

WEB APPLICATION FOR TIMETABLE PLANNING IN THE HIGHER TECHNICAL COLLEGE OF INDUSTRIAL AND TELECOMMUNICATIONS ENGINEERING

Dra. Marta E. Zorrilla Pantaleón

*Dpto. Applied Mathematics and Computer Science
Avda. Los Castros s/n 39005 Spain
zorrillm@unican.es*

Dr. Eduardo Mora Monte

*Dpto. Applied Mathematics and Computer Science
Avda. Los Castros s/n 39005 Spain
moraem@unican.es*

Dr. José Luis Crespo Fidalgo

*Dpto. Applied Mathematics and Computer Science
Avda. Los Castros s/n 39005 Spain
crespoj@unican.es*

ABSTRACT

The aim of this paper is to show a Web application developed in the University of Cantabria which allows for timetable management in an Engineering school with the peculiarity of having been developed using Informix Web datablade. First, the proposed Informix architecture to manage complex data is described; next, our project as application case is showed and finally, the experience obtained is summarized.

KEYWORDS

Web application, Applied Information Systems, Datablade, Complex Data Types

1. INTRODUCTION

At present, Relational Database Management Systems (RDBMS) are the support for most data centres. These systems accept only certain types of simple data and a restricted number of indexing techniques. As the scope of application of computer technology is increasing at great speed, it is becoming more and more necessary to use other types of more complex data, such as documents, images, multimedia data, time series, geographical data, etc. At present, the best representation for these complex data is that of "objects", understood in the sense of object-oriented programming, as these can contain all the details of the information, its structures and its methods.

Applications require high-performance storage, retrieval and analysis of non-structured data. To this end, some institutions have turned to non-relational database systems to support this type of data, such as the Object Oriented Database Management System (OODBMS). However, non-relational database systems are often more costly to develop and maintain than those using the relational model. The lack of a common query language such as SQL greatly increases the complexity of development and integration.

Thus, at present, the most suitable approach to the problem consists in extending traditional relational DBMS to manage rich, diverse data without sacrificing the reliability and scalability characteristics of these systems. In other words, the RDBMS tends to adopt certain characteristics of the (OODBMS) to become the so-called Object Relational Database Management System (ORDBMS).

With a view to facilitating the understanding of these management systems and the applications supported by them, the present paper outlines a practical case developed in the University of Cantabria.

The software application presented solves the problem of timetable planning in the Higher Technical College of Industrial and Telecommunications Engineering of the University of Cantabria (UC) in which 8 different engineering degree courses are given. The original aim was not only to develop an administrative application to address this problem but also to provide a system through which teachers and students could enquire, among other things, their timetable, the availability of classrooms on a certain date or the subjects which dealt with a specific topic (search by content).

To do this, an application was designed with an interface which is simple, intuitive and easy to maintain and had access to a database for managing the class planning of each academic year, guaranteeing the integrity and consistency of the data.

This paper is organized as follows. In the following section, the architecture of an ORDBMS, particularly that offered by Informix, is described, explaining the solutions it offers for cases of data access via the Web and for full-text search. Section 3 outlines the application characteristics, presents the interface designed and describes the hardware and software solution used and finally, section 4 summarises the impressions and experience obtained.

2. ARCHITECTURE OF AN OBJECT-RELATIONAL DBMS

The first attempts to provide a system which could satisfactorily manage complex data consisted in designing and handling the logic of these data outside RDBMS, using this simply as a storage system. This involved the use of proprietary techniques which were difficult to co-ordinate with the functioning of the relational system.

Given that the field of application normally requires specific data types whose treatment is different in each case, it is pointless to attempt a global solution for the management of all possible complex data. The tendency of the main companies in the sector, such as Informix, Oracle, IBM or Sybase, consists in seeking specific responses for each field of application by means of the development of independent modules supported by a common layer, the User-Defined Data Type Manager (UDTM), which acts as an interface between each of the modules and the RDBMS. Developing one of these modules involves defining data types and the routines that operate them. Figure 1 shows the full system architecture in the case of Informix, where each specific module is called Datablade.

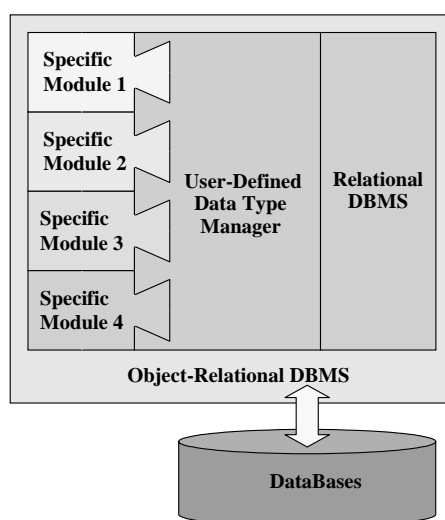


Figure 1. Modular architecture of an Object-Relational DBMS

2.1 DataBase driven web sites (WIO Datablade)

One of the main troubles of Web applications lies in the difficulty of neatly storing pages in the Web Server for their subsequent publication. An interesting response to this problem is that which is offered by Informix which, through its Web Integration Option (WIO), provides the resources necessary for developing complex database driven web sites. In other words, in this case, the Web pages themselves are a part of the information to be stored by the ORDBMS.

The Web Integration Option allows the user to create Web applications which incorporate and retrieve data dynamically from an Informix data-base: that is, to create HTML pages containing tags in which SQL instructions are included which, when are executed, generate results which are incorporated to the page.

In Figure 2, architecture of system with Informix Web Integration Option is shown.

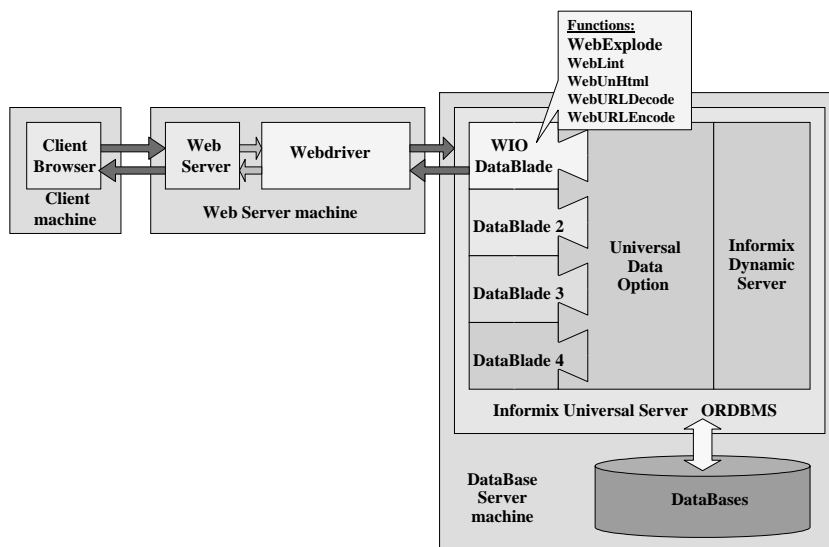


Figure 2. Architecture of System with Informix Web Integration Option.

WIO DataBlade has a user tool to aid in the development of applications, called Data Director for Web. This facilitates the generation and incorporation of HTML pages to the database, the coding of Informix tags included in the pages themselves and the incorporation of images and other multimedia objects. It also allows dynamic tags to be developed for reusing code which is repeated on several HTML pages (such as footnotes and headings) and, finally, it also enables the page to be previewed as it will be seen by the user, that is, executing all the Informix tags found in it.

Example of use

Through the URL: http://cuss6/escuela/webdriver.exe?MiVal=planest_search, the Client Browser requests the viewing of the AppPage "planest_search". When this instruction is received by the Informix Universal Server, this obtains the AppPage from the database and the WebExplode function executes the instructions included in the tags (Figure 3) and formats the page with the results (Figure 4), sending the resulting HTML file to the Webdriver. This sends this page to the Web Server and this, in turn, sends it to the Client Browser, which shows it to the user in the form shown in Figure 5. This page has more code than that affected by the texts of Figures 3 and 4.

```

<TD vAlign=top></TD>
<TD COLSPAN=2 vAlign=center>
  <SELECT name=titulacion_id MULTIPLE SIZE=5>
    <?MISQL SQL="SELECT titulacion_id,titulacion_nombre FROM titulacion
      ORDER BY titulacion_nombre">
    <OPTION VALUE=$1>$2</OPTION><?/MISQL></SELECT><BR><BR>
  </TD>
</TR>
<TR>
  <TD vAlign=top></TD>
  <TD vAlign=top>FOR:&nbsp: </TD>

```

Figure 3. Example of the content of an AppPage.

```

<TD valign=top></TD>
<TD COLSPAN=2 valign=center>
<SELECT name=titulacion_id MULTIPLE SIZE=5>
  <OPTION VALUE=ITEL      >INGENIERO DE TELECOMUNICACION      </OPTION>
  <OPTION VALUE=IIND      >INGENIERO INDUSTRIAL      </OPTION>
  <OPTION VALUE=IQUM      >INGENIERO QUIMICO      </OPTION>
  <OPTION VALUE=ITTEL     >INGENIERO TECNICO DE TELECOMUNICACION </OPTION>
  <OPTION VALUE=ITTELTC   >INGENIERO TECNICO INDUSTRIAL, ESPECIALIDAD ELECTRICIDAD </OPTION>
  <OPTION VALUE=ITTELIN   >INGENIERO TECNICO INDUSTRIAL, ESPECIALIDAD ELECTRONICA </OPTION>
  <OPTION VALUE=ITIMEC    >INGENIERO TECNICO INDUSTRIAL, ESPECIALIDAD MECANICA </OPTION></SELECT><BR><BR>
</TD>
</TR>
<TR>

```

Figure 4. HTML code returned by WebExplode function.

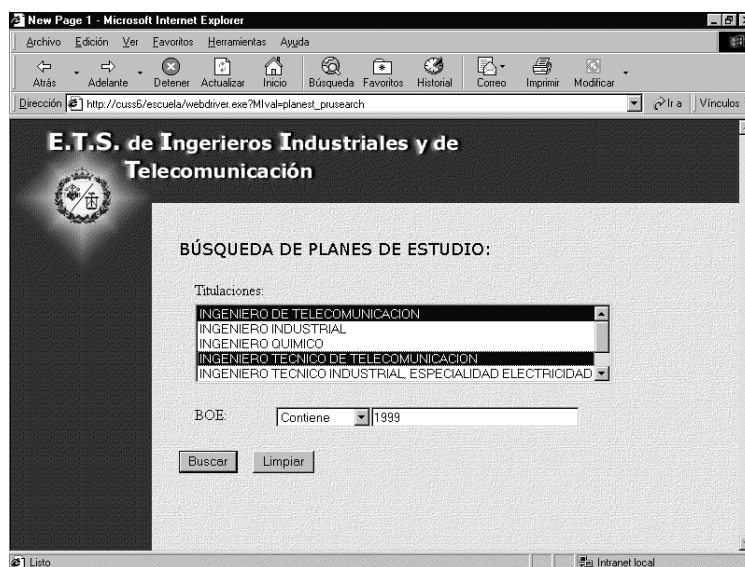


Figure 5. Aspect of requested page.

2.2 Information Retrieval DataBlade

For the retrieval of documents from the content of its text, Informix provides, among other tools, the Excalibur DataBlade. This module is independent of the access to information through the Web, although it can be used from it. This product allows several indexing techniques to be implemented, from inverse indexes to artificial neural networks, through which the following types of search can be effected: by word, by word with Boolean conditions, for an exact sentence, for a sentence by approximation, by proximity, etc.

3. TIMETABLE MANAGEMENT WEB APPLICATION

At present, in the Higher Technical College of the Industrial and Telecommunications Engineering of the University of Cantabria, three higher degree courses of five academic years each and three technical degree courses of three years each are given. This means that there are up to thirty classes being given at any one time. Each student of these courses receives around 25 hours of class per week, meaning that, bearing in mind the multiplicity of optional subjects, it is necessary to assign classrooms and teachers for more than 900 class hours per week, which gives some idea of the complexity of the system.

The initial motive for carrying out the present project was to provide a support tool for devising class timetables and assigning the dates of exams for all the subjects of the various courses taught in the College. The intention was to facilitate the planning and at the same time to guarantee the integrity and consistency of the data.

As stated above, a further objective is that of providing teachers and students through the Web with the information they need about their activity at all times. Administrative staff and other interested persons can also benefit from the system.

3.1 Database design

First of all, an in-depth analysis of the real situation was performed for the design of the database. From this analysis, it was concluded that the timetables are organised per week (from Monday to Friday) and are grouped together according to degree course, year (or cycle, in the case of optional subjects) and semester. The timetables are also divided into boxes or cells which generally correspond to periods of one hour. Thus, each box corresponds to year of a degree course for one semester and is characterised by the day of the week and the start and finish time of the class. In each box, the subject, teacher and classroom is shown, as well as the subgroup of students into which the teaching of this subject is divided (see Figure 6).

Telecommunication engineer 5 ^o year, 1 st semester						2003-04
Time	Monday	Tuesday	Wednesday	Thursday	Friday	
15:30-16:30					OPTICAL COMMUNICATIONS Cobo, A. Room 07 Group 1	
16:30-17:30				OPTICAL COMMUNICATIONS Cobo, A. Room 07 Group 1	OPTICAL COMMUNICATIONS Cobo, A. Room 07 Group 1	
17:30-18:30	OPTICAL COMMUNICATIONS Cobo, A. Room 07 Group 1	ELECTRONIC INSTRUMENTATION FOR COMMUNICATIONS Drake, J. Room 07 Group 1	ELECTRONIC INSTRUMENTATION FOR COMMUNICATIONS Drake, J. Room 07 Group 1	ELECTRONIC INSTRUMENTATION FOR COMMUNICATIONS Drake, J. Room 07 Group 1	ELECTRONIC INSTRUMENTATION FOR COMMUNICATIONS Drake, J. Room 07 Group 1	
18:30-19:30		NETWORKS AND TELEMATIC SERVICES Zamarrillo, J. Room 07 Group 1		NETWORKS AND TELEMATIC SERVICES Zamarrillo, J. Room 07 Group 1	NETWORKS AND TELEMATIC SERVICES Zamarrillo, J. Room 07 Group 1	
19:30-20:30				NETWORKS AND TELEMATIC SERVICES Zamarrillo, A. Room 07 Group 1		

Figure 6. Traditional presentation of an afternoon timetable

3.2 Application Interface

In order to present an efficient and user-friendly environment, concealing the complexity of the database from the user, an interface was designed based on Web pages whose content is generated dynamically from the information stored in the database. Moreover, these Web pages are found in the database itself, so that the reliability of the system increases.

As can be appreciated in Figure 7, the page is divided into three areas. At the top is the heading, with the corporative image of the University of Cantabria and the name of the user who has connected. On the bottom left are the various menu options available at each moment and on the right the specific contents of each page.

In order to obtain this structure of the pages, 3 dynamic tags have been defined, one for each section, which allows the appearance of these to be easily modified, if necessary. A fourth dynamic tag is used for the presentation of the menu. With all of this, the code common to all the pages is:

```
<?startpag>
<?menubar_option OPTION=6 SUBOPTION=2>
<?halfpag>
(Page content)
<?endpag>
```

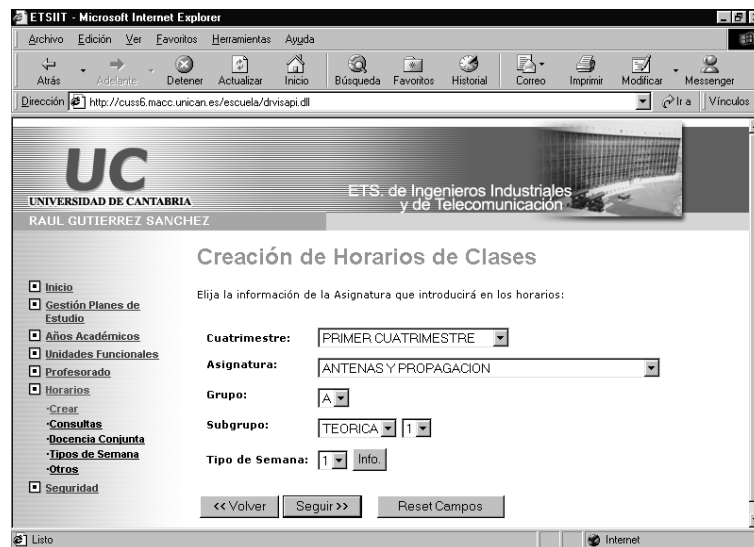


Figure 7. Example of management type page.

As for the contents of the pages, it may be said that there are two types: linking pages and management pages. The linking pages are pages aimed at storing option and suboption menus which will guide the user to a specific page. The management pages are pages aimed at performing operations directly on the database such as requesting, inserting, modifying or deleting information.

One element common to all of the management pages is the presence of buttons which link to the pages which make it possible to move from page to page. In the pages destined for requesting, there are buttons which link with the pages which allow this information to be modified, provided that the connected user has the necessary permissions.

It should be pointed out that, though it is true that the database, through its structure and triggers, will be the vehicle through which the validity of the information introduced will be guaranteed, at the same time, in order to make things easier for the user, all those pages in which information is introduced in forms have available JavaScript functions which make an initial control of the information introduced.

3.23 Software and hardware used

This section outlines all of the software and hardware elements required for the development and exploitation of the system.

3.3.1 Data Storage Software

The software selected for the storage of data was the Informix Universal Server for three reasons: one, the College had this already thanks to a collaboration agreement with the company; two, the documentary retrieval and Web access capacities required by the system were provided by this tool by means of specific datablades and three, it was a way of learning about the functioning of the capacities offered by ORDBMS.

3.2.2 Application Software

Two areas of the application software should be distinguished: first, that which is oriented towards the database design and that oriented towards the design, management and exploitation of Web pages.

For the database design Sybase's application *Data Architect* was used. This tool allows, among other things, the generation of the physical schema of tables from a conceptual schema, the management of triggers and stored procedures and the modification of the database using ODBC.

For the development and administration of the Web pages, the software *Data Director for Web* was used, a tool which Informix supplies with the WIO datablade (see Section 2.1).

For the visualisation of the pages created, the browser Internet Explorer 5.5 was required and as the Web server, the Internet Information Server.

3.2.3 Hardware

The database server and the web server were installed in a Pentium III at 500 Mhz, 528 Mb of RAM and 20 Gb of disc on O.S. WinNT Server.

The personal computer aimed at the development of the web application did not have great hardware performances. Pentium III computers were used with 300 Mhz and 128 Mb Ram with O.S. Windows 98.

The user devices only require a browser with a connection to Internet.

4. EXPERIENCE ACQUIRED

The Web application was designed and developed as an end-of-course project, by two students of Telecommunications Engineering. As the project developed, the advantages and disadvantages of the use of the selected software tools came to light.

In relation to the database design, working with a case tool such as Data Architect, which facilitates the conceptual design and its physical materialisation including the non-structural restrictions has proven highly beneficial, not only for the overall view it offers of the system but also for the dictionary it generates and reuses and for the time saved in modifications, as it generates the scripts required for them.

As for the Informix Universal Server Database Management System, this makes the task of integrating specific modules (datablades) in a database very easy. Moreover, the use of these follows a syntax along the lines of the SQL. However, it should be pointed out that a certain degree of difficulty has been encountered when migrating the database designed to another machine, as the datablades are registered according to the path in which they are installed. Moreover, Informix does not provide sufficient information about this.

In this particular case, the use of the WIO datablade offers two clear advantages: the pages are stored in the database and the dynamic code is embedded within the page itself, by means of specific tags which incorporate the SQL language. It also allows reusable code to be created using dynamic tags, the pages to be classified and permission for access to the pages to be controlled.

Moreover, the use of the datablade Excalibur, allows the full-text search of documents or texts stored in the database to be easily integrated. It also offers the possibility of creating different indexes for the same field in order to deal with the various search types.

The main disadvantage of the use of datablades is the fact that they are linked with one specific software company and are not at all flexible in this sense.

As regards the development of the application in the area of the interface and the process logic, it was decided to use Web technology as opposed to a traditional application developed in Visual C++ or Visual Basic. The advantages of this alternative are: the simplicity and agility in setting up the service, its easy maintenance, the low requirements of the client devices (only a browser is required) and the use of the application from any geographical point without the need for any specific installation. However, it should be pointed out that the applications prove to be rather slow, basically due to the fact that they are developed in interpreted programming languages and that these languages are still lacking certain utilities.

Finally, in the opinion of the users, the use of the application developed is easy and intuitive. However, it has up to present proven difficult to set it up due to the lack of administrative staff to perform the initial loading of all of the data of the system.

REFERENCES

Excalibur Text. Search DataBlade Module. User's Guide version 1.1. (1997). Part No. 000-3719.

Gutiérrez, R., 2001. Proyecto fin de carrera: Gestión Académica Virtual: Aplicación WEB para la Gestión de los Horarios de la E.T.S.I.I.T. de la Universidad de Cantabria.

Informix Software, Inc., 1997. An Overview of INFORMIX-Universal Server.

Informix Software, Inc., 1998. Web Integration Option. Data Director for Web.

- Nicolás, M., 2001. Proyecto fin de carrera: Gestión Académica Virtual: Aplicación WEB para la Gestión de Planes de Estudio y Personal docente de la E.T.S.I.I.T. de la Universidad de Cantabria.
- Zhen, H. L., 1999. Object-Relational Features in Informix Internet Foundation 2000: UDTs, UDRs, and User Defined Access Methods. *Informix Technical Features*. Volume 9, Issue 4, pp. 77
- Zorrilla, M.E. et al, 2003. Non-structured Data Management by means of Object Relational Database Management Systems. *Systems Analysis Modelling Simulation- Taylor & Francis Ltd*. Vol. 43, No. 9, pp 1173-1187