontiers in Education. 29th Annual Frontiers in Education Conference. Designing the Future of Science and Engineering Education. Conference Proceedings (IEEE Cat. No.99CH37011. Stripes Publishing L.L.C. Part vol.2, 1999, pp.12C2/24-7 vol.2. Champaign, IL, USA.
18. Mathematics and Equations on the WWW
Marion Hagler, 1998 Frontiers in Education Conference, Tempe, Arizona , November 4-7, 1998.
19. Multimedia Collaborative Content Creation (mc 3)
- the MSU Lecture*Online* System , *Gerd Kortemeyer and Wolfgang Bauer* , 1998 Frontiers in Education Conference, Tempe, Arizona , November 4-7, 1998.
20. Online Evaluation in WWW-based Courseware
Lucio C. Tinoco, Edward A. Fox, N. Dwight Barnette, SIGCSE'97, 1997

21. Online M.S. Degree in Electrical and Computer Engineering --User Perspectives and Key Implementation Issues Haniph A. Latchman, Proceedings of the IEEE, Vol. 88, No. 5, May 2000

22. THE VIRTUAL LEARNING ENVIRONMENT SYSTEM

*ANUP KUMAR, RAJ PAKALA, R. K. RAGADE, J. P. WONG**, 1998 Frontiers in Education Conference, Tempe, Arizona , November 4-7, 1998.

23. The Design, Development and Evolution of Web-Based Materials Featuring Computer-Animated Simulations

Sam Villareal, Clint Wynn, Dawn Eastwood and Behbood Zoghi, 1998 Frontiers in Education Conference, Tempe, Arizona , November 4-7, 1998.

24. The Future of the University in an Age of Knowledge

James J. Duderstadt, JALN Volume 1, Issue 2 - August 1997

25. Use of ATM and Other Computer Networks in Delivery of Engineering Education – A Perspective Patrick E. Mantey, J. J. Garcia-Luna, Craig Wittenbrink, Arul Ananthanarayanan, FIE'96 Proceedings, 1996 ASEE/IEEE Frontiers in Education Conference, Salt Lake City, Utah.

26. Web-Based Grading of Symbolic Answers

M. J. Maron, P. A. Ralston, T. E. Howe, M. B. Townsley

29th IEEE Frontiers in Education Conference, pp. 12a3-12, November 10-13, 1999, San Juan, Puerto Rico

27. Web-based Interactive Course Homepage Builder

Bijan G. Mobasseri, Xiaoqin Fei, 1998 Frontiers in Education Conference, Tempe, Arizona , November 4-7, 1998.

28. Why the Distinction between On-campus and Distance Learning is Blurring

John R. Chandler, A. Dean Fontenot, Marion O. Hagler, William M. Marcy

29th IEEE Frontiers in Education Conference, pp. 12a2-11, November 10-13, 1999, San Juan, Puerto Rico

29. Yet Another Virtual Classroom over the Internet

Sam Hsu and Abhi Pandya, Rajesh Patel, 1998 Frontiers in Education Conference, Tempe, Arizona , November 4-7, 1998.

30. ToolBook, Asymetrix, Toolbook CBT Edition, http://www.asymetrix.com

31. http://www.cen.uiuc.edu/Mallard/ - Description of Mallard containing examples and design issues.

32. http://cyber.ccsr.uiuc.edu/cyberprof/ - Description of CyberProf which contains a wealth

of information including examples and hardware requirement.

33. http://www.pa.msu.edu/educ/CAPA/ - Description of CAPA which is highly professional, thorough, and inspirational

34. Lotus Notes. http://www.lotus.com

35. First Class. http://www.softarc.com

36. WebNotes. http://www.spyglass.com

37. http://www.sloan.org/programs/edu_careers.htm#careers

38. http://www.microsoft.com/windows/NetMeeting/Features/default.ASP