# AulaNet: An Object-Oriented Environment for Web-based Education

**Sérgio Crespo, Marcus Felipe M. C. da Fontoura, and Carlos José P. de Lucena**

Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro
Rua Marquês de São Vicente, 225, 22453-900 Rio de Janeiro, Brazil
e-mail: [crespo, mafe, lucena]@inf.puc-rio.br

PUC-RioInf.MCC14/98 Abril, 1998

ABSTRACT

As yet there are no specialists in the application of information technology to education and training. However, this is one of the fastest growing areas on the Internet, due to the recent perception of the enormous potential for the use of Web resources for this purpose. This potentiality has attracted the attention of researchers in industry and the academic world, which are currently developing various models and products for Web-based education and training. In this paper we present AulaNet™, an environment for the creation and maintenance of Web-based courses designed for the layman, highlighting its architecture and design model. We also compare AulaNet™ to other Web-based education (WBE) environments.

KEY WORDS: Web-based education, learningware, software, information technology, object-oriented frameworks, Web-based educational environments.

RESUMO

Ninguém ainda é especialista na aplicação das tecnologias da informação à educação e ao treinamento. No entanto, o recente reconhecimento do potencial do uso dos recursos da WWW para educação e treinamento faz com que esta seja uma das áreas de aplicação que mais cresce na Internet. Este grande potencial tem atraído a atenção de grupos de pesquisa e desenvolvimento da universidade e da indústria que se ocupam atualmente em desenvolver modelos e produtos para a educação e treinamento, basedos na Web. Neste trabalho apresentamos AulaNet™ um ambiente para a criação e manutenção de cursos por não especialistas, enfatizando sua arquitetura e modelo de design. Uma comparação do AulaNet™ com outros ambientes para educação baseada na Web também é apresentada.

PALAVRAS CHAVES: educação baseada na Web, aprendizado, software, tecnologia da informação, frameworks orientados a objetos, ambientes educacionais baseados na Web.

# 1. INTRODUCTION

From the technological point of view, the first motivating factor that led the LES[1] to teaching/learning based on the Web (Web-based Education [7]) is linked to the great transformations that Web technology has brought to the process of software development and to the actual concept of software. More and more often developing software means developing a Web-based solution. Therefore, it is becoming essential that research in Software Engineering investigates models, languages and tools for the specification and development of Web-based software.

Studying software technology independently of application domains, for which one could, for example, develop real case studies, often leads to very general models, languages and tools that are inappropriate for the majority of practical situations. Whereas the study of technologies in the context of a specific domain does not restrict the discovery of methods and tools that could be used for general purpose. From the methodological point of view the LES chose to dominate Web-based solutions focusing on the use of the Web in the area of education.

Obviously, education was a relevant choice for us with our daily experience of a university environment, which naturally complemented our technological motivation described above. The possibility of using Web technology in education only began to be seriously explored at the end of 1996 (when the term "Web-based Education" began to become more widely known). Since then, various projects, similar to the one that will be discussed in this article, began to be developed, without anyone in the academic or technological communities being able to call himself a specialist on the subject.

Another reason why the LES was motivated to research Web-based education and training is that this area offers vast potential for universities to cooperate with industry and society in general. Furthermore, the clear emergence of an Information Society, with global characteristics, has enormously stimulated industry's interest in education and training as a way of ensuring competitiveness.

There is great hope that the actual information technology leading to the construction of the Information Society (Internet, Intranets, WWW etc.) will bring solutions for the problems of offering education and training on a wide scale and at a more accessible cost.

International Data Corporation [7] believes that the adoption of education and training based on the Web, either through corporate Intranets or through the Internet, barely started in the second semester of 1997 and will explode into a 2 billion dollar market in the year 2000. The main driving factor for Web-based training is the necessity to find methods of bringing training directly to the desktop in a continuous just-in-time way.

The convergence of the traditional media to a digital format is drawing computers closer to the telephone and television, revealing practical results that are not fully understood by the specialists in the areas involved.

The pedagogic resources available for educators to give lectures in person are countless and include for example: books, printed material and CDs, blackboard and chalk, slides, personal conversations with the instructor and assistants, discussion groups, illustrative films and videos, laboratories, and excursions. The technological resources available for distance education until the recent past included printed material radio, telephone, TV as well as

[1] Software Engineering Laboratory, of the Computer Science Department at PUC-Rio (http://www.les.inf.puc-rio.br).

videoconferences for interaction in real time. There is a great deal of optimism relating to education and training based on the Web. This comes from the verification that it is easy to imagine how the Web could simulate all resources used in lectures with the instructor present at acceptable transmission speeds, which is already the case with Intranets.
Education and training based on the Web uses the WWW as a means of publishing didactic material, applying tutorials, applying exams and tests and communicating with students. The Web can also be used as a vehicle of communication for the presentation of distance lectures (multimedia conferences).
This paper is divided as follows. Section 2 describes the AulaNet™ conceptual model for applying technology to education; its architecture and main features are presented in Section 3; Section 4 illustrates its design model through OMT diagrams; Section 5 describes other approaches for Web-based education (WBE); the conclusions and future work are presented in Section 6.

## 2. A CONCEPTUAL FRAMEWORK FOR APPLYING TECHNOLOGY TO EDUCATION

The environment provided by AulaNet™ aims to reflect a conceptual framework for the implementation of technological innovations in education. This is based on three arguments:

- Adoption precedes change. So that any intended change can take place the innovation must be adopted by the students and instructors first;
- Creation is re-creation. The process of implementing a new innovation is, essentially, a process of re-creation in which instructors and students reinterpret the innovation in their own terms. In this way the production of a new innovation frequently reflects a set of solutions of compromise between the new and old way of doing things.
- Apprenticeship is the evolution of knowledge. Human beings are active, but fallible, creators of knowledge. The knowledge created must be improved by the criticism of their peers.

Therefore, the objectives of AulaNet™ are: (a) to promote the adoption of the Web as an educational environment; (b) to contribute to pedagogic changes, giving support to re-creation and (c) to encourage the evolution of knowledge (as much for the students as well as for the instructors).
To implement these objectives the following resources were foreseen for the AulaNet™ environment:

(i) **Total integration**

AulaNet™ will have a set of tools for students and instructors designed for the various stages of the learning process. An integrated environment allows students and instructors to interact in multiple dimensions, to analyze the same data with different tools and to achieve multiple objectives in the context of the same environment.

(ii) **Global collaboration**

AulaNet™ was created in order to give assistance to the learning process. An integrated environment allows students and instructors to interact in multiple dimensions, to analyze the same data with different tools and to attain multiple objectives in the cooperative apprenticeship context of its students. Apart from

collaborating on pre-established projects the students can also start new projects and meet new partners.

(iii) **Transparent technology**

In order to facilitate its adoption, AulaNet™ is being designed to function in the most transparent way possible for the users. For this purpose it has a Web browser as an interface. Thus, technically, the only software that a user needs is a browser. Secondly, the independent nature of the Web platform means that it is not necessary for the user to change platform. Thirdly, AulaNet™ will allow the user to re-use material that he is already familiar with. For example, the user could make a course description typing directly onto a form or uploading from a file of a text editor. Finally, AulaNet™ has an internal structure and mechanisms that allow students and instructors to manage their data and to make it available on a Web server. The users do not need to know anything about programming or coding and transferring Web documents (eg: HTML, CGI, FTP etc.)

(iv) **Flexibility**

In order to promote its adoption and re-creation, AulaNet™ is being designed to have maximum flexibility in terms of pedagogic beliefs. AulaNet™ foresees resources that support both old and new forms of teaching and learning. On the one hand it gives support to the use of slides and conventional printed material, and on the other hand it uses mechanisms like News and videoconferences.

(v) **Construction facility and improving knowledge**

To help students see the evolution of the growth of knowledge, AulaNet™ keeps a log of the various possibilities of interaction. AulaNet™ also has the facility for students and instructor to continue discussing the lecture through a News server, after it has been presented by the instructor.

(vi) **Common technology**

The use of available technologies reduces costs and encourages adoption reducing the task of training the users. AulaNet™ has a set of existing tools interfaced by the Web.

(vii) **Evolution of knowledge and ability**

Instead of offering a fixed set of instructional material, AulaNet™ creates an environment in which courses can be in a state of permanent evolution. For example, the students' work from one course will be available as subsidies for students on the next course.

(viii) **Access to course files**

The content of the courses created with AulaNet™ evolve with time and are stored in a database. All available information can be easily retrieved at any given moment.

## 3. AulaNet™ DESCRIPTION

AulaNet™ [12] is a software environment based on the Web developed at the Software Engineering Laboratory, LES, for creating and assisting distance courses.

AulaNet™ was conceived from the experience gained in three courses during the second semester of 1997 [20, 9, 21], and is based on the following basic premises:
1. The courses created must posses great capacity of interactiveness, in order to encourage intense participation by the student in the learning process ("learningware") [8].
2. The author of the course does not need to be a specialist on the Internet.
3. The resources offered by the creation of the courses must correspond to those available in a conventional classroom, plus others normally available in a Web environment.
4. It must be possible to re-use the contents already existing in digital media, through importing archives, for example.
Attending to the above premises led to the formulation of the concept of an **Apprenticeship Development Process** (ADP), which demands that the author previously specifies the teaching resources he will use while developing the course, as can be seen in the figure below.

*Figure 1. AulaNet™ Authoring Site: Selection of Teaching Resources*

With AulaNet™ it will be quite easy to create distance courses through the Internet, that have a high level of interactiveness and intense student participation, without the author needing to be an expert in Web environments. Furthermore, the course created will use contents that already exist that have been recorded on a digital medium.
At the present moment there are about 30 courses being developed by professors from may different areas at PUC-Rio and in other Brazilian universities (http://les.inf.puc-rio.br/aulanet).

### 3.1 Actors

AulaNet™ considers that the following actors are involved in the authorship/apprenticeship process.

1. The Author: is the creator of the course, who participates from the initial description until the content is entered. He may or may not be the person responsible for applying the course.

2. The Student: is the final user of the course, representing the target-public, for whom the final product obtained by the use of AulaNet™ is intended.
3. The Instructor: is the one responsible for applying the course (can also be the author). He may or may not have the help of monitors, who deals with the practical aspects of the course and helps in assessing students

### 3.2 Resources

With reference to the creation of the course, AulaNet™ offers the author the possibility of selecting various resources that will be used by him or by the other actors in the final assembly of the course and that will be converted into services, afterwards. The resources in AulaNet™ come under four group headings:

1. Teaching Resources: correspond to the pedagogic tools that will be used during the application of the course and must be previously selected by the author.
2. Assessment Resources: correspond to the formats that will be used by the instructor to make an assessment of how well the students benefited from the course.
3. Administrative Resources: are necessary for establishing operational communication between students and the institute responsible for the course.
4. Fixed Resources: are resources used in any type of course, that support the basic operation of AulaNet™.

The resources selected by the author are converted into services supported by AulaNet™.

In order to provide all the necessary services, AulaNet™ has a set of software tools some of which, like freeware, are available on the Internet, others have been developed by the team at the LES.

### 3.3 Architecture

The architecture of AulaNet™ is based on the Web, where the environment interface is being developed, and CGI (Common Gateway Interface), which implements all the features of the server. Figure 2 is a view of this architecture in layers.

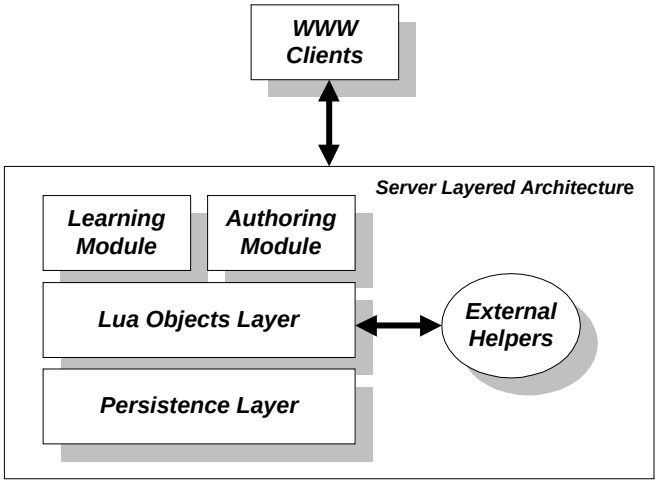


*Figure 2. AulaNet™ Layered Architecture*

All the objects manipulated by the environment, like for example courses, students, institutes and departments, are stored in a relational database, that is responsible for the persistence of these objects. The CGI interface between the database and the application is done by a Lua [6] objects layer. This layer is responsible for the manipulation of all information about the

environment and for the integration of AulaNet™ with external helpers, used to perform some of its services like discussion groups (list server) and transference of archives via the Web (file upload).
The Lua objects layer is also responsible for the definition of the application interface of the AulaNet™ services. It works like a facade, as defined in the Facade Design Pattern [4]. The mapping of the AulaNet™ objects and the relational persistence system is also carried out at this layer. AulaNet™ authoring and learning modules were implemented in CGILua [6]. Their respective interfaces are responsible for validating data on the client's side using JavaScript code. In this way the processing of AulaNet™ is distributed, as the clients are responsible for validating the input data while the servers are responsible for the persistence, manipulation of information and navigation control.

## 4. AulaNet™ DESIGN MODEL

We present the AulaNet™ design model through OMT [17] diagrams, which describe the object-oriented class structure for the Lua objects layer.
AulaNet™ allows several institutions to use simultaneously the same environment. Each institution may have several departments. The courses are related to institutions and may be either composite or single. Courses are also classified by a degree of difficulty (course levels).
Each course has actors and groups assigned to it. Groups are composed of actors and other groups. Each actor may either be a student, an instructor, or an author. The students can also be assigned to a student level.
A course consists of a selection of resources. Teaching, evaluation, and administrative resources may be incorporated by the author to the course.
Each resource may be mapped into one or more Internet services. As an example, we can use a teaching resource, which is a "Lesson Text", implemented by Internet services for text file uploads or just through URL references. Each service can also use one or more external tools, such as chat, CU-SeeMe, e-mail, and list servers. The whole design structure is presented in Figure 3.

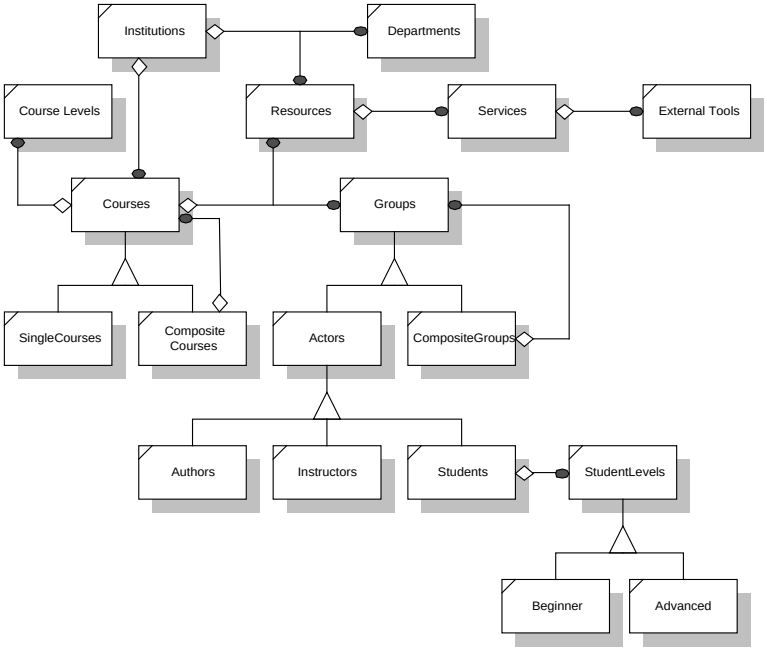


*Figure 3. AulaNet™ OMT Class Diagram*

In AulaNet™ the final user can modify the visual representation (Interface) for each one of its entities as shown in Figure 4.

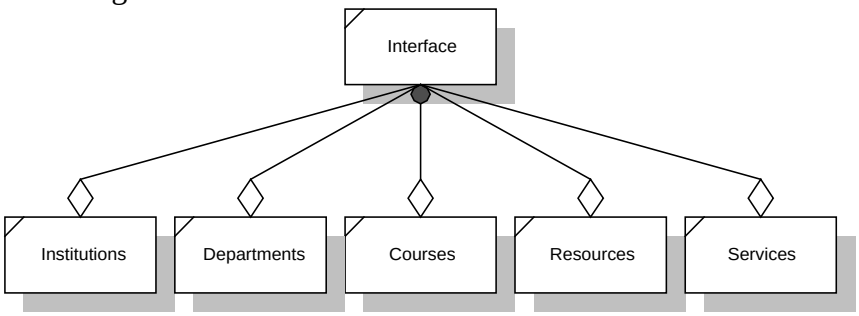


*Figure 4. AulaNet™ Interface Class Structure*

Each WBE environment, such as AulaNet™, is composed of two sites: a learning site and an authoring site. The learning site is used by the students to attend a specific course, while the authoring site is used by the authors to create and maintain the courses. In AulaNet™, the structure of both sites are defined here as hot-spots or flexible-points [15, 18], which are implemented by the classes Authoring Site and Learning Site. In this way, the AulaNet™ environment can be seen as an object-oriented framework [10, 16], where the final user can modify the sites interface and navigational structures at run-time.

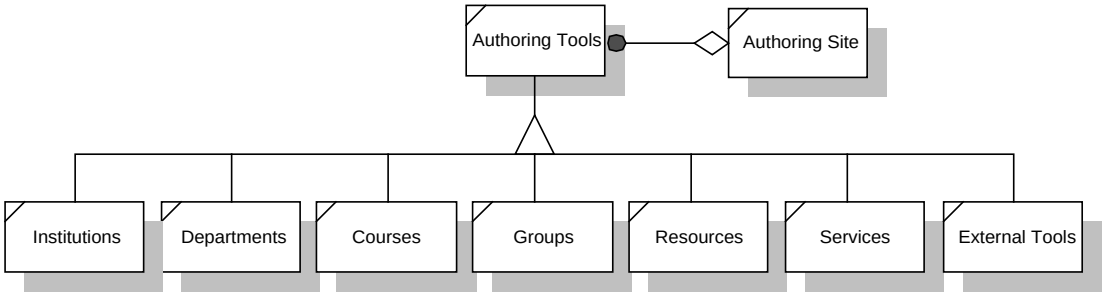


*Figure 5. AulaNet™ Authoring Site*

Figures 5 and 6 show the OMT models for the AulaNet™ authoring and learning sites, respectively. Notice that the learning site has to cope with multiple language requirements, since a single institution may want to have their courses published in several languages.

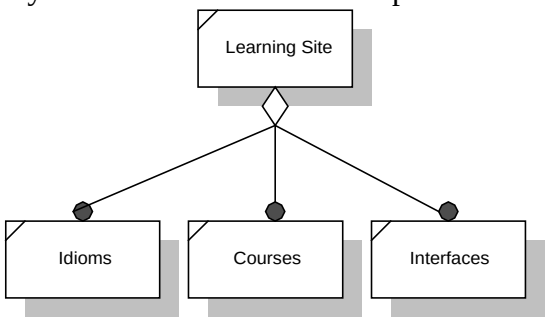


*Figure 6. AulaNet™ LearningSite*

# 5. SOME EXISTING MODELS FOR WBE

This section describes other Web-based education (WBE) environment through OMT design diagrams. Two important aspects must be considered when analyzing the following models. First, except for LiveBOOKs, we do not know the exact original object-model of the analyzed environments. The models presented here have been specified by studying and using these environments. Second, when the models are similar we just refer to the figure that describes it in order to avoid presenting similar models twice.

## 5.1 LiveBOOKs

The LiveBOOKs [2] distributed learning/authoring environment is a computer-based teaching/learning and authoring system that has supports learning and authoring activities. It is designed for easy of use; the environment is easy for the student to use and also easy for the author of teaching materials to create those materials and include them in a specific LiveBOOK environment. We now describe the LiveBOOK class structure.

A LiveBOOK class consists of an aggregation of a LiveBOOK content class and this simple structure is illustrated in Figure 7. The LiveBOOK class has methods to support browsing, updating, generation of the Table of Contents (TOC), and providing the current location of the browsing operation in the LiveBOOK. This latter method is used by the LiveBOOK control class to determine which controls should be presented at any point in the LiveBOOK. In other words the object model supports context-sensitivity. The TOC method will be used by the user interface to display the Table of Contents when the LiveBOOK is displayed on the screen.

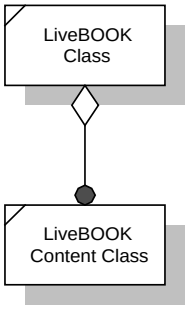


*Figure 7. An OMT description of a LiveBOOK class*

A LiveBOOK content class illustrated in Figure 8 is an aggregation of one or more LiveBOOK chapter classes.

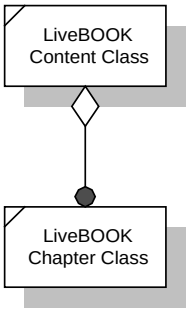


*Figure 8. LiveBOOK Content Class Structure*

A LiveBOOK Chapter class shown in Figure 9 is an aggregation of zero or more LiveBOOK Fragment classes and zero or more LiveBOOK Component classes. A LiveBOOK Component class is an aggregation of zero or more LiveBOOK Fragment classes and zero or more LiveBOOK Section classes. This definition is recursive in that a LiveBOOK Section class is an aggregation of a LiveBOOK Component class. This relationship is also shown in a LiveBOOK Fragment class illustrated in Figure 10, where the different types of fragment classes inherit from the fragment base class. The fragment base class is an abstract class because the methods on the class depend on its specific type. For example, the display method would depend on the fragment being displayed on the screen. A fragment can consist of text, a binary large object (blob), an image map, a hyperlink source or hyperlink destination, and a hyperlink reference class. A blob is usually a graphic, a picture, or an audio or video clip. An image map is a graphic or picture with a map to allow accompanying hyperlinks. A hyperlink reference class activates an external program.

The user interface of the LiveBOOK is important because it controls the variability of the presentation and in fact the format and style of the educational material. There are many possible user interfaces. This LiveBOOK user interface consists of three windows: a contents Window, a Table of Contents (TOC) Window, and a Control Window. The Contents Window displays the contents of a Subsection and allows the user to interact with the individual LiveBOOK fragments in that subsection. The Control Window represents the various controls that are accessible when a specific Contents Window is displayed. In other words the controls are context sensitive.

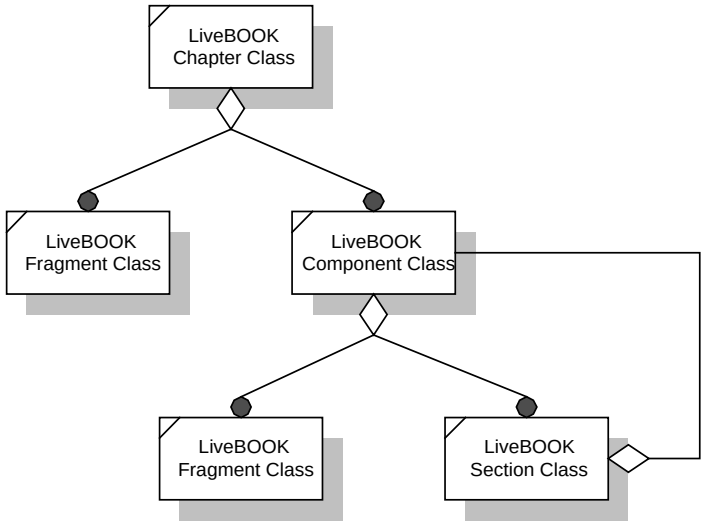


*Figure 9. LiveBOOKs' Chapter Class OMT Description*

This structure presented for LiveBOOKs represents the implementation of the LiveBOOKs service, and every course developed in LiveBOOK has this service as default.

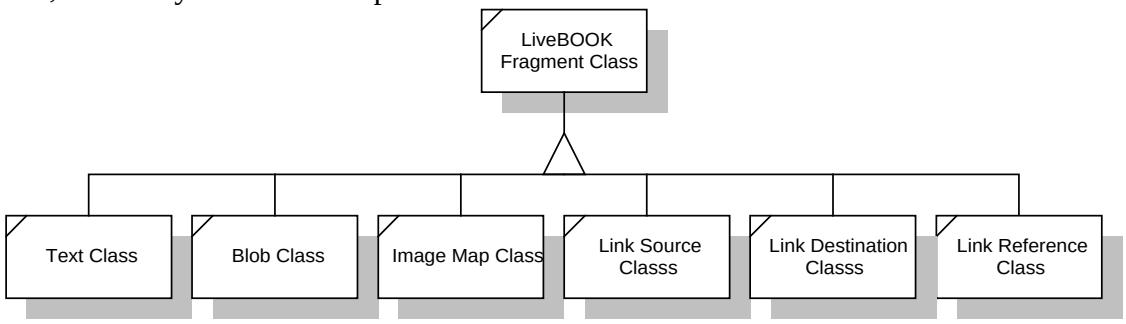


*Figure 10. An OMT Description of a LiveBOOK Fragment Class*

### 5.2 Web Course in a Box

Web Course in a Box (WCB) [23] is a course creation and management tool for Web-based or Web-assisted delivery of instruction. It enables creation of basic course pages such as syllabus, class schedule, personal home pages, as well as interactive Web functions such as discussion forums and self-correcting exercises. Authoring is done using a Web browser and does not require any knowledge of HTML or other technical skills.

The WCB class structure (Figure 11) is very similar to the one presented in Figure 3. The main difference is that WCB does not use the concept of resource: the Internet services are directly selected by the author. Note that unlike AulaNet™, WCB does not support the concepts of institutions, departments, composite courses, and groups.

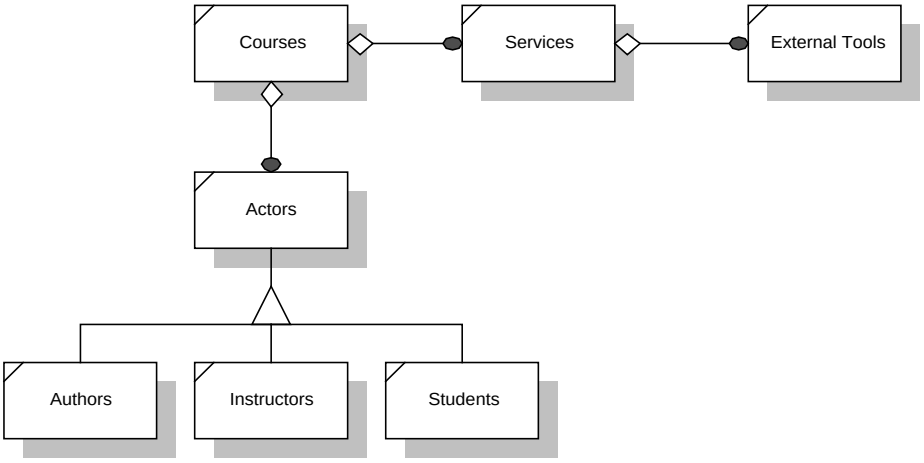


*Figure 11. WCB OMT Class Diagram*

In relation to visual representation, WCB allows changes only to the course appearance. This structure is presented in Figure 12.

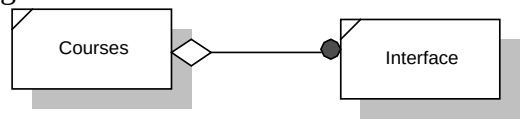


*Figure 12. WCB Interface Class Structure*

Both the authoring and learning sites are handled here in the same way: as hot-spots that are implemented by the Authoring Site and Learning Site classes. Note that the multiple language requirements are not supported by WCB.

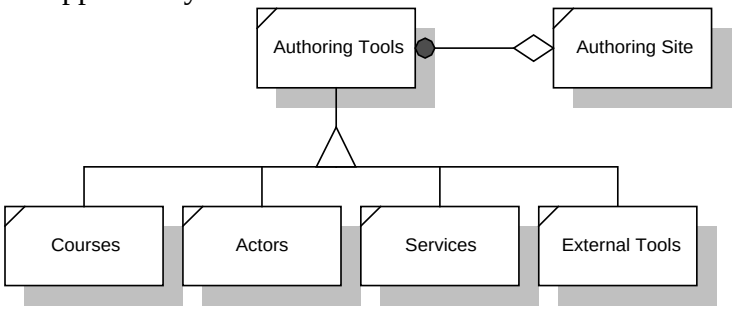


*Figure 13. WBC Authoring Site*

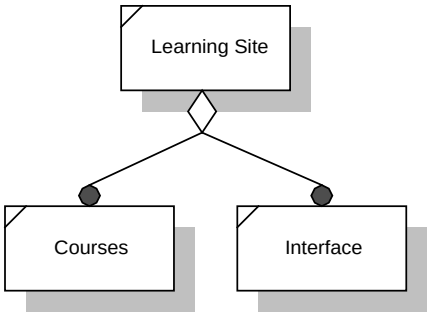


*Figure 14. WCB Learning Site*

### 5.3 WebCT

WebCT [22] is a tool that facilitates the creation of sophisticated WBE environments. It requires minimal technical expertise on the part of the developer of the educational material and on the part of the student.

The course developer or designer provides the content of a course. Interactivity, structure, and educational tools are provided by WebCT. WebCT, as well as allowing the incorporation of a set of educational tools in a course, also allows the manipulation of layout and look of the course. It was developed at University of British Columbia to produce various courses to that university in a easy way [5].

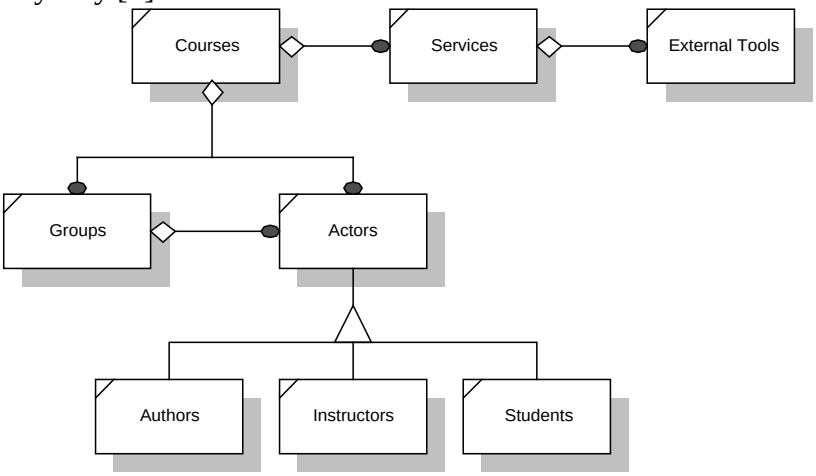


*Figure 15. WebCT Class Structure*

The WebCT (Figure 15) class structure is very similar to WCB (Figure 11) but it also supports the concept of groups. In relation to visual representation (Interface), WebCT allow changes not only in the course appearance, but also in the services and external tools (Figure 16). The authoring and learning site also have the same structure presented in WCB (Figures 13 and 14).

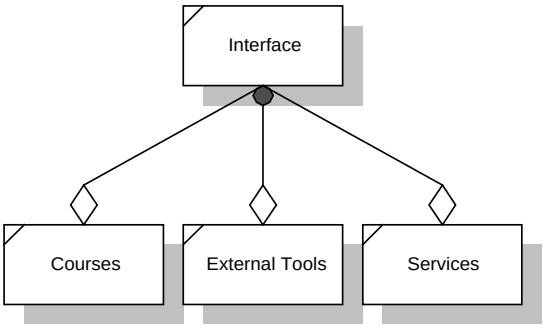


*Figure 16. WebCT Interface Class Structure*

### 5.4 LearningSpace and Virtual-U

These two environments are grouped together in this Section because they have similar class structures. Lotus Education and IBM are responsible for the research and development of Lotus LearningSpace [11], an educational technology with supporting services for distance education.

The LearningSpace and Virtual-U [19] class structures are essentially the same as the one presented in Figure 13. However, they additionally introduce the concepts of documents and

tasks (Figure 17). The services are based on documents, and each document may have various tasks assigned to it. In LearningSpace documents and tasks can be classified into categories (Figure 18).

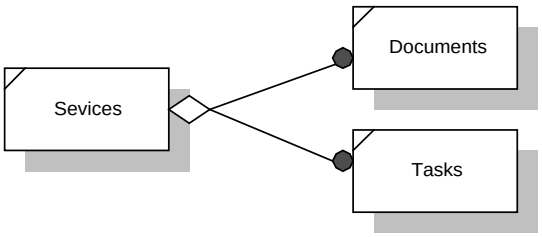


*Figure 17. LearningSpace and Virtual-U Services Class Structure*

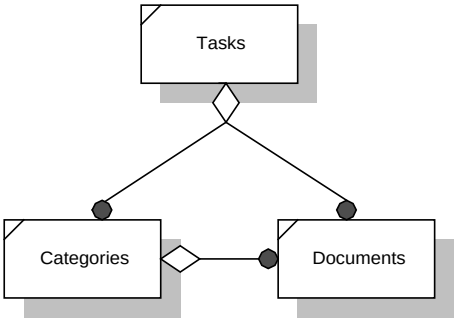


*Figure 18. LearningSpace Tasks, Categories, and Documents*

## 6. CONCLUSION AND FUTURE WORK

The future of the AulaNet™ environment, as well as the other systems mentioned in the previous section will depend mainly on their acceptance by a wide community of users. It is very probable that interaction between the developing team and users will lead to major modifications, for example, in the system interface.

Judging from experiments conducted at the same time as AulaNet™ was being developed, the authors believe that, the use of the Web as a complementary or exclusive form (eg. “just-in-time” training) of offering courses substantially enriches the current methods of applying technology to education.

Spider, a visual language and environment [14], was also developed in the LES, the laboratory where AulaNet™ was developed. In Spider, the logic as well as the interface of a system can be expressed graphically as long as an architecture of classes is developed for a specific application area. There is already an architecture for Web-based applications available [13] that will be complemented by an architecture for Web-based education [3]. We anticipate that in the near future the evolution of AulaNet™ will take place through this strategy.

Another related work we are developing in parallel is the design and implementation of a new WBE environment based on the experience learned from AulaNet™ and LiveBOOKs. The complete description of this new environment is presented in [1].

## REFERENCES

1. P. Alencar, D. Cowan, S. Crespo, M. F. Fontoura, and C. J. Lucena, “OwlNet: An Object-Oriented Environment for WBE”, MCC/98, Monografias em Ciência da Computação, Departamento de Informática, PUC-Rio, 1998.

2. D. Cowan, “An Object-Oriented Framework for LiveBOOKs”, Technical Report, University of Waterloo, Waterloo, Canada, 1998.
3. D. Cowan, S. Crespo, M. F. Fontoura, C. J. Lucena, and L. M. Moura, “An Architecture for the Visual Programming Development of Web-Based Education Environments”, MCC/98, Monografias em Ciência da Computação, Departamento de Informática, PUC-Rio, 1998 (also submitted to SBES’98).
4. E. Gamma, R. Helm, R. E. Johnson, and J. Vlissides, “Design Patterns, Elements of Reusable Object-Oriented Software”, Addison-Wesley, 1995.
5. M. Goldeberg and S. Salari, “World Wide Web-Course Tool: An Environment for Building World Wide Web-based Courses”, Computer Networks and ISDN Systems, 28, 1212-1231, 1996.
6. R. Iersalimschy, R. Borges, and A. M. Hester, “CGILua A Multi-Paradigmatic Tool for Creating Dynamic WWW Pages”, SBES’97 (Simpósio Brasileiro de Engenharia de Software), 1997.
7. International Data Corporation, Special Report, “Web-based Education/Training”, February 1997.
8. Internet 2, <http://www.internet2.edu>.
9. ICC, <http://www.les.inf.puc-rio.br/icc> (in Portuguese).
10. R. Johnson, “Frameworks = (Components + Patterns)”, Communications of the ACM, Volume 40, Number 10, October 1997.
11. LearningSpace, <http://www.lotus.com/home.nsf/welcome/learnspace>.
12. C. Lucena, H. Fuks, R. Milidiu, L. Macedo, N. Santos, C. Laufer, M. Ribeiro, M. Fontoura, R. Noya, S. Crespo, V. Torres, L. Daflon, and L. Lukowiecki, “AulaNet™ - An Environment for the Development and Maintenance of Courses on the Web”, in ICEE’98, 1998 (to appear).
13. L. M. Moura and C. J. Lucena, “O Ambiente Visual Spider para o Desenvolvimento de Aplicacoes para a Internet”, RITA, Universidade Federal do Rio Grande do Sul, 1998 (to appear, in Portuguese).
14. L. M. Moura, C. J. Lucena, and A. v. Staa, “The Spider Environment”, MCC46/97, Monografias em Ciência da Computação, Departamento de Informática, PUC-Rio, 1997 (also submitted to “Software Practice and Experience”)
15. W. Pree, “Design Patterns for Object-Oriented Software Development”, Addison-Wesley, 1995.
16. D. Roberts and R. Johnson, “Evolving Frameworks: A Pattern Language for Developing Object-Oriented Frameworks” in “Pattern Languages of Program Design 3”, Addison-Wesley, 1997.
17. James Rumbaugh, Michael Blaha, William Premerlani, Frederick Eddy, and William Lorensen, “Object-Oriented Modeling and Design”, Prentice Hall, Englewood Clifs, NJ, 1991.

18. H. A. Schmid, “Systematic Framework Design by Generalization”, Communications of the ACM, Volume 40, Number 10, October 1997.
19. Simon Fraser University, <http://virtual-u.cs.sfu.ca/vuweb/>.
20. Sociedade da Informação, <http://www.les.inf.puc-rio.br/socinfo> (in Portuguese).
21. Transferência de Calor, <http://www.les.inf.puc-rio.br/transcal> (in Portuguese).
22. University of British Columbia, <http://homebrew.cs.ubc.ca/webct/>.
23. Virginia Commonwealth University, <http://views.vcu.edu/wcb/intro/wcbintro.html>.