GDEEOM\textsuperscript{1}

AN INTEGRATED MULTIMEDIA SYSTEM FOR WEB-BASED UNIVERSITY EDUCATION

A. González, J.F. Nogueras and M.J. Pérez-Luque
Departamento de Ingeniería Informática, Universidad Antonio de Nebrija
Campus de la Dehesa de la Villa. C/ Pirineos, 55. 28040 – MADRID (SPAIN)
Correspondence Email: mjperzeluque@gmail.com

ABSTRACT
This paper presents a pilot educational project aimed to develop an integrated system to allow the teaching community to experiment and evaluate the benefits of multimedia lessons. The key issues of the project are the development of a structured database, the design of multimedia templates, the implementation of a web based authoring application that enables professors to be course builders, and the delivery mechanisms to dynamically generate multimedia content. In this paper we discuss the concept and realization of the system as well as the first results accompanying the project.

Keywords: E-learning Technologies, Multimedia Languages, Structured Multimedia Databases, Authoring Systems, Streaming Video, Web-Based Learning.

1. INTRODUCTION

The use of multimedia resources and web supported learning environments is nowadays a crucial issue in education. It has been proved that its use causes an enhancement of human learning and cognitive processes. This is particularly important when complex explanations are required [1]-[3]. The integration of text, image, audio-visual information, 3D graphics, animation, and simulation processes improves the quality of teaching and learning.

Nevertheless, multimedia learning systems are not widely installed within most universities, mainly due to three problems:

1. \textbf{Lacks of knowledge} of how effectively integrate multimedia information into the teaching process. Most applications are technology-driven instead of being content-driven. As a consequence, very fancy but useless designs are produced. It is important to study what the multimedia communication language is, and how to add interactivity in an appropriate way. Usability techniques should study how to adopt a more user centric approach and how to evaluate real student’s improvements [4]-[6].

2. \textbf{The negative response of professors to be course builders}. Professors usually lack the tools and experience to create multimedia data. As a consequence, they are reticent to carry out the cumbersome task of editing, integrating and maintaining multimedia course materials [7]-[8]. Many approaches are under research to fill this gap, particularly to create authoring applications that do not require any programming knowledge and are easy to learn in a short period of time.

3. \textbf{The cost of integrated multimedia systems design and implementation}. The complexity of establishing a multimedia learning infrastructure is well known and studied. Reliable, scalable, open, concurrent and transparent systems are difficult to create. There are many components to be designed and implemented to create effective delivery mechanisms to offer multimedia content over the Internet, specifically on a Web basis so that it is accessible to every student. Recent studies point out that research should focus to achieve two main goals [9]. One: any system should contemplate the share and reuse of multimedia data. Two: the

\footnote{\textsuperscript{1} Spanish acronyms of GDEEOM: \textit{Generador Dinámico de Estructuras Educativas Orientadas al Multimedia}.}
system must contemplate the separation of content, presentation and navigation as a very important requirement.

In this paper we describe how GDEEOM has overcome the above mentioned difficulties. The success of the project is mainly due to the following items:

- **Structured data storage, searching mechanisms, reuse of information, content-presentation independence** and dynamic generation of multimedia content are achieved by the use of multimedia templates and the design of a well structured database.

- The **editing process** has been simplified significantly by the development of a web based authoring tool, which is totally transparent for professors. Usability tests have been run to verify the developed application.

- The **designed scalable multi-layer architecture** achieves a system which is accessible to every student, easy to maintain, scalable and reliable. As scalability is a key issue in developing multimedia applications, especially when a large number of users are expected, the following multi-layer architecture has been designed: the client layer for thin clients, the middle layer for the application server and web server, and the back-end layer for the databases. This architecture allows us to distribute client requests and back-end services among several sets of collaborative cluster servers.

The proposed system architecture is shown in Figure 1.

![Figure 1: GDEEOM System Architecture.](image)

It has three main components:

A) **The Database System.** The database stores templates, raw multimedia material and metadata.

B) The **Multimedia Content Application Server**, which has two main components, the Data Parser and the Database Content manager and Authoring Application. The former works with the Delivery System to generate the multimedia lessons on-the-fly. The later provides editing mechanisms for professors to manage the database and to create multimedia lessons.

C) The **Multimedia Delivery System**, that delivers the dynamically generated multimedia content over the Internet to the student’s web browser. The streaming server is used to adapt audio-visual content to the user’s speed connection.

The database system and the multimedia content application server are the main focus of this paper, which is organized as follows. The following section describes the database system. Next we explain the designed and implemented multimedia templates, followed by the description of the components of the Multimedia Content Application Server: the Database Management System, the Multimedia Authoring Application and the Data Parser. The following section describes briefly the used software technologies. We then present experimental results, and end with conclusions and future work.
2. DATABASE

As was commented previously, the system is based on the dynamic generation of web pages. This fact breaks the traditional model of a website, where HTML pages are stored in a web server. In our application, there are no physical web pages stored. The database system stores multimedia data and metadata. The Data Parser gathers that information from the database and dynamically generates the HTML pages upon request. This approach presents several advantages, which could be summarized as follows:

- Data is stored independently of the way it will be presented. As a consequence, the same multimedia raw material could be presented in a variety of multimedia lessons and be integrated in many different ways.
- Data duplication is avoided and data reusability is achieved.
- Data is structured and classified by the use of metadata, which is descriptive information about resources for the purpose of finding, managing, using and reusing of them more effectively. Searching mechanisms can be implemented to locate data in an easy and efficient way. The implementation of intelligent agents could also be considered.
- The use of multimedia templates for creating multimedia lessons simplifies the editing process. If there is a change in the template, this is the only data to edit; the rest will stay the same.
- Multimedia lessons are easy to maintain, the role of the web master or administrator has been cut down considerably, because there are no physical HTML pages to maintain. It is the professor who assumes the content and web manager tasks at the same time, through the authoring and database management application.

This approach also has two drawbacks. First, the possible overload of the system when a large number of users are accessing the database. Second, the dynamic generation of the lessons could be slower than desired due to the access to the database and the parser process. But these problems have been solved by a careful database design, a cache system, and a good cluster structured server working with efficient concurrent accessing and generators mechanisms.

The designed database is structured as follows:

- **File system to store multimedia raw material.** Text (plain or HTML), image, image maps, audio, video and generic data (not specified a priori like, for example, PDF files).
- **Relational database to store metadata.**
  - Course information and multimedia metadata for multimedia lessons.
  - Multimedia templates.
  - Multimedia raw material metadata: title, data type, format, size, date, keywords, author, and related topics.
  - Multimedia raw material references to the file system

![Course Structure](image)

**Figure 2:** Course Structure.
The designed course structure is presented in Figure 2. Multimedia lessons are accessible by students at a single starting point: the professor. The professor will offer one or several online courses. Each course will be organized into several sessions; each session will be composed of one or more conceptual units (CU). The CU is the last piece of information within the hierarchy and corresponds to the integration of multimedia information. The designed hierarchy, which has been designed following both educational and usability criteria, is a proposal to organize in a visual and usable way the big amount of information that usually composes a course. Next Section explains the designed and implemented multimedia templates for the integration of multimedia material.

3. MULTIMEDIA TEMPLATES

The use of multimedia templates simplifies the editing process for professors, lessens maintenance tasks, but most important, guarantees the quality of the produced teaching material by establishing carefully design standards. The three main issues to consider when designing multimedia templates are related to pedagogical, navigation and usability requirements:

- Integration of multimedia elements: which elements should be integrated for a specific educational purpose and how.
- Navigation: how to provide an easy way to navigate through the teaching materials without getting lost.
- Usability: how to make the learning process “usable” and worth it.

Within this project, an interdisciplinary team of multimedia experts and professors has designed a number of multimedia templates which cover most teaching scenarios. Each designed template is flexible enough to include different multimedia types of information and a varied number of them. Also, coherent navigation and usability is achieved because all the templates follow the same scheme and general design. Therefore, students get used to the way they interact to the system and focus their attention on the lesson, rather than the interface. Current templates could be modified if needed, as well as new templates could be added to the database.

![Figure 3: Generic Template Structure.](image)

The generic template structure, depicted in Figure 3, is divided into three main parts: head, body and foot. The head includes information related to the conceptual unit title, professor and course.
The foot provides the session title and navigational information: on the left a dynamic HTML Session Index, with the Conceptual Unit Index for each session; on the right, a navigation bar to go back and forth between sequential conceptual units within each session.

The head and foot are common parts to all the templates. It is the body which changes for the specific chosen template and offers a wide variety of multimedia elements integrations. To illustrate these concepts, Figures 4 and 5 describe one of the implemented templates with an example.

**Figure 4:** Template Structure A.

The multimedia elements of template A’s body, as shown in Figure 4, has three main components. The lower side consists of a common text set which is present for all the available segments. The upper part is divided into two areas which are associated and numbered as segments: a text set on the left and a variable multimedia element on the right. By clicking on the segment tab on the right part, both areas change: the multimedia element on the right (e.g., image, video) and the explicative text on the left. The text set is either plain text or HTML with internal or external links. The flexibility of this template is very significant: one to N segments, any multimedia element on the right hand side of the window, and plain or HTML text sets.

**Figure 5:** Example of conceptual unit with template structure A.
A multimedia lesson created with template A is shown in Figure 5. It consists of a conceptual unit with five segments. On the left, segment one is selected (plain text plus image). On the right, segment two is selected (HTML text plus video).

As presented in Figures 4 and 5, the navigation and generic structure of the multimedia lessons is preserved. The multimedia content only changes the screen area of the lesson’s body. As a consequence, the student visualizes the lessons in a very coherent, usable and easy to navigate course structure. Fourteen templates have been designed to cover most teaching scenarios, ranging from simple to very complex multimedia data integrations.

Next section explains how the Multimedia Content Application Server manages the templates and their associated information on the database.

4. MULTIMEDIA CONTENT APPLICATION SERVER

4.1 Database Management
A web based tool to manage the multimedia database has been developed. This is a very easy tool for professors; they just need to know how to navigate through a web browser, upload data and use filtering and searching mechanisms to find what they are looking for.

Figure 6 shows two interface screens of this management tool. On the left, the uploading interface which is used by professors to introduce multimedia raw material and its associated metadata into the database. On the right, the available searching mechanisms to find data into the database are presented. The professor can search data by format, title, size, date, related topics, and keywords.

![Figure 6: Interface to manage the database to upload data (left) and search data (right).](image)

4.2 Multimedia Authoring Application
The Authoring Application to create multimedia lessons is a web based tool, integrated with the Database Management application. Figure 7 shows how data flows during the authoring process. First, the professor, through the authoring application, retrieves the desired template, the multimedia course data structure, which has its reference fields empty. Second, the professor specifies the number and types of multimedia data, and selects from the database the correspondent multimedia files. Then, the authoring application fills the data structure with references to the database material and stores the created multimedia lesson into the database.
As mentioned above, the professor follows very simple steps to create multimedia material. From the user interface point of view, the developed tool is very easy to use. First, the professor logs into the system and then chooses to create or edit a course, a session or a conceptual unit. Once one is at the conceptual unit level, the following steps must be followed:

1. Select the template from a list, which includes template images and other helping information.
2. Upload the template and customize its specific characteristics (e.g., number of multimedia elements). The interface offers help images with explanations (see Figure 8).
3. Insert data from the database. Use searching mechanisms if needed. This interface is the same as shown in Figure 6. To insert plain or html text, a common text editing tool is available.
4. Preview the created conceptual unit. Edit if necessary.
5. Store it in the database.

**Figure 7:** Data flow during the editing process.

**Figure 8:** Authoring interface: the template is already selected.
The developed authoring tool allows the professor to create complex multimedia lessons without any programming knowledge but the simple use of a web browser.

4.3 Data Parser

The Data Parser is the responsible for gathering data from the database and composing the HTML page to be delivered to the student’s browser. The parser is general enough to understand all the designed templates, including future template definitions. It has been developed following the DOM (Document Object Model) Tree architecture, which is a mechanism to explore the metadata as a node hierarchy, as shown in Figure 9. It works as follows:

- The parser reads metadata from database.
- Then it builds the DOM Tree for the specific lesson: root node, sub-nodes, and so on. Each node corresponds to a database element. Each element has associated the correspondent HTML tags.
- Finally, it gathers the references to the database multimedia row material for all the sub-nodes and creates the HTML code, to be sent as a String to the Web Server.

![Figure 9: Data Parser Process.](image)

The parser works with a set of established syntax’s rules, therefore it is independent of the specific stored data. Any change of template structures and/or multimedia data will not affect the parser process.

5. SOFTWARE TECHNOLOGIES

The system software components have been designed following the Unified Modeling Language approach. The developed Use Case methodology has considered three main actors: the Administrator, the Professor and the Student. The employed database is Oracle 9i. The parser and access mechanisms to the database have been deployed with XML DOM technologies (eXtensible Markup Language DOM) and Java/J2EE. The client side of the Multimedia Authoring System uses JSP (Java Server Pages) and DHTML (Dynamic HTML), while the server side has been developed with Java/J2EE. The Multimedia Content Delivery System makes use of streaming video technology and the WebLogic 7.1 platform.
6. EXPERIMENTAL RESULTS

GDEEOM is an ongoing project under the first testing phase, the database and the professor editing side. The evaluation of the system has been carried out with professors with different level of expertise, belonging to three Departments of the University: the Mechanical Engineering Department, the Linguistics Department, and the Informatics Department. Most of them had created a traditional web site with course materials like PowerPoint presentations, PDF files, and the like. The testing phase consisted in allowing them to create several multimedia lessons with the Authoring Application (AA) during different days. Usability questionnaires were run for all of the participants after their experimentation with the application. The designed questionnaire had fifteen questions, aimed to evaluate learnability, efficiency, effectiveness, productivity, user interface, functionality, memorability, system errors, and subjective satisfaction [10]-[12]. Answers ranged from one to five (bad to good). The results are presented in Table 1.

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>It was easy to learn to use the application</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>I can effectively complete my work using the AA.</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>I am able to complete my work quickly using the AA.</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>I am able to efficiently complete my work using the AA</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>The interface is pleasant, I like it</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>The organization of the application on screens is clear</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>It has all the functions and capabilities I expect it to have</td>
<td>4</td>
</tr>
<tr>
<td>8</td>
<td>I found enough templates to create multimedia teaching materials</td>
<td>5</td>
</tr>
<tr>
<td>9</td>
<td>I found the various functions in this application were well integrated</td>
<td>5</td>
</tr>
<tr>
<td>10</td>
<td>The information (such on-screen help images and messages) provided with AA is clear and easy to understand</td>
<td>4</td>
</tr>
<tr>
<td>11</td>
<td>It was easy to remember how to find my way to complete tasks</td>
<td>5</td>
</tr>
<tr>
<td>12</td>
<td>The application gives error messages that clearly tell me how to fix problems</td>
<td>3</td>
</tr>
<tr>
<td>13</td>
<td>I feel comfortable and confident using the AA</td>
<td>5</td>
</tr>
<tr>
<td>14</td>
<td>Overall, I am satisfied with how easy it is to use the AA</td>
<td>5</td>
</tr>
<tr>
<td>15</td>
<td>Overall, I am satisfied with the AA</td>
<td>5</td>
</tr>
</tbody>
</table>

Table 1: Usability Tests Results for the Authoring Application.

The right hand side of Table 1 presents the average results from all the participants. So far, tests have been quite reasonable. The experimental results pointed out that professors did like the authoring application, but they suggested that system errors and on-screen information could be improved. Also, suggestions were made to add some functions to make even easier the editing process.
CONCLUSION AND FUTURE WORK

We can conclude that the development of this project has provided us with adequate multimedia tools to improve the quality of teaching in our University. The experimentation during the first testing phase of GDEEOM has been very satisfactory. The proposed templates have been proved to be sufficient for most teaching scenarios. The organization of the database has been considered by the testers as well structured for use and reuse of information. The web based authoring tools have been confirmed as being appropriate applications to efficiently and effectively create multimedia lessons, in a very easy way.

Student’s tests will be run during the second phase of the project. Future work will include improvements from both testing phases’ feedback. The first application of the system, which will operate with the Linguistic Department, will be used to create multimedia courses to teach Spanish to the growing immigrant community. This project will be supported by a governmental public fund.

ACKNOWLEDGEMENT

We would like to acknowledge and thank all the people who have contributed and shared their expertise to help develop and test GDEEOM. Special thanks to Antonio Rodriguez Senin and Jesus Pancorbo for their valuable contribution to the project.

REFERENCES