A Web Based Intelligent Tutoring System(WBITS)

Ahmed S. Salama[1]

Computer and Information Systems Department[1]
Sadat Academy for Management Sciences (SAMS)
Cairo, Egypt

Summary
This paper presents the architecture of a web-based intelligent tutoring system that can be applied to a number of different scientific or literature courses in different domains. The proposed architecture takes the advantages of the web based systems generally and many advantages of building tutoring systems that are based on the web. The proposed web based Intelligent Tutoring System (IITS) integrates a domain knowledge base system, database system, a student model, a course model, an instructor model, and a user interface model. The proposed architecture is a multi tiered architecture and it consists of the following main components: the user interface on the client side through the web browser, the user interface manager on the web server, the student model, course model, instructor model on the application server, database management server and knowledge management system server. The proposed architecture is adopting very important concepts in educational systems including applying adaptive hypermedia in student and instructor interface, evaluating the student capabilities and adapting the teaching strategy to the student's level, using different teaching and evaluation strategies, case based learning, an instructor agent and generating multiple tests and other important concepts making this proposed system a compatible, flexible, adaptable intelligent tutoring system.

Key words:
Intelligent Systems; Multi-Tiered educational Systems; Tutoring Systems; Web Applications; ICAI model; Distant Learning Introduction.

1. Introduction

The tutoring systems have evolved over decades. The first generation of computer assisted education tools were called, Computer-Aided Instruction (CAI) systems. One of the early examples of such a tutoring system is the system by Uhr in the year 1969 [1]. This system generated problems on arithmetic and questions on vocabulary. But main problem of this system were it had no user modeling or adaptation technique. Sleeman and Brown reviewed the state of the art in computer aided instruction and first coined the term Intelligent Tutoring Systems (ITS) to describe these evolving systems and distinguish them from the previous CAI systems [1]. Andes (Conati et al, 2002; Gertner & VanLehn, 2000) is an ITS which was developed to teach physics for the students in Naval Academy. Bayesian networks were primarily used in Andes for decision making [2]. InterMediActor (Kavcic et al, 2003) is an ITS which used fuzzy inference mechanism. It had a data structure called navigation graph. This graph determined which concept comes after which. When there were multiple choices, decisions were made using fuzzy rules [3]. SQL-Tutor (Wang & Mitrovic, 2002; Mitrovic, 2003) is an ITS, which as the name suggests is to teach SQL. Artificial Neural Network (ANN) was used in SQL-Tutor for decision-making. An agent was present who analyzed the student and selected an appropriate problem from the database. That agent
was modeled using an ANN [4]. C++ Tutor is a rule-based ITS (Baffes & Mooney, 1996). Here the concepts of C++ were explained using some rules. These rules were in form of Horn sentences and were called the Theory. The problems were produced to the students in the form of feature vectors [5].
CIRCSIM is a dialogue based system which taught some topics on Physiology to medical students (Evens et al, 2001) Here the problems were presented in the form of dialogues and the interactions with the students were also done in dialogues [6]. An Intelligent Tutoring System that comprises individual learning and collaborative problem-solving modules was proposed by Chitaya & Surasak in 2007. The individual tutoring module was designed to provide appropriate lessons to individual learner. The collaborative problem-based tutoring module was designed to present tutorial problems and provides facilities to assist learners with some useful information and advice for problem solving [7].
Also, O. P. Rishi et al 2007 proposed a Distributed Case Based Reasoning for Intelligent Tutoring System which was an agent based student modeling Paradigm [8]. Leen-Kiat Soh, Todd Blank 2008 introduced an integrating Case-Based Reasoning and Meta-Learning for a Self-Improving Intelligent Tutoring System [9]. Rahim Ocak et al developed a system called "ZOSMAT" that responds almost every need of a real classroom. ZOSMAT can be used for the purpose of either individual learning or real classroom environment with the guidance of a human tutor during a formal education process [10]. The primary focus of this paper is introducing a web based interactive tutoring system that combines the benefits of an intelligent tutoring system with the benefits of the web based learning which enables the distant learning over a multi-tiered distributed system.

Manuscript received August 5, 2011
Manuscript revised August 20, 2011

2. Why Web Based ITS

Developing web based intelligent tutoring systems has many advantages compared with the standalone tutoring systems:

- Global reach of the system by a large number of students.
- If the tutoring system is available on a multi tiered architecture through the web as the proposed system, there will be no need for the users to pay for a copyrighted standalone system.
- Ubiquity: Web based tutoring systems allows students, and instructors to use the system year round, 24 hours a day, at school, home, public libraries, community centers from almost any location that has access to the Internet.
- Having a student model in the proposed system, enables the system to recognize each student's current knowledge level, his progress in the course and many other important student's related information no matters of the location from which the student log in the system.
- Taking advantage of the web browser capabilities on the client side to display animation, cartoons, mpegs, gifs, and other types of multimedia data easily without the need for special purpose programs built especially for enabling the display of such multimedia data in the tutoring system interface.

3. Architecture of WBITS

The proposed Web based ITS which is shown in Figure 1, is a multi-tiered client/server architecture which consists of the client side, the web server, the application server, the Database system(DBS) server, and the knowledge based system (KBS) server.

3.1 The Client Side

The client side would be the entry point for the student and the instructor, through using a web browser that is capable of displaying rich media web pages that form the customized interface for each student and each instructor. Once, the instructor or the student was authenticated, his profile is accessed from the database server, which has all the personalization settings. The entire program that interacts with the student or the instructor can reside in Java applets which are downloaded by visiting the site's URL and that are executed on the student's or the instructor's client machine.

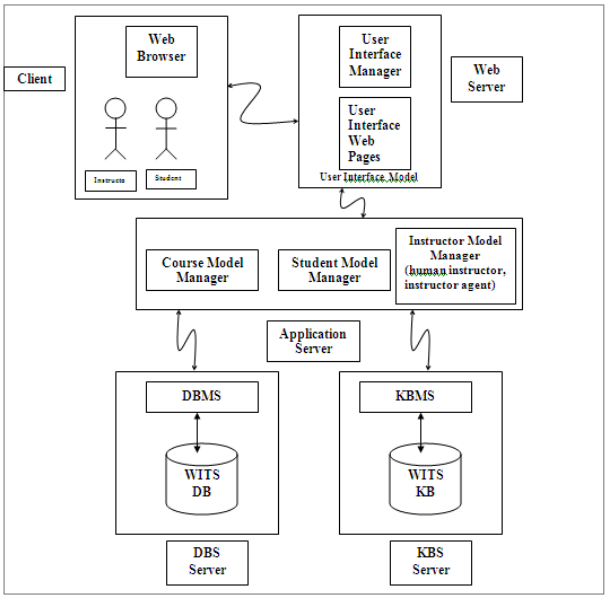

Figure 1 Web Based Intelligent Tutoring System Architecture (WITS)

3.2 The Web Server

The web server will host the User interface Manager and the User Interface Web Pages. The user interface manager consists of two types of modules: the student interface manager module and the instructor interface manager module.

3.2.1 Student Interface Manager Module

This module is responsible for the direct interaction between the system and the student and acts like a mediator between the student model manager on the application server and the student's client. Depending on the current requested use case or operation whether it is an authentication process, presentation of a specific topic material, evaluation using online exam, displaying a progress report, etc., this component will select the most appropriate way for presenting the required web server pages after referring to the student model manager on the application server.

3.2.2 Instructor Interface Manager Module

This module is responsible for the direct interaction between the system and the instructor and acts like a mediator between the instructor model manager on the application server and the instructor's client. Depending on the current requested use case or operation of the instructor whether it is an authentication process, presentation of a specific course topic material using text, audio, graphics, video, and other data formats, selecting an

online exam, preparing or retrieving a student progress report, etc., this component will select the most appropriate way for presenting the required web server pages after referring to the instructor and the course model managers on the application server. Also this module enables opening an online virtual class session room between the instructor and students transferring online video and sketches done on sketchpad, selecting and playback recorded classroom lectures, uploading supplemental educational materials, and short teaching movie clips.

3.2.3 User Interface Web Pages

The user interface web pages include all the students' and instructors' dynamic web pages and forms. They are based on many web based application technologies used in designing server side pages ranging from simple HTML, style sheets, to XML (Extensible Markup Language) and JSP (Java Server Pages). The dynamic web pages and web forms are primarily used for retrieving and/or modifying information from databases.

3.2 The Application Server

The application server will host three important components of the WBITS that represent the educational engines of the system. These three components are: student model manager, instructor model manager, and course model manager.

3.2.1 Student Model Manager

This model can deal with students that have different mental capabilities, educational skills, learning attitudes and personalities. The student model manager module collects student identification personal information, student's visual presentation preferences, diagnostic information about the student's preparation when he first registers through an IQ test to assess his mental capabilities and also his progress in each studied course through its interactions with the student. So a complete report is created on the student profile about the student's strengths and weaknesses with respect to studied courses in the WBITS domain. Another function of the student model manager is to create a student log file, which serves as a source of feedback of the student's usage of the system. This log file can be reviewed by the instructor to be an extra method of student assessment to show his progress and total performance in browsing and tackling different topics, quizzes, exams, and other activities that show his efforts to learn.

The student model manager works in close with the knowledge based system (KBS) server, because it depends on it in different ways. First, once the student registered for the first time, the assessment of the student capabilities through the generated IQ tests is done through the KBS. The classification of the student according to his cognitive style, assessment of his progress level, and other required intelligent functions to manage the e-learning process which are provided by the KBS for the student model manager.

3.2.2 Course Model Manager

The course model manager is an important component in the WBITS. The main functions of this component are as follows:

- Setting courses teaching plans,
- Enabling the instructor to design his different course topics sequence based on AND/OR graphs, using different multimedia representation (video, text, audio, animation, etc.) of each topic.
- Auto selection of the most appropriate course topics sequence for each student separately after consulting the student model manager and the knowledge based system.
- Providing two important modules for exams builders, first module is the auto exam builder module which is capable of generating different exams and quizzes on each topic or on the course as a whole with many styles (essay questions, multiple choice questions, true/false questions, etc.) and second module is the manual exam builder module which enables the instructor to create exams, and quizzes manually. All the created exams and quizzes are then stored in the exams Database on the database system server.
- Using the state space search AI method in creating and solving specific domain problems that requires testing many states and employing different search techniques until reaching the final solution. The module that is responsible for this function is developed as part of the knowledge based system for WBITS and it is problem oriented development. One example of a topic that requires this method in explaining and solving its problems is the transportation problem solving methods such as stepping stone method.
- Correcting tests, analyze the results based on the knowledge rules that are retrieved and fired from the knowledge based system. As a result for the intelligent assessment of the student answers in the tests, a process of updating for various fields in the student profile in the database will be done. Accordingly, the student model will revise the selected teaching plan of the topic or course to that specific student.

3.2.3 Instructor Model Manager

The proposed WBITS instructor model manager supports two modes of operations, first is the human instructor mode and second is the instructor agent mode. So, it contains two main categories of modules:

First category of modules that supports all the functions that can be performed by the human instructor which can be summarized as follows:

- Enabling him to upload to the WBITS database system server the course explanation and teaching materials files whether they were presentation, or video, or audio files.
- Preparing a teaching strategy of a specific course using a wizard that enables him to choose the sequence of topics and deciding level of complexity for each explanation of a topic which can be used in the knowledge based system as a basis for preparing a teaching strategy for each student according to his level of progress.
- Preparing a course schedule, with its milestones and exams dates.
- Following up his students' progress through executing different programmed queries to retrieve performance reports about each student in his virtual class with the assistance of the student model manager.
- Preparing exams and quizzes with different levels of complexity with the assistance of the course model manager and the knowledge based system.
- Opening an online synchronous sessions (such as chat rooms) with a student or many students in his virtual class which enables him to make an interactive teaching session using different interaction tools such as a teaching pad , or a direct video of the instructor himself.

Second category of modules that supports all the functions that can be performed by an instructor agent which is defined as software that can perform and simulate the previous stated functions of a human instructor. The instructor agent is also capable of encoding instructional methods that are appropriate for the target domain and the learner. In addition, based on the KBS and the student model manager which identify each student's skill strengths and weaknesses, expertise levels, and student learning styles, the instructor agent selects the most appropriate instructional intervention.

3.3 Database System Server

This server hosts the WITS Database and the DBMS (Database Management System) which can be Oracle, or SQL Server that manages the WITS DB. The WITS database is an object oriented relational database that saves important data supporting both the student model, course model, and the instructor model. For supporting the student model, database contains data concerning the following for each student: personal student data, the student's educational progress, data showing his level, stopping point in each course, online exams results, his GPA, etc. For supporting the course model, database contains for each course its basic data such as course code , title, required teaching credit hours, quizzes, course topics online explanation files (presentation, video, audio, etc.) , and for each topic its associated lectures, and so on. For supporting the instructor model, database contains data about each instructor id, name, courses he is teaching, virtual classes he is teaching, his students' grading sheets and reports, instructors' online teaching schedule, etc.

3.4 The Knowledge Based System Server

The source of intelligence in the WBITS is the Knowledge Based System (KBS) Server. This server hosts the knowledge base and the knowledge base management system of the WBITS. The knowledge base holds different types of knowledge represented in different representation methods. The knowledge in the knowledge base can be mainly categorized as follows:

1) *Student Model Knowledge:* is knowledge that is represented as production rules. The production rules are used to determine many attributes values concerning the student model covering the student mentality level, appropriate cognitive style for the student whether teaching using the narrative style or graphics or applied examples, the student scoring progress analysis to choose whether to move to the next advanced level in a specific topic or not, etc.

2) *Course (Subject Domain) Model knowledge:* is knowledge that is classified into first: the procedural knowledge which can be used to solve problems in a specific domain. The procedural knowledge can be represented as production rules that apply the state space search with backtracking as an important AI technique for explaining and solving problems that their solutions can be reached in different steps and stages. Second type of knowledge is represented as production rules which can be used to choose the best teaching strategy for a topic at specific stage in the student's progress in a course according to the student model attributes values. Also the production rules will

choose the most appropriate course topics sequence for each student separately. In addition, the production rules are used to generate different levels of problems, tests, quizzes, examples and exams on the fly, combine and apply rules to solve them, and then they will be saved on the DB system server to be retrieved when needed through the student interface manager.

3) *Instructor Model Knowledge:* is knowledge that is also represented as production rules which support the instructor agent to select the most appropriate instructional intervention based on each student's skill strengths and weaknesses, and expertise levels which are determined by the student model manager.

4. Conclusions and Future Work

This paper proposed a multi-tiered web based intelligent tutoring system (WBITS) that takes the advantages of the web based applications in providing the ubiquity, flexibility, scalability, and other advantages of distributed multi-tiered client server architecture. The WBITS consists of the first tiered: the client side which is based on the web browser and the downloadable java applets, second tired: the web server which hosts the user interface model which consists of the user interface manager and the user interface web pages, third tiered: the application server which hosts student model, course model, and instructor model managers. The instructor model manager can work in two modes: supporting the human instructor or acting as an instructor agent that imitates the activities of a human tutor. Fourth tiered is the database system server which hosts the DBMS and WBITS DB, and the fifth tiered is the intelligent component which is the knowledge based system server which hosts all the required knowledge represented in different methods to support the student, course, and instructor model managers in their work. The proposed architecture is adopting very important concepts in educational systems including applying adaptive hypermedia in student and instructor interface, evaluating the student capabilities and adapt the teaching strategy to the student's level, using different teaching and evaluation strategies, case based learning, and generating multiple tests and other important concepts making this proposed system a compatible, flexible, adaptable interactive intelligent tutoring system.

Future work will focus on incorporating additional interaction features for the proposed system's user interface development methodology to support the interaction with different types of disabled users whether they were students or instructors including able-bodied, blind and motor impaired.

References

[1] D. Sleeman, and J. S. Brown, "Introduction: Intelligent Tutoring Systems", Ed. Academic Press, New York, 1982, pp. 1-11.

[2] C. Conati, A. Gertner, , K. VanLehn and M. Druzdzel, "On-line student modeling for coached problem solving using Bayesian networks", *Proc. Sixth International Conference on User Modeling*, Vienna , 1997, pp. 231-242.

[3] A. Kavcic, R. Pedraza-Jimenez, H. Molina-Bulla, F.J. Valverde-Albacete, J. Cid-Sueiro, A. Navia- Vazquez, "Student modeling based on fuzzy inference mechanisms", *EUROCON 2003, Computer as a Tool. The IEEE Region 8*, vol.2, 22-24 Sept. 2003, pp. 379- 383.

[4] T. Wang, and A. Mitrovic, "Using neural networks to predict student's performance", Proc. Of International Conference on Computers in Education, 2002, pp. 969-973.

[5] P. Baffes, and R. Mooney, "Refinement-Based Student Modeling and Automated Bug Library Construction", Journal of Artificial Intelligence in Education, Vol. 7, 1996 , pp. 75-116.

[6] M. W. Evens, S. Brandle, , R. Chang, R. Freedman, M. Glass, Y. H. Lee, "CIRCSIM-Tutor: An intelligent tutoring system using natural language dialogue" , Proc. *Twelfth Midwest AI and Cognitive Science Conference*, Oxford, 2001, pp. 16-23.

[7] T. Chitaya, and M. Surasak, "Design of an Intelligent Tutoring System that Comprises Individual Learning and Collaborative Problem-Solving Modules" , Fourth International Conference on eLearning for Knowledge-Based Society, November, Bangkok, Thailand, 2007, pp. 18-19.

[8] O P Rishi, G. Rekha, and S. Madhavi, "Distributed Case Based Reasoning for Intelligent Tutoring System: An Agent Based Student Modeling Paradigm, Engineering and Technology ", Volume: 23, 2007, pp. 273-276.

[9] S. Leen-Kiat, B. Todd, "Integrating Case-Based Reasoning and Meta-Learning for a Self-Improving Intelligent Tutoring System", International Journal of Artificial Intelligence in Education, **Volume:** 18, 2008, pp. 27-58.

[10] O. Rahim, K. Ali and G. Aslan, "Web-based intelligent tutoring system for teaching–learning process" , Expert Systems with Applications, Volume 36, Issue 2, Part 1, March 2009, pp. 1229-1239.