Design of an e-learning Environment for Teaching Databases and Information Systems

Carlo DELL’AQUILA, Ezio LEFONS and Filippo TANGORRA
Dipartimento di Informatica
Università di Bari
via Orabona 4, 70125 Bari
ITALY

Abstract: - We present a system that provides an e-learning environment in the database and information system fields. The system has been designed to support different teaching needs deriving not only from the computer science degree, but also from other University degrees that require database skills. For this purpose, it has been developed a repository of didactic modules on database and advanced database topics organized to support the different needs of potential students.

Key Words: - Educational software, Courseware, LCMS platforms, Database skills.

1 Introduction

Database management concepts are a common and necessary component of computer science education. In addition, the growth of database needs has recently received a significant impulse from the emerging applications and technology requirements [1-3].

With the advent of the World Wide Web and browser based computing, the user models for both the design of and access to database systems have changed significantly. In fact, it is well-known that the simple relational data model is insufficient to respond to these requirements. New data structures and data definition/manipulation languages have been introduced to permit the most complex data views. As an example, we refer to logic databases and object-oriented databases, or specialized databases for particular application areas (i.e., temporal/historic data, territorial/spatial data, multimedia data, and so on) [4].

Usually in computer science curricula, the traditional database contents have been extended with advanced database concepts. Frequently, as in our University context, the demand for database skills is not limited to undergraduate and graduate degree courses, but the formative offer covers also further requirements in specific advanced topics; viz., courses for industry education, qualification of the internal employees of the University institution, or for teacher’s Diploma. Each formative request must be adequately satisfied at the appropriate deepening level that corresponds to specific learning objectives.

The regular sequence of course lectures must be also covered by didactic materials that permit the students to reinforce and study concepts of lectures and to verify by opportune tests and exercises the course effectiveness. The choice of a suitable textbook plays an important role for completing the study with those details on which the lectures do not pay sufficient attention. Unfortunately, in this wide context of didactic proposal, it is difficult to find a text that covers the topics in the outline at an appropriate level for each specific need. Modern technologies offer a valid aid in arranging didactic materials for improving the learning process of students. The flexibility of e-learning tools and easy access to resources allow the teacher to adapt contents and didactic tools to the specific teaching needs, while maintaining the perception of the uniqueness of the didactic issues.

The aim of the implementation of the present learning environment is to provide teachers of a repository of lectures organized in didactic units, from which they can dynamically select a consistent sequence for specific teaching requirements on database field concepts. In a sense, these materials are an extension of the textbook, which has replaced, along generations, the need for each teacher to develop own class notes.

The system has been organized in different modules supporting the behaviour of the main actors; namely, the teacher and the student. It allows the teacher to manage theoretical contents and exercises, to obtain statistics about its use, to
evaluate the student learning process, and eventually to attend with suitable suggestions.

At the same time, it helps students to improve the quality of educational services with allowing a deep presentation of the course arguments and choice of personalized formative paths, useful to fill cultural gaps and/or to give all that needs to learn new concepts during the lectures in the classroom, to train by exercises on the learned concepts, and to self-evaluate the obtained progress.

2 Didactic needs of database contents

In this Section, we make a survey of the database topics of our educational efforts, focusing mainly the attention on the institutional courses of degree curricula and their co-ordination at the national level with other similar study courses.

The didactic needs of database management contents split into two different study levels of the Computer Science: the undergraduate level degree and the graduate level degree.

In such contexts, the organization of the corresponding courses follows some guidelines established from GRIN, or the National association for the Computer Science degrees, that specifies the minimum contents (syllabus). The courses that are currently offered at several major universities across the Italy, with certification from GRIN, provide the following contents:

- **database contents for first level degree in Computer Science**
  - Logical models
  - Conceptual design
  - Logical design
  - Query database languages
  - Database management systems

- **database contents for the second level degree in Computer Science**
  - Programming language for database
  - Normalization
  - Physical organization and query management
  - Transactions and concurrency
  - Advanced databases

According to these proposals, the core topics of database are arranged in two distinct fundamental courses for students of (a) undergraduate and (b) graduate computer science curricula, respectively. However, by choosing individual programmes of study, students can integrate the typical courses on introductory database concepts and acquire further abilities with specific courses expanding the matter. They achieve the twofold purpose: on one side, to

![Core undergraduate database course]

- Relational data model
- Entity-relationship modeling
- Logical design
- SQL language
- Functional dependencies
- Normalization
- Database management system Lab

![Advanced undergraduate database course]

- Database system analysis and design
- Physical organization Indexing
- Object oriented model
- Active databases
- Multimedia database systems
- SQL 3

(a) Undergraduate course topics

![Core graduate database course]

- Reading relational databases
- Object oriented database
- Query optimization
- Transaction processing
- Concurrency control
- Recovery
- Database security

![Advanced graduate database course]

- Advanced Data models
- Deductive and Knowledge
- Distributed databases
- Interoperability
- Web databases
- Advanced query processing
- Advanced transaction processing

![Information system course]

- Decision support systems
- Data warehouses
- Multidimensional models
- MOLAP
- ROLAP
- OLAP operations
- Data mining
- Information system security

(b) Graduate course topics

Fig. 1. Overview of database education needs.
deal deeper the arguments, and, on the other side, to treat advanced aspects that cover the recent database developments.

Taking these considerations into account, Figure 1 provides an overview of the distribution of database contents into several courses at different degree levels and at different levels of deepening. Figure 1(a) provides a summary of the topics covered in the undergraduate level database courses, while Figure 1(b) shows the topics that are offered at the graduate level. Courses at both undergraduate and graduate levels pursue the same aim distinguishing the preparation objectives for two different student targets. The first objective is devoted to furnish the common core of database knowledge and skills that must be acquired for all students of the computer science degree. Mainly, it regards the relational data model and language for undergraduate course and the object-oriented approach and the physical aspects the graduate course. These fundamental courses implement the recommendations of the GRIN association previously reported. The second objective relates to the possibility to offer further specific courses individually chosen by the students that intend to gain very deep knowledge and skills in database field. In particular, at the undergraduate level devoted to a professional preparation, the further database course reinforces capabilities in planning, constructing, and managing information in database systems, also by developing real study cases. At the graduate level, one of the offered formative profiles is oriented to prepare professionals who want to plan, to analyze, to design, to construct, to maintain, and to manage distributed applications, information systems, and databases, with particular attention to the typical aspects concerning the role of the system administrator in the matter of data security, optimization, information retrieval, and web databases. At the same time, the specific curriculum deals with advanced database concepts such as the evolution of databases, data warehouses, data mining, logic data bases, and databases with constraints.

3 The Learning System
The courses are supported by the e-learning environment using ATutor, an open-source Learning Component Management System (LCMS), developed at the University of Toronto [5].

LCMS platforms have acquired growing popularity in the e-learning field and they constitute the evolution of the Learning Management Systems (LMS). In addition to the usual functionalities of LMS systems to manage e-learning courses, the LCMSs emphasize the organization of the e-learning courses by a content point of view, as composed of basic reusable and independent learning units, called Learning Objects (LO).

We have organized the repository of the learning objects concerning the database learning units for composing different types of courses based on specific requirements.

![Fig. 2. LCMS structure.](image)

Figure 2 shows the typical structure of a LCMS environment with particular reference to the support given to subjects which can access to the system. Students can navigate through contents of the course following personalized learning paths, in order to acquire the opportune theoretical knowledge. They can also compile exercises and tests for evaluating the obtained progress in the learning process. The system is able to store the student's learning path (visited units, results of exercises and tests), so that it is possible to recall the learning process starting from the breakpoint. Moreover, the system also provides functionalities for the student-teacher interaction. Teachers create, update, and manage the didactic unit contents, tests, and exercises. They can analyse the results of student exercises and tests and control the study path of students and statistical parameters on the accessed didactic units. Finally, the system considers also the administrator profile, that is, a skilled technical user providing functionalities for the management of aspects such as, for example, the authorizing process for the access to the course, the coordinating and monitoring of its use, and so on.

The choice is fallen over this system after a careful examination of the most popular available LCMS open-source platforms. In particular, we
have analyzed: ATutor, Spaghetti Learning, Moodle, Ilias and Claroline LCMS platforms [5-9].

We examined these systems, taking into account the following characteristics: 1) specific interface for the course management, 2) easy building and navigation among the courses, 3) import and export of learning contents according to the IMS/SCORM standard, 4) availability of the integration of e-mail functions to contact all students of a given course, 5) management of forum and chat functions, 6) user access tracing and storing, 7) availability of statistics on course accesses and test results, 8) context help, 9) glossary, and 10) server multiplatform support (Windows, Linux, and Unix).

The result of the comparison is deeply discussed in [1].

With the present release of the system, we have updated the platform with the new version of ATutor LCMS. The relevant innovations introduced regard the following aspects:

Accessibility

The new system maintains the same easy selection way for course access, and good login and authorization mechanisms. The predefined profiles are the administrator, the teacher, the student, and the guest. In addition, the new version of the system permits to define administrator profiles also for restricted functionalities, for example: the administrator only for the students and/or only for the single courses and/or only for the forum/chat and so on. This new possibility is very useful in the case of very large and complex implementations, as in our case, where the learning needs to manage different courses and degree levels. Moreover, the module Acollab integrates group management functionalities that allow groups of users for accessing to collaborative tools.

Forum/Chat/e-mail

Groups of students can manage their blog, access to the blog of other groups, and send private or public messages to the blog. A collaborative environment (the Marretech module) allows the student groups to work via web conferencing (up to five users at the same time in the free version), including the video synchronized with the voice and an interactive blackboard.

Traceability

In addition to the usual requirements of the user traceability (that allows both the student to take in account the previous accesses to the course and the teacher to examine the student tracing with the possibility to summarize with statistics and diagrams the more visited contents), the system maintains the traceability with the versioning of the shared files that contains the common work documents of the collaborative group uploaded and downloaded in the workspace.

The ATutor system has emerged as the tool that better meets project requirements. It has been written using PHP, Apache and MySQL, which are useful standards to match our system requirements.

4 An example: the information system course

For each course shown in Figure 1, the didactic materials have been organized according to three different types of navigation (Figure 3). In the lectures section, the student consults the didactic unit supporting theoretical lectures. In the synthetic section, the system shows the contents of the same didactic units in a synthetic form (often are reported the same slides used by the teacher, in his/her lectures). Links from the synthetic level to the corresponding argument in the detailed level assure the deepening of unclear contents. Finally, in the exercises section, the student tests the acquired knowledge of contents of didactic units for a self evaluation of the obtained progress in the learning process. In case of negative results of the tests, the system suggests to the student the specific arguments to be re-examined. In fact, tests are related to the arguments at the end of each didactic unit.

The course contents are organized in didactic units and they are activated and shown progress-
progressively, following the sequence of teacher’s lectures, to avoid the complete substitution of lectures with a virtual interaction with the on-line course.

For example, the information system course presents contents (in Italian) corresponding to the didactic units of the course. In addition, the basic concepts of theoretical lectures can be reinforced with exercises that allow the student to be familiar with the application level [10].


The didactic material has been organized according to the course structure examined in the previous Section and following the methodological choice to apply the same learning strategy between the lectures in classroom and the course. So, the student can retrieve in the course the same guidelines of the presentations used by the teacher in the lessons. In addition, the navigation among the items of didactic units is organized by means of opportune links to other items or didactic units that are related to current items or exercises.

The course is accessible by using a general index and it is presented to the student in a suggested logical order to avoid the getting lost in hyperspace [12].

The course is composed of (eight) didactic units that illustrate respectively the following fundamental concepts: Introduction, Information System Life Cicle, Information System types, Decision Support Systems, Data Warehousing, Data Mining, and Information System Security. Each didactic unit is equipped with exercises. As an example, the organization of the didactic units concerning Data Warehousing and Mining is shown in Figure 4. The didactic unit is an overview of data warehousing and data mining which explains critical issues in planning, design, deployment and ongoing maintenance of data warehousing. It describes techniques for extraction of data from sources, data transformation, data staging, data warehouse architecture and infrastructure and various methods for delivery. Topics also include an overview of On-Line Analytical Processing, Knowledge Discovery Database Process Model, Regression Analysis as they relate to data warehousing.

In Figure 4, the structure of the didactic units of the database course is reported conforming with contents of the general index. The dependencies among units do not follow an exact process. In fact, the student can customize the access to lesson arguments. The system only suggests the default path for a better comprehension of the course contents.

![Fig. 4. The structure of the Data Warehousing didactic unit.](image)

The system has the capability to store the student’s learning path (the visited units, the results of exercises, and tests), so that it is possible to recall the learning process starting from the break point. It uses a data base of general information about students, linked with information about the learning progress of each student (tracks on already read didactical units, exercise results, and so on…) so as to permit a self-evaluation of the obtained learning progress.

Figure 5 shows an example of the interaction with the Information System course.

5 Conclusions

This paper has presented our efforts in the development of a repository of didactic material for database and advanced database topics. The wide demand for database skills at different level of deepening has suggested to organize modular material for supporting the different needs of potential students. The project has been implemented on the freeware and multiplatform Learning Component Management System developed at the University of Toronto. The didactic material collected in online courses for supporting the corresponding theoretical lectures has been profitably used by students in computer science at the University of Bari.
The course evaluation has been based on Nielsen’s heuristics, which has inspired the questions presented in the module to the user to improve the course usability. The heuristics list has been determined identifying the more frequent problems that can arise when accessing to web sites [13].

Actually, the composition of didactic material results a bit difficult and requires the careful support of teachers. The future work is devoted to the automatic assembly of multimedia didactic material using appropriate environment [14].

References:
[8] www.ilitas.uni-koeln.de