A Template Based System for Automatic Construction of Online Courseware for Secondary Educational Institutes

Fotis Lazarinis
Department of Applied Informatics in Management and Finance
Technological Educational Institute of Mesolonghi
Mesolonghi 30200, Hellas, Greece
Tel: +30-26310-58212
Fax: +30-26310 25183
lazarinf@teimes.gr

ABSTRACT

Many educational sites are static and rarely updated, diminishing the dynamism of the Web. An online learning system should be adjustable, expandable and regularly updated, preferably by educational staff without the direct intervention of computer experts, reflecting the dynamic character of education. To achieve this aim, we propose a system based on three roles: Administrator, Educator and Learner. Each role has a different set of responsibilities and permissions to the system. The model is built around a repository of educational material, vital for its expandability and its dynamism. The repository contains a set of customizable templates, dynamically filled with the educational material, assembling the online course. This paper discusses current difficulties in developing online courses and then presents the development progress of such a system and its anticipated advantages to Secondary Educational Institutes.

Keywords

Learning Technology Systems, E-Learning Tools, Educational Databases, Dynamic Content Generation, Secondary Education

Introduction

Following the popularity of computers, the nature of education has changed significantly, over the last decade. Many researchers and academics predicted that education would rely heavily on technology (Phillips, 1992; Daniel 1996; Crossman, 1997).

Initially the utilization of computers in education created new terminology, where the compound terms Computer Assisted Instruction (CAI), Computer Based Training (CBT), and Computer Assisted Learning (CAL), were extensively recorded in the literature (Grieve, 1992; McDonough et al., 1994; Serdiukov 2000). Soon after, with the appearance and expansion of the Internet and especially its World Wide Web service (Berners Lee et al., 1992), the above terms evolved to Web Based Training (WBT) or Web Based Learning (WBL) or Online Learning (Brahler et al., 1999).

The real power of WWW arises when the results returned to the client are dynamically produced based on a client's request. In education measurement of this potential is vital, as static information would be of little use. Producing customized and rapidly evolving educational material specifically cut to the needs of the learner or educator is essential.

However, most of the online courses offered, are static and infrequently updated, diminishing the real power of WWW. These locations are often developed with MS-Office tools (Tiedemann, 2002) and are sporadic attempts of teachers, therefore increasing significantly the overall development and maintenance time (Golas, 1993; Brahler et al., 1999).

In this paper we present the results of the first phase of an in progress project, aiming at building a system that enables the creation of online courses by the educators themselves. This system could be easily utilized by high schools or other secondary education institutes where funding is limited and technical staff is not available.

Problems of WBT Systems

Academics or school teachers develop online courseware to enable students to quickly gain electronic access to their course material and courses are re-engineered to be technology based (Meier & Simon, 2000). However, most of these attempts are standalone endeavors and trigger new problems.

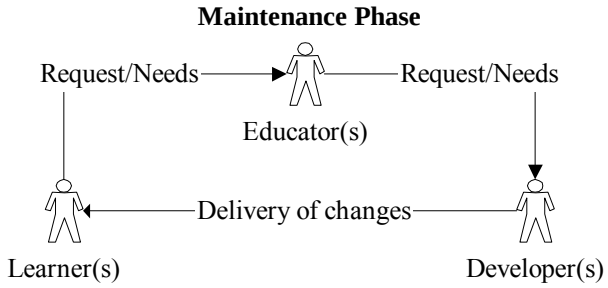

Figure 1: Maintenance phase in WBT development

Development of medium or high complexity online learning sites, as in all information systems, undergoes the phases of analysis and design, implementation, testing and maintenance (Brahler et al., 1999). The individual types of users involved in these phases are the: Educator, Developer, and Learner. These types of users form larger teams which in turn develop the project.

One of the basic problems of this tactic (see figure 1), is that every time an educator or learner needs an alteration (maintenance) of the online learning site, the developer should be contacted. This leads to a series of problems and inconveniences as the maintenance, in reality, initiates a new development cycle. Hence, an online learning project is typically confronted as a new project and starts anew (Kinshuk & Patel, 1997). Obviously, this tactic increases the expenses of development for educational institutes (McCormack & Jones, 1998; Sklar & Pollack, 2000) and frequently is a barrier to further development.

The nature of design of these sites is mostly presentational (Brahler et al., 1999) or the delivery of educational material is typically static, and this is one of the reasons why these systems, according to Carswell and Murpfy (1994), fail to fulfill their aims.

In other studies (Golas, 1993; Brahler et al., 1999), it was presented that for every hour of basic level online courseware 30-200 hours of development is needed. For offering 1 hour of online courses of high complexity, at least 200 hours of development are needed. Even though the situation is dramatically improved with the utilization of graphical tools, such as MS-FrontPage or even MS-Word, the time required for medium to high complexity online courses, where the learner interacts, is still high and requires the systematic involvement of computer experts.

Another problem with the current WBT systems is the massive existence of data in Web pages and their personalized style, associated to their creators (Butler, 1995). Even in the same educational institute the web pages retain a quite dissimilar appearance, without cross-reference linking between topics.

In a recent study Markwell and Brooks (2002) monitored 515 nonredundant URLs from educational sites for 14 months and discovered that during this period, 16.5% of the monitored URLs have ceased to function. Clearly, this is a serious inconvenience to both students and teachers, who cannot rely on online content, as there is no certainty that it will be available next time on demand.

More problems in the integration of online learning can be identified in secondary educational systems. The majority of teachers is moderate computer users and use only MS-Office tools and Internet Browsers and do not use computers in the learning process. That is why massive attempts have been done worldwide to increase ICT to schools (Dagiene, 2003; Sudhakar, 2002; Asan, 2003; Key, 2003). Another inconvenience arises from limited funds available to secondary schools. So utilizing one of the commercial online courseware systems is quite difficult (Pavlik, 2000; Jacobs & Meyer, 2001; Godwin-Jones, 2003). Furthermore, there is no technical staff regularly on hand to help teachers develop and maintain their online courses.

A feasible solution to the problems mentioned is the creation of dynamic educational web sites, where educational material is categorized and thus easily retrievable, allowing the association of subjects (Lazarinis, 2002). Additionally, in order to minimize development costs and time, the maintenance should be left mainly to educators, who can promptly update their content without heavily relying on a computer expert.

A Template Based System

System Overview

According to Harbeck & Sherman (1999) online learning sites should be simple with a concrete design. To realize it and to alleviate the inconveniences mentioned formerly courses are categorized in our automated system for developing online courseware. In this layout (see figure 2), learners can easily navigate through the online courses.

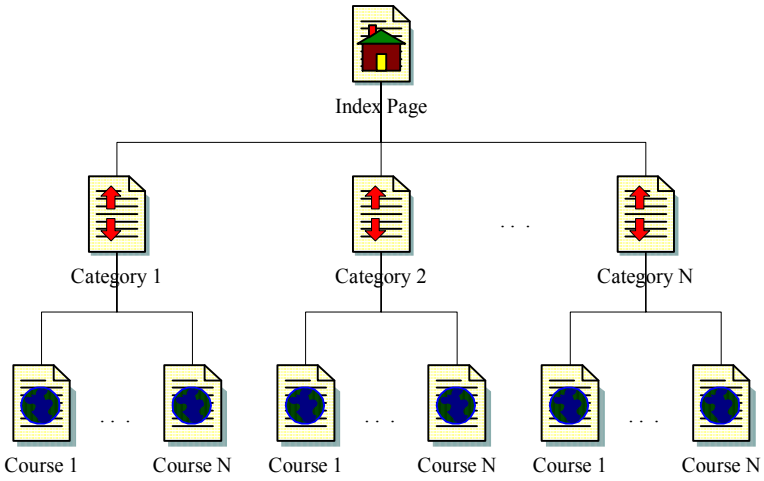

Figure 2: Thematic based structure of the system

Web content is shaped dynamically as the learner traverses down the tree, advancing to the courses. To dynamically create the courses the construction of a database is necessary in which educational material of varying formats exist along with a number of templates. The templates will be dynamically filled with the educational material to structure the courses. In this framework we identify three discrete roles: Administrator, Educator, and Learner.

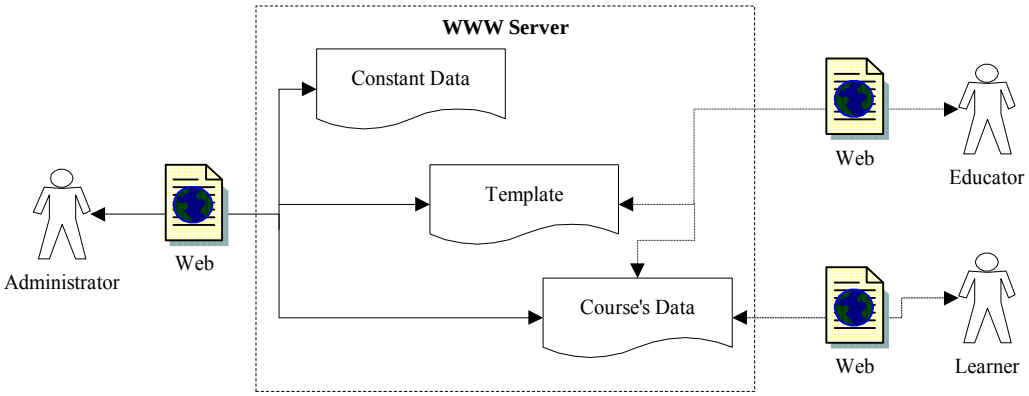

Figure 3: Simplified view of the architecture

These roles (see figure 3) interact with the system through a set of web interfaces, performing their assigned tasks or simply studying the online content.

Administrator

This specific role is responsible for the overall administration of the system. He or she handles the general data, the creations and modifications of other users, and generally the smooth operation of the system.

His or her first and most important function is the creation, manipulation and association of categories of courses and the population of the database with relative information.

Another important administrative function is the translation of the standard headers and messages existing on the system's pages and templates from English to the institute's native language. The main language of the system is English. However, unlike most of the commercial automated systems that support mainly English, we aim to expand our language database through the utilization of an additional database table. This table contains the standard system headers and text. Through a web interface administrators can translate all the standard headers to their native language (see figure 4). These new data are inserted in the database and used automatically on the produced web pages. The selection of the appropriate headers is based on the language and the character set selected by the administrator.

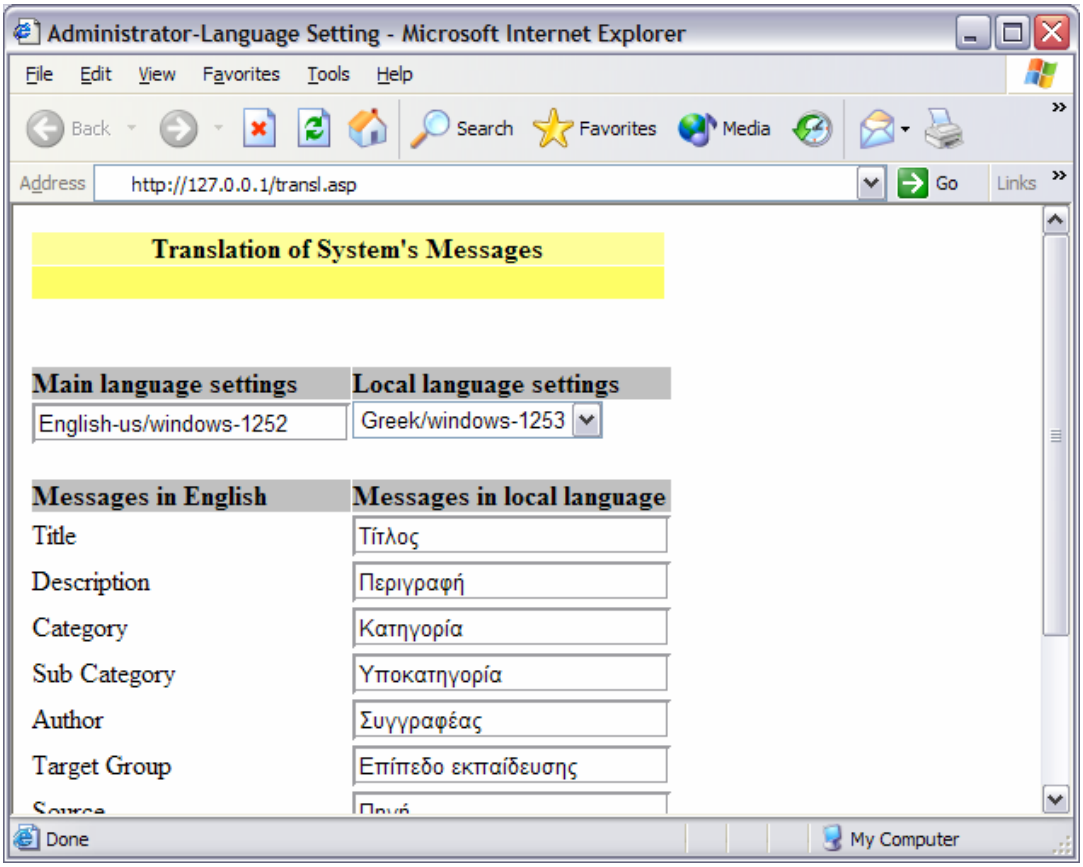

Figure 4: Translation from English to Greek

Other operations of the administrator include monitoring of the system, production of statistics and creation of new templates. An administrator has access to all the repositories of the system in order to perform his or her duties.

The administrator, communicates with the database through a well defined set of web pages which restrict potentially hazardous access to the database and prevent abnormalities in the system behavior. The role of the administrator does not require advanced computer knowledge, since all operations are available through simple web interfaces. So, this role could be performed by a computer science school teacher.

Educator

An educator provides new educational material (or exploits the existing), selects the templates and constructs a particular course as a set of pages. The educational material may be either theory, in textual or other multimedia format, or exercise in one of the formats supported. Every course is linked with the appropriate category and a general title and a brief description is given. Table of contents can be constructed automatically based on the pages that structure the course.

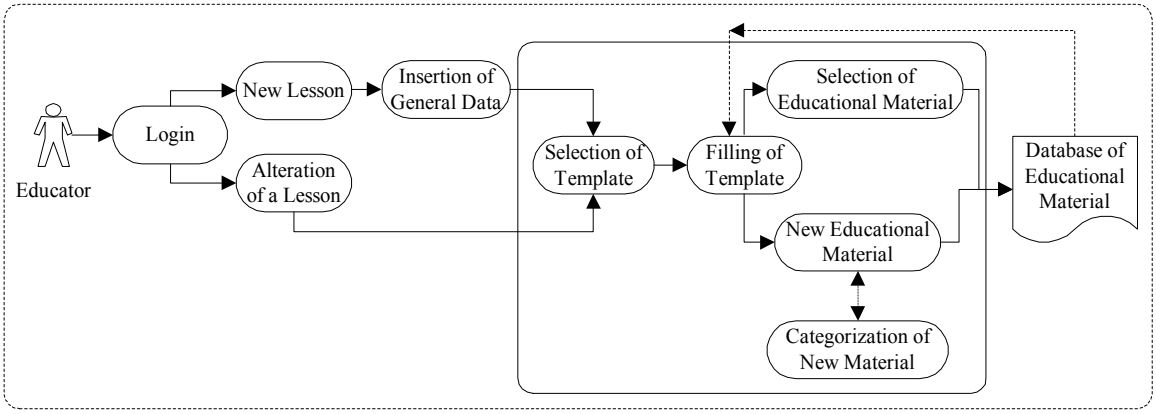

Figure 5: Dataflow diagram of a lesson's creation process

Through a web interface an educator performs the following tasks to construct an individual lesson (see figure 5) consisting of a set of pages:
Task 1:
Login and identification

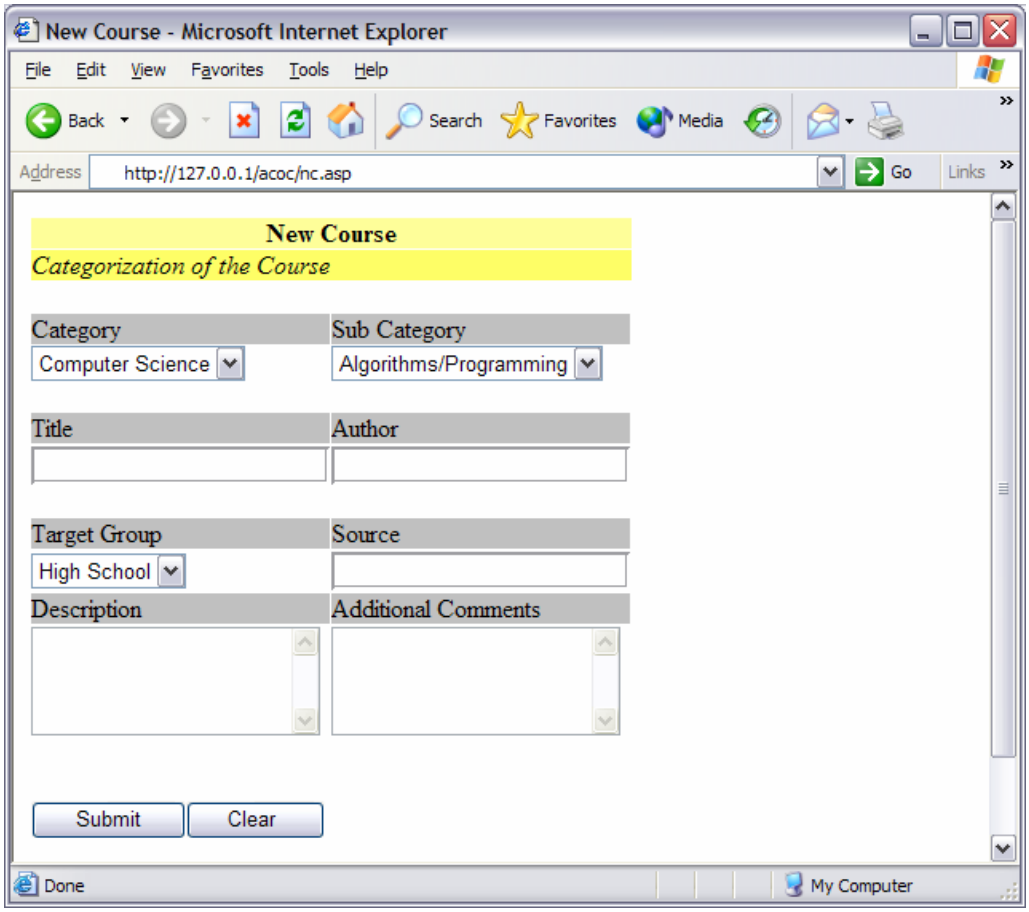

Figure 6: Selection of general attributes of a course

Task 2 (see figure 6):
Selection of category
Definition of title and author
Definition of metadata, i.e. description, target audience, source of data, etc

Task 3:
Selection of a template for a specific page (see figure 7)
Filling of the template with material
Possible linking to other pages
Repetition of last three steps and completion of tasks (i.e. insertion of the material in the DB).

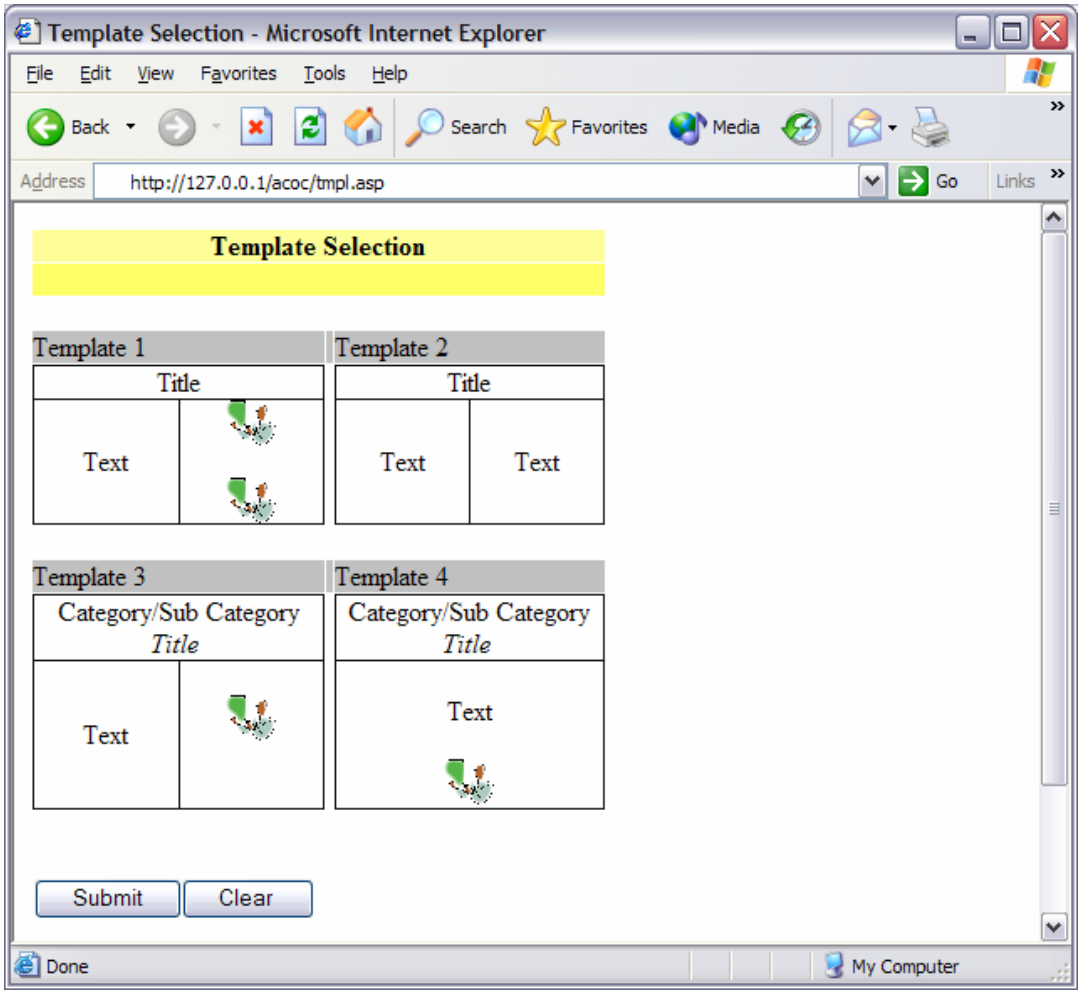

Figure 7: Selection of the template

Thus, educators can create new online courses at any time. In other words the maintenance phase of our system is quite different from that portrayed in figure 1. Here the educator handles the requests of the learners, or their needs for change and promptly adjusts the educational pages. Educators can also associate their online course, with the aid of links, to already available courses, thus enabling cross-reference linking between topics.

Learner

The final user type identified is called learner. This particular user type can connect via Internet and through a Web interface may follow one of the available courses. First the category is chosen and then the desired course based on the title and the possible brief description.

From that point on, the respective sequence of web pages is presented to learners and it can be followed linearly or in the order of their choice. Pages will contain either theory or exercises which should be completed by the learner. If the exercises are multiple choice, or short calculations, then the assessment is instant. When the exercises require extensive answers, then these replies will be sent to the appropriate educator and a respond will be sent after the marking process.

The data presented and the layout of the information are based on the template chosen by the educator. For instance figure 8 shows a sample template. The information in curly brackets will be replaced by actual data when "executed" in the server and projected back to the client, as shown in figure 9. The information is dynamically filled as retrieved from the database. The parameters pass from one page to the other as attributes via URLs.

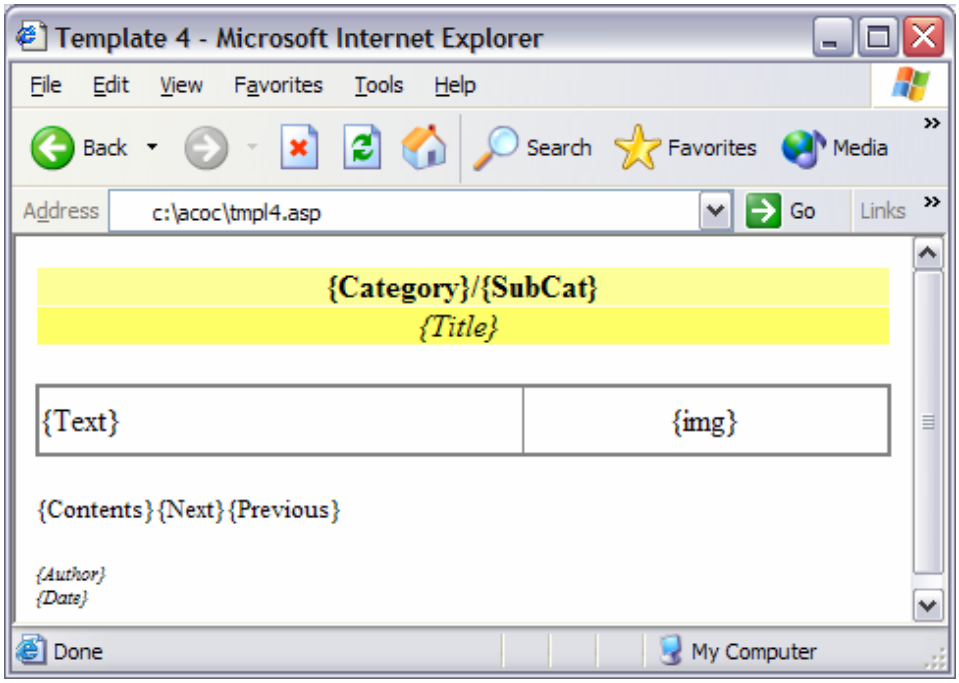

Figure 8: An example of a template

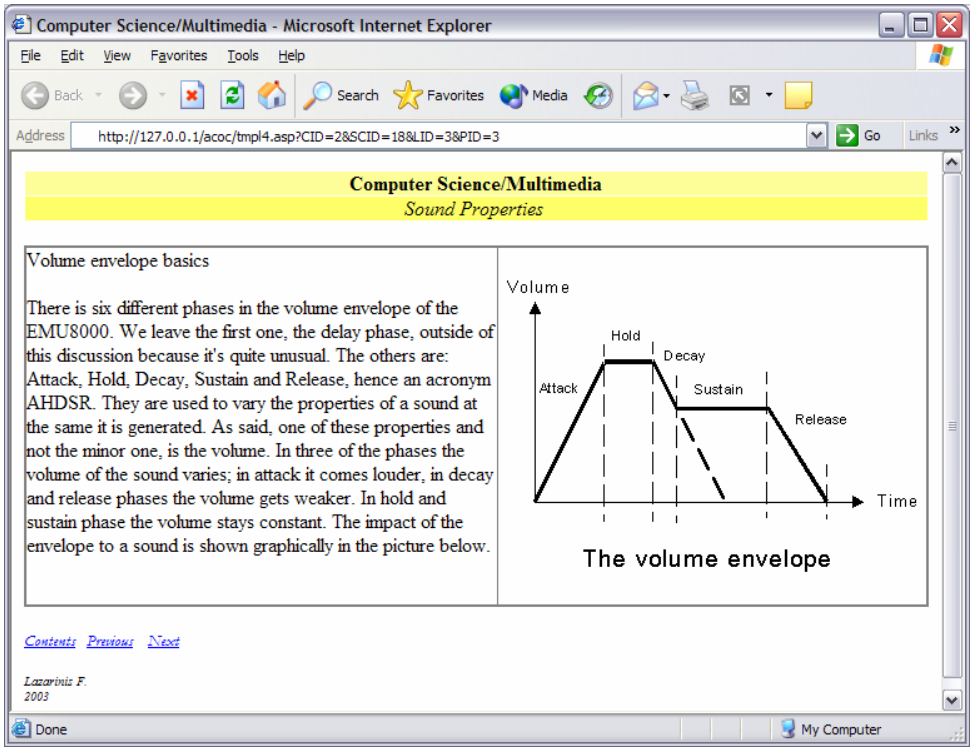

Figure 9: A page of a course based on the template of figure 8

Database Scheme

All the constant and variable data resides to a repository, implemented as a single database in this first phase of the project's development cycle.

In this context we identified the following entities: Educational Material, Type of Educational Material, Category, Sub Category, Template, Page, Course, User, Rights. An elementary Entity Relationship Diagram (ERD) (Yourdon & Constantine, 1979), as perceived during the analysis, would be as in figure 10.

Clearly, to form a course an educator must create a set of pages, which in turn is a set of educational material. When the learner navigates the online site, the courses will be constructed dynamically based on the implementation rules and strategies.

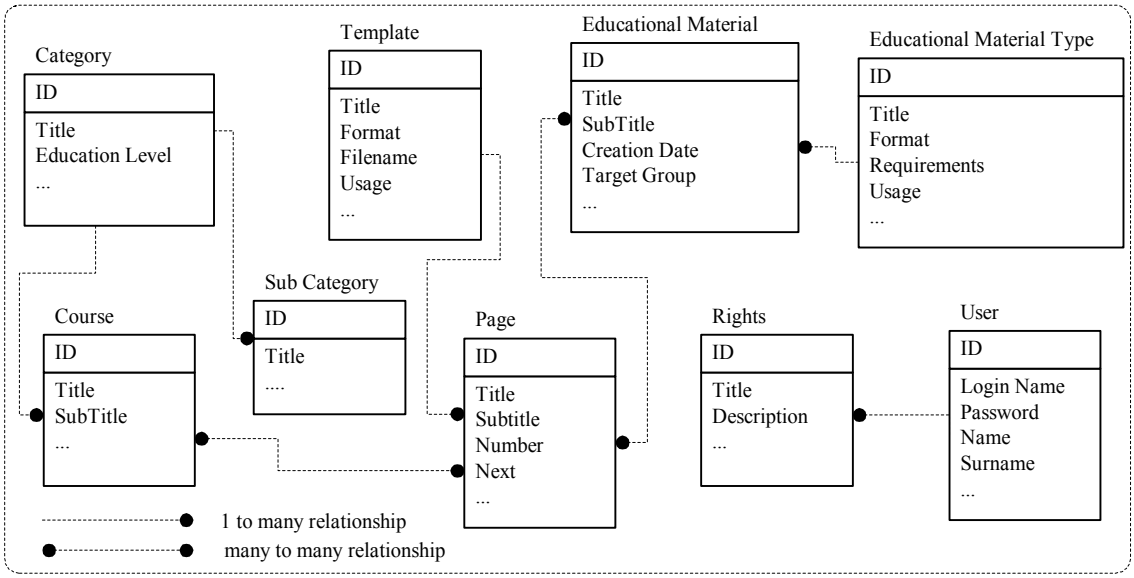

Figure 10: Database scheme

Implementation Details

Suitable technologies for implementing such a system are Active Server Pages (ASP) or Java Server Pages and an ODBC or JDBC compliant database. ASP and MS-Access are preferred. MS-SQL Server could be used as an alternative but costs more and also strong DBMS administration skills are required. All textual data reside on the database and templates and graphical information reside on the disk, referenced in the database fields.

To set up the system, a clean copy of the MS-Access database (as it is administered easier than SQL Server and costs less) needs to be copied to a folder on a Web Server's disk. Secondly a virtual directory on their Web Server Software, e.g. Internet Information Server, must be set up and an ODBC connection to the database should be created. Finally, an additional folder on the Web Server machine is needed. In this folder the produced html files are to be stored.

Evaluation

An alpha version of the system is developed under Windows 2000 and is available only on the intranet of the institute so that to avoid potential security problems that would be caused from the direct Internet access of an in progress project. The system was tested only against MS-FrontPage, which is the tool of choice of many institutes worldwide when it comes to online courseware and the only available tool in the institute's native language. Our goal was to reach some initial conclusions that could be incorporated in the development and we aimed at identifying the anticipated advantages of such a system.

The prototype interface of the system (see figures 5 to 9) was demonstrated to a dozen of high school teachers, for 20 minutes. Then, the teachers, acting as initial evaluators, used a series of interfaces so to construct a web page with a title, some text as theory and one image. Teachers could then access their page from the automatically produced URL added on the index of the site, under the category they had chosen (e.g. computer science, physics etc).

Most of the participating teachers were beginners and not advanced computer users (table 1). The task assigned to the subjects was the creation of a page containing a heading, one small paragraph of text and one photograph. 83.33% (10/12) of the subjects completed the task in less than 20 minutes. Most of the time spent in typing the text. The remaining 16.67 (2/12) completed their task in 30 minutes.

Table 1: Computer knowledge of the 12 teachers

Computer handling capability	
Beginner	9
Intermediate	3
Advanced	0

During the evaluation only three teachers faced a problem and we had to help them. The problem had to do with the insertion of the photograph. Teachers could choose between three template alternatives. 91.67% chose the template where the heading is centre aligned with a larger font and the photograph is on the right side of the text.

In the course of the process the administrator had to create 10 new categories and subcategories to enable users to categorize their courseware. Category and subcategory were then displayed automatically in the produced page. Also the heading (lesson's title) was automatically centre aligned and emboldened, based on the template used.

The 12 participating teachers were asked to complete the same task using the MS-FrontPage. Initially teachers had to be trained for 1 hour. So the training time was substantially greater in this case. Subsequently they tried to create a web page containing simply a title, some text and an image. First they had to type, embolden and centre align the title. So they needed to be more competent users.

The second step was to type the text and thirdly, following the same procedure as before, to insert a picture. They could only insert the picture below the text and not on its right side. To fulfill such a request the insertion of a table is needed, a technique not taught to the teachers. In order to publish their results on the intranet they had to save the file as an html file, to name it and to put it in a specific directory in the hard disk.

The average time required for the completion of the task was approximately 45 minutes. The problems raised during the process related to the formatting of the text, the saving of the file and the publishing of the final outcome on the institute's intranet. The insertion of the picture was dealt in the previous experiment, so in this case the necessary steps were easily accomplished. Finally, another issue that troubled 83.33% of the participants, was the placement of the picture on the right side of the text.

In table 2 we quantify the results of the comparison of the two applications. As is can be concluded more competent users needed in the second case.

Table 2: Results of the comparison of the system

Our system		**MS-FrontPage**
Training time	20 min	60 min
Average time for task completion	22 min	45 min
Actions	Login Category selection Sub category selection Template selection Typing of title Typing of text Insertion of the picture	Create a blank page Font size selection Type the title Embolden the title Centre align the title Typing of text Insertion of the picture Naming the file Saving it to a specify directory Insert a link on the index page
Problems	2 [Creation of categories, insertion of a picture]	5 [Text formatting, saving of the file, publishing of the created picture, insertion of a picture, placement of a picture]
Administrator's interventions	2	5

After this small scale evaluation, the main conclusion drawn is that our system is far simpler compared to MS-FrontPage. The only technical capabilities required were using the mouse, typing text and inserting images from the hard disk. On the contrary, utilizing MS-FrontPage was more difficult as they had to learn a completely new environment, save the file to a specific directory and work with a computer expert to publish it. Almost all of the actions performed in the second mode of developing the page required technical knowledge, whereas in our system the process is a sequence of one step actions.

Using MS-FrontPage, links on the index of the institute's site should be manually inserted so as to access the new pages. Another point resulting from this short assessment is that the produced page was more appealing in the first method. Background color and layout, associated with the chosen template, made the automatically produced page more attractive than the one constructed using the MS-Office tool. Applying layout and colors is obviously a feature of MS-FrontPage but requires more actions and knowledge on behalf of the users.

Another major advantage of our system is that navigation buttons are added on the fly and thus navigation is not a source of worry for the creator. Broken links are not an issue as every page is constructed dynamically and is definitely present. Pages consist of objects which are named reusable entities while in a static html file information is unexploited.

In order to be fair we compared the capabilities of our software with a commercial learning management system as well. We preferred WebCT (2003) which has powerful capabilities and an international established base of users. Many academic institutes rely on its power to re-engineer their distance learning courses. If we compare these two systems solely as systems that offer automation of the online learning process, then WebCT is the unquestioned winner.

However, it must be taken into account that our endeavor, at this point, concerns only secondary educational institutes where the online learning requirements are not as heavy as in the case of Universities. So it must be simpler than WebCT, more affordable and with fewer capabilities in order to be easily mastered. These are the mainstreams of our work.

Another plus point of our work is the dynamic generation of content material. Through the use of the centralized repository and the structured data, courseware is dynamically assembled and thus the material is reused. Moreover, another advantage over other automated learning systems is the ability to adjust to various languages without the need to be re-developed.

Concluding, the evaluation proves that our software is able to meet the needs of the secondary educational schools wishing to offer course materials online so that students can continue their classroom studies at home. The prospect of remote teaching and assessment is possible at a fraction of the cost, with respect to time and money, of a commercial tool. However a more rigorous evaluation is needed in order to realize the real gains and pitfalls.

Synopsis and Future Work

In the previous sections we presented briefly some of the problems in the development of online learning systems and we recommended a framework where teachers could construct their own courses, without the constant involvement of the development team. In this framework, courses are organized as a set of pages which could be followed in the order of his or her choice. Each page is a combination of educational material presented dynamically to the learner based on the template used. All the required information is amassed in a database which is projected back to the learner.

Within this "PowerPoint like" system, most of the problems and inconveniences confronted in the development of other online learning systems, as presented previously, can be resolved. The learning material may be constantly updated, links function regularly and association between different subjects is feasible.
Moreover, its maintenance is a straightforward procedure, reducing the development costs and time. Finally, the style of the pages is uniform and the learning process can be interactive.

Our implementation attempts, at this first phase, focus on constructing a system that would be usable and easily mastered by high school educational personnel. Teachers would be able to create lessons that contain theory, images and questions of false/true type. We tested a prototype of the system using a dozen of teachers of various specialties and we reached some useful conclusions, which were taken into consideration.

However, it is our conviction, that to implement all functions of the system, the analysis and design phase should be extensive and various categories of educators should participate. Thus, the analysis and design stage would be the most resource intensive in terms of human effort. But, in this way, the maintenance stage will involve less effort and consequently the delivery of new courses would be rapid.

In the analysis and design phase, aside from the creation of the database and the set of the initial templates, it is necessary to confront a series of other issues. These issues regard the permissions and responsibilities of the three roles. Also it is important to establish rules for the various operations of the system. Especially this final requirement is of critical importance as it will allow the unproblematic implementation of the system and prohibit conflicts during the population of the database.

Another issue to be addressed is the documentation and ownership of the educational material, possibly though the utilization of metadata (BECTA, 2001), and the construction of rules for future alterations of the educational material. These issues should be dealt with in the second phase of the development.

Nevertheless, the system should balance simplicity of use and wealthy of features so that it will not require extensive efforts in order to be utilized rendering it complicated and eventually not practical.

References

Asan, A. (2003). Computer technology awareness by elementary school teachers: A case study from Turkey. *Journal of Information Technology Education, 2*, 153-164.

BECTA, (2001). *Metadata in Education*, Retrieved July 19, 2004 from http://www.ictadvice.org.uk/index.php?section=te&rid=1788&catcode=as_int_pub_03.

Berners Lee, T., Caillau, R., Groff, J., & Pollermann, B. (1992). World wide web: The information universe. *Electronic Networking: Research, Applications and Policy, 2* (1), 52-58.

Brahler, J., Peterson, N., & Johnson E. (1999). Developing on-line learning materials for higher education: An overview of current issues. *Educational Technology & Society, 2* (2), 1-8.

Butler, B. (1995). Using WWW/Mosaic to support classroom-based education: opportunities and challenges for is educators: an experience report. *Interpersonal Computing and Technology: An Electronic Journal for the 21st Century, 3* (1), 17-52.

Carswell, L., & Murphy, M. (1994). *Pragmatic methodology for educational courseware development*, Retrieved July 19, 2004 from http://www.ulst.ac.uk/cticomp/carswell.html

Crossman, D. (1997). The evolution of the world wide web as an emerging instructional technology tool. In B. H. Khan (Ed.), *Web-based instruction* (pp. 19-23), NJ: Educational Technology Publications.

Dagiene, V. (2003). Focus on the pedagogical dimension in ICT literacy for teachers. In McDougall, A., Murnane, J. S., Stacey, C. & Dowling, C. (Eds.), *Proceedings of the ICT and the Teacher of the Future - Selected Papers from the International Federation for Information Processing Working Groups 3.1 and 3.3 Working Conference* (pp. 27-29), Melbourne, Australia: ACS.

Daniel, J. S. (1996). *Mega-universities and knowledge media: technology strategies for higher education*, London: Keegan Press.

Godwin-Jones, B. (2003). Emerging technologies. Tools for distance education: towards convergence and integration. *Language Learning & Technology, 7* (3), 18-22.

Golas, K. C. (1993). *Estimating time to develop interactive courseware in the 1990s*, Retrieved July 19, 2004 from http://www.coedu.usf.edu/inst_tech/resources/estimating.html.

Grieve, C. (1992). Knowledge increment assessed for three methodologies of teaching physiology. *Medical Teacher, 14* (1), 27-31.

Harbeck J., & Sherman T. (1999). Seven principles for designing developmentally appropriate web sites for young children. *Educational Technology, 39* (4), 39-44.

Jacobs, G., & Meyer, D. (2001) *Online testing without the overheads – a computer-mediated solution for smaller players*, Retrieved July 19, 2004 from http://www.icce2001.org/cd/pdf/p07/NZ008.pdf.

Key, P. (1999). Teaching teachers to be computer literate. *Philadelphia Business Journal*, Retrieved July 19, 2004 from http://www.bizjournals.com/philadelphia/stories/1999/06/28/newscolumn2.html.

Kinshuk, & Patel A. (1997). A conceptual framework for internet based intelligent tutoring systems. In A. Behrooz (Ed.), *Knowledge Transfer* (Vol. II) (117-124), London: pAce.

Lazarinis, F. (2002). Proposing a discrete role architectural model for categorized online learning sites. *Paper presented at the 3rd International Conference on Information Technologies in Education ICICTE'2002*, July 17-19, 2002, Samos, Greece.

Markwell, J., & Brooks, D., (2002). Broken links: the ephemeral nature of educational www hyperlinks. *Journal of Science Education and Technology, 11* (2), 105-108.

McCormack, C., & Jones, D. (1998). What's it all about? In T. Hudson (Ed.), *Building a web-based education system* (1-28), New York: Wiley Computer Publishing.

McDonough, D., Strivens, J., & Rada, R. (1994). Current development and use of computer-based teaching at the University of Liverpool. *Computers & Education, 22* (4), 335-343.

Meier, P., & Simon, B. (2000). Reengineering undergraduate education by introducing internet-based information systems. *Paper presented at the 8th European Conference on Information Systems*, July 3-5, 2000, Vienna, Austria.

Miller, J., & Tittenberger, P. (2001). *Integration of WebCT into the distance education administrative model at the university of manitoba (one year later)*, Retrieved July 19, 2004 from http://naweb.unb.ca/proceedings/2001/P7Miller.htm.

Pavlik, P. (2000). Collaboration, sharing and society - teaching, learning and technical considerations from an analysis of Webct, Bscw, and Blackboard. *The Educational CyberPlayGround,* Retrieved July 19, 2004 from http://members.fortunecity.com/pgp5/Collaboration.Learning.and.Society.htm.

Phillips, R. L. (1992). Opportunities for multimedia in education. In S. Cunningham & R. J. Hubbold (Eds.), *Interactive Learning through Visualization: The Impact of Computer Graphics in Education* (pp. 25-35), Berlin: Springer-Verlag.

Serdiukov, P. (2000). Educational technology and its terminology: new developments in the end of the 20th century. In J. Bourdeau & R. Heller (Eds.), *Proceedings of ED-MEDIA* (pp. 1022-1025), USA: AACE.

Sklar E., & Pollack J. (2000). A framework for enabling an internet learning community. *Educational Technology & Society, 3* (3), 393-408.

Sudhakar, S. (2002). Technology partnership with K-12 schools. *The Technology Source*, Retrieved July 19, 2004 from http://ts.mivu.org/default.asp?show=article&id=942.

WebCT (2003). The structure of WebCT. *WebCT Conference Technical Report*, Retrieved July 19, 2004 from http://www.infodiv.unimelb.edu.au/telars/lms/.

Yourdon, E., & Constantine, L. (1979). *Structured design*, Englewood: Prentice Hall.