Gerosa, M.A., Cunha, L.M., Fuks, H. & Lucena, C.J.P. (2002) AulaNet-eLabora: Developing a groupware to bring the work and learning environment together, Electronic Proceedings of the 7th International Conference on Engineering and Technology Education - INTERTECH 2002, Santos-SP

AulaNet-eLabora: Developing a groupware to bring the work and learning environment together

Marco Aurélio Gerosa, Leonardo Magela Cunha, Hugo Fuks e Carlos José Pereira de Lucena

***Abstract* — In an information society, people must learn to work in groups with new communication and information technologies. With the AulaNet environment, developed using a prototyping approach, it is possible to carry out this type of cooperative learning. In this article, it is proposed the eLabora project, a component-based framework approach for the management of work groups based upon the AulaNet environment experience.**

***Index-Terms* — development of CSCW tools, groupware, learningware.**

I. INTRODUCTION

Society is changing. The rhythm of knowledge production and the new telecommunications technologies are changing the way that humanity lives and works [1]. Professionals dedicated to intellectual work—Knowledge Workers—are increasingly in demand. Besides having the necessary knowledge to carry out their jobs, these workers must have skills that are, perhaps, even more important: know how to learn, so they can adapt themselves continuously to the constant evolution of the work environment; know how to work within a group, which is one of the aspects most required today by corporations; and know how to creatively change an old knowledge set into new knowledge, the most important element in modern institutions. The pace at which information is being produced and made available also generates the need for workers to be in a state of lifelong learning in order to remain up to date.

These new needs required by the new work concept create a situation where work and learning environments should converge, and the teaching-learning process undergoes adaptations. The use of the Internet enhances learning and cooperative work [2], making the computer an appropriate tool for communication, for the sharing of information and for cooperation between members of a knowledge community.

Marco A. Gerosa, gerosa@inf.puc-rio.br, Leonardo M. Cunha, leocunha@inf.puc-rio.br, Hugo Fuks, hugo@inf.puc-rio.br, Carlos J. P. Lucena, lucena@inf.puc-rio.br, Laboratório de Engenharia de Software - Universidade Católica do Rio de Janeiro (PUC-Rio) - Marquês de São Vicente, 225 - Rio de Janeiro - RJ - Brasil, Tel. +55-21-2274-2731.

O projeto AulaNet é parcialmente financiado pela Fundação Padre Leonel Franca, pelo Ministério da Ciência e Tecnologia através de seu Programa de Núcleos de Excelência (PRONEX) bolsa nº 76.97.1029.00 (3366) e também através de bolsas individuais do CNPq e da CAPES.

Taking advantage of the characteristics of the Internet and looking for the construction of an environment where the new demands of learning are satisfied, the AulaNet [2] has been under development since July 1997 at the Catholic University of Rio de Janeiro. The AulaNet is an environment for the creation, application and administration of Web-based courses, based upon cooperative work manifested in the interaction between learners and instructors, with other learners and with educational content. Besides this, it adopts a groupware approach [3], being a system that is designed to support group learning.

The services offered by the AulaNet are organized according to the principle that, to learn in a group, a person must share ideas (communicate), be in tune with the other members of the group (coordinate), and satisfactorily carry out tasks (cooperate) [4]. In order to be able to work in groups, individuals must also communicate, coordinate and cooperate with each other, so the characteristics of the AulaNet learning environment can be used to prepare a group work environment. Thus the work environment will be brought closer to the learning environment and vice-versa, strengthening the relationship between work and learning.

This work environment, the eLabora, will be developed in a way that takes advantage of the experience acquired during the development of the AulaNet. Upon proposing eLabora, it is understood that it will be a groupware with a more general purpose. With some adaptations, mainly in terminology, it is possible to imagine that a course on the AulaNet would correspond to a project in eLabora, a class corresponds to a team, a teaching unit to a stage of work etc. In the same way that it can be "generalized", it can also be supposed that if eLabora had been implemented, the reverse would be possible, that is, it would be possible to instantiate the AulaNet or diverse other groupware starting from eLabora.

To facilitate these instantiations, it is intended that eLabora will be designed using a component-based framework approach. Some of the advantages of using this approach are [5]: a greater re-use of software, more speed in the integration of new services into the environment and the possibility of outside teams developing new functionalities.

II. The AulaNet

The AulaNet is an environment based upon a groupware approach for the creation, administration and application of Web-based courses. One of the objectives of the environment is to relieve the teacher of the task of programming for the Internet. The teacher can produce content through his usual work tools, such as a word processor, or re-use content that already exists in digital format, not having to manage and implement navigation around the environment. Moreover, a number of communication, coordination and cooperation mechanisms are offered to be used in the course in order to make it more interactive and structured.

A. The AulaNet Services

Upon investigating the dynamics of the interaction that favors cooperation within a work group, it is observed that in order to cooperate people have to coordinate, and that to coordinate people have to communicate [6].

Services offered by AulaNet are organized according to their main purpose, which can be communication, coordination or cooperation. These services are put at the disposal of the teachers at the time the course is created and while it is running, making it possible to select and configure those to be part of a course and available to the learners.

In Figure 1, it is shown the standardised interface for participating in courses on the AulaNet. The interface is made up of a main window and a menu that is presented graphically as a remote control unit. The main window is where the learner interacts with the teaching materials, with the instructor and with the other learners. The remote control unit contains the service menu that provides access to the selected communication, coordination and cooperation services.

Fig. 1. AulaNet interface

The communication services make it possible to exchange, to send information as well as to provide shared space to create shared understanding [7]. The AulaNet's communication services are implemented through an asynchronous forum-style text discussion tool (Interest Group), a synchronous conference chat style tool (Debate), an instantaneous message exchange with online participants tool (Contact with Participants), and tools for electronic mail with the teachers (Contact with Teachers) and with the class (Discussion Group).

The coordination services provide the means for group agenda and skills management. These services include tools for notification (Notices), basic course flow (Lesson Plan), assessment (Tasks and Examinations) and qualitative and quantitative accompaniment (Participant Follow-up).

The cooperation services provide the means for cooperative learning [8] through co-authoring (Teacher Co-Authoring and Learner Co-Authoring) as well as a course reference list (Bibliography and Webliography) and transferable content (Downloads).

III. The AulaNet 2.0 Architecture

The AulaNet works in a client-server architecture on the World Wide Web. Thus, if individuals are prepared to access the Internet and their browser version is compatible with the environment, it is not necessary the installation of any special software nor special configurations on their computers. The need to use plugins arises from the format of the files of the course contents. The most common image formats, for example, can be seen through the browser without the need of any additional plugin, although files of a specific program, such as Flash, need a plugin to be viewed.

Version 2.0 of the AulaNet uses an architecture based on servlets, which are Java classes instantiated by the Web server to receive requests from users, deal with them and dynamically generate pages in answer [9]. The AulaNet was developed using the Scriba technology [10], and uses a servlet as intermediary for all communication between the client and the server. Scriba provides its own language for embedding in HTML files that permits, among other facilities, access to databases, definition of variables for temporary data storage and calls to classes implemented in Java. The Java classes group the application's more complex and specific functions.

On the AulaNet, the Java language also is used to generate applets that are executed on user machines in order to implement the environment's synchronous communication services. The AulaNet 2.0 architecture is shown in Figure 2.

The AulaNet server is composed of a central module that interacts with the client's browser through the Internet and supplementary modules that carry out specific functions. The central AulaNet module is made up of a group of Java classes, HTML pages with embedded Scriba code and an ODBC database. The pages also have Javascript code [11] and dynamic HTML [12] to authenticate fields and to define the software interface. All data that is handled by the central module, such as the institutions and the courses, are stored in a relational database. The communication between the classes and the database is conducted by the JDBC-ODBC bridge [13], permitting the use of compatible SQL and ODBC databases.

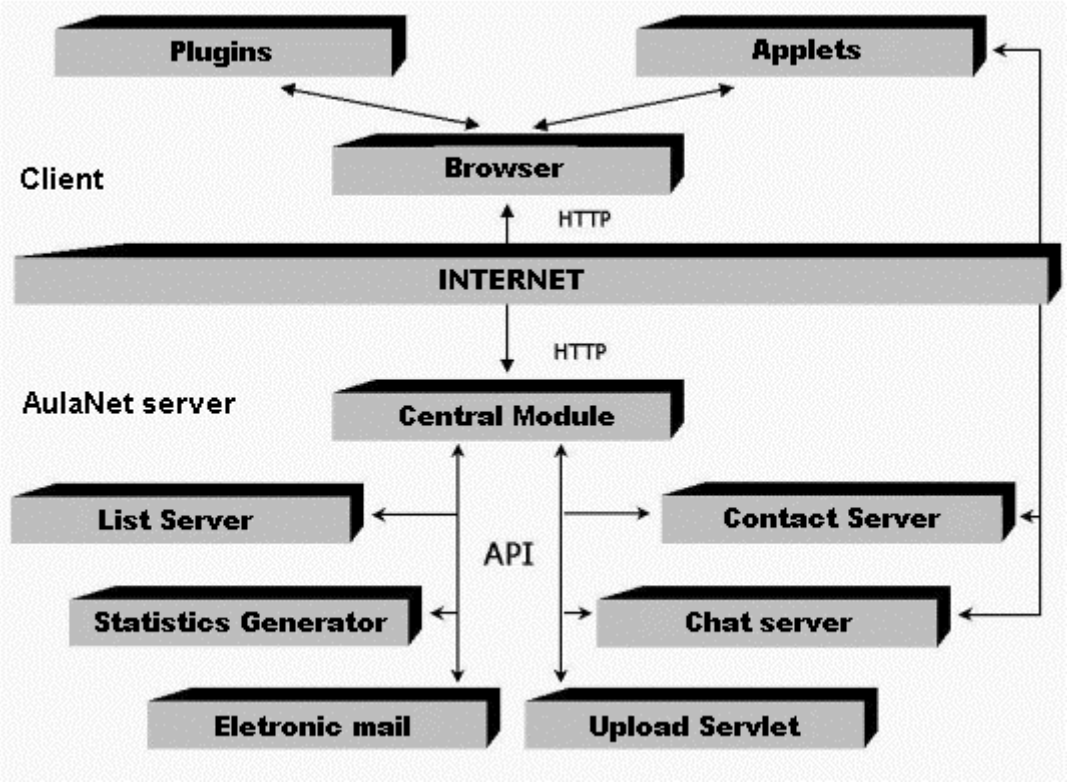

Fig. 2. The AulaNet 2.0 Architecture

The tools that implement the environments specific functions shown in Figure 2 are:

Lists Server: implements the discussion list of the Group Discussion;

Chat Server: implements the Debate service server;

Upload Servlet: permits transfer of files from the clients to the server;

Contact Server: permits the participants of a course to identify the online presence of other participants and to send them personal messages;

Electronic Mail: takes care of sending all the messages generated in the environment; and

Statistics Generator: supplies access statistics to the AulaNet server based on the analysis of the log of the Web server that hosts the environment.

IV. Evolving with the Technology

Scriba was created as a base technology for structuring the AulaNet environment. Despite the possibility of language extensions, the existing commands essentially permit page assembly based upon values stored in databases, the definition of variables and class calls. When Scriba was developed, Sun Microsystems still had not distributed the JSP technology [14] that although having some objectives in common with Scriba, also has other functionalities that would be interesting for the development of the AulaNet project. Moreover, the use of proprietary technology could make it difficult to bring new members into the team and even to develop new products when, for example, it became necessary to extend the existing language.

Another problem deriving from the use of Scriba was the dissemination of code through HTML files and Java classes. In the current AulaNet architecture, HTML files work as templates. Besides html tags, the HTML files also have Scriba and JavaScript commands, for the dynamic generation of pages. The Scriba commands can call classes implemented in Java that are used for functions that are more complex. This dissemination of code between different languages and files makes environment maintenance more difficult. When a problem is identified, it is necessary to check if the it has occurred in the template file, checking the HTML tags, the JavaScript instructions and the Scriba code, and when that is not the case, it is necessary to check for the problem in the Java classes called by the template file and the classes called by the classes themselves.

The Java classes, called by the template files, implement the specific functions of each file. Thus, despite having been developed using an object-oriented language, these classes works like a library of functions, which is similar to the functional paradigm [15]. As a result, when an alteration takes place in the database, such as adding a field to a table, it is necessary to find and update all classes and HTML files where that table is used. The object-oriented paradigm would resolve this difficulty, encapsulating data in the form of objects [16].

The current version of the AulaNet is aimed at education. It is intended that the new environment will be able to accept different groupware concepts, and not only educational ones. Other service proposals are being prepared, such as coordination based upon workflow, the use of distributed educational content applying the IMS (Instructional

Management Systems Global Learning Consortium) standard [17] and the use of software agents for group formation.

The AulaNet has been developed through prototyping. Its developers are doctorate, master's degree and undergraduate students at PUC-Rio. They, besides maintaining it, use it for their theses, dissertations and monographs, implementing and testing concepts from their works on the environment. As a result, the AulaNet grew and its functionalities were implemented as necessary. Now it is requiring a restructuring of the code.

V. eLABORA

As explained above, the AulaNet was developed in order to improve cooperative group learning, and its principles could be re-used in software for group work. This groupware would be the eLabora environment, a groupware with a more general proposition than the AulaNet. With some adaptations, mainly in nomenclature, it is possible to imagine that an AulaNet course would correspond to an eLabora project, a class would be a work team, a teaching unit a stage in a job etc. Just as it can be "generalized", it also can be supposed that if eLabora already had been implemented the reverse would be possible, that is, it would be possible to instantiate the AulaNet or other groupware starting from eLabora.

As stated by [5] a component framework is a software entity that supports components according to certain standards and permits that instances of these components to be connected to it. It establishes "environment conditions" for instances of the component and regulates interaction among them.

Component frameworks can be used alone, or they can cooperate with other components or component frameworks. Therefore, it is natural to model component frameworks as components. The eLabora architecture thus would be constituted of a component framework made up of an integrated set of component frameworks. These frameworks handle the communication, coordination and cooperation and would permit the structuring of the environment in software components.

A. The Architecture of the Software Components

The initial proposal for the eLabora Framework project and the communication, coordination and cooperation frameworks is to use a design pattern (Component-Based Design Pattern) reflected by the component technology [18] presented in Figure 3.

In this design pattern, a component (1) is a software implementation that can be executed in a logical or physical device. A component implements one or more interfaces that are "imposed" on it (2). This implies that the component satisfies some obligations, which are described later as a contract (3). These contractual obligations ensure that components developed independently may interact or not, and that they may be connected in environments in execution or design time (4). A component-based system is composed of a small number of component types, each one with its own specialized role within the system (5) and that is described by an interface (2). A component model (6) is the set of component types, their interfaces and the specification of interaction standards permitted between these component types. A component framework (7) provides a variety of runtime services (8) to support and reinforce the component model. Sometimes the component frameworks are like special-purpose operating systems despite the fact that they operate at a much higher level of abstraction [18].

The use of this design pattern is motivated by the subdivision of the frameworks into components with more specific purposes. For instance, Discussion Group and Debate services would be communication framework components that, for its part, would be a component of the eLabora Framework.

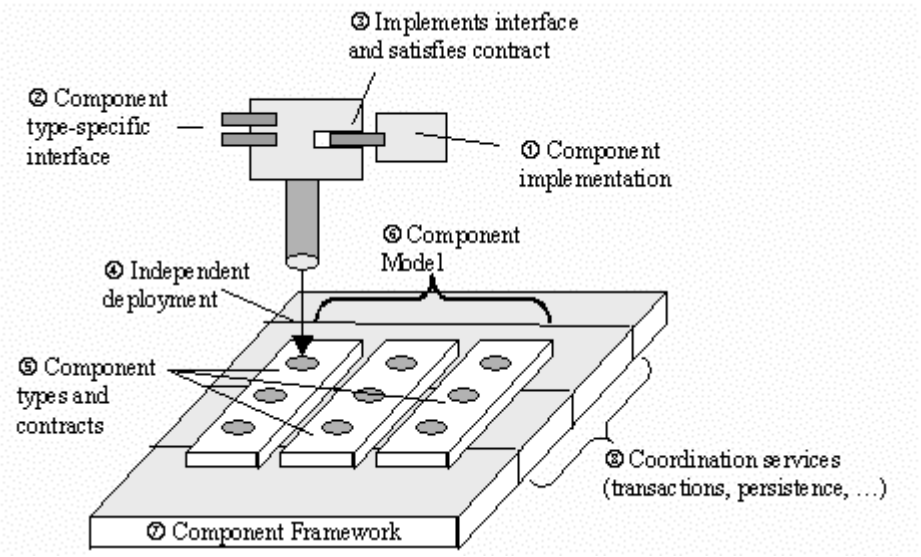

Fig. 3. Component-Based Design Pattern [18]

VI. Concluding Remarks and Future Work

The eLabora project can be influenced by technologies such as component-based development, since environment extendibility [18] is always taking place and is encouraged. The re-use of components can be justified by the need to implement other groupware with similar approaches, such as the AulaNet. The collaboration of external teams, through the development of components for the environment, will be encouraged and will favor the use of leading-edge technologies [5]. Finally, the code will be more modularized and the amount of time spent in maintaining it will be reduced through the use of software components [18], thus making it possible to study new concepts.

With teams from other institutions and companies able to develop new components to be coupled to both new services in the projects/courses as well as new management mechanisms, the AulaNet team will be able to focus its efforts on the component model, on the eLabora framework and on the integration of the various components. Thus, the institutions that may use the eLabora will have at their disposal the means to adjust it and add components to it according to their requirements. Moreover, the quality of the environment will improve in view of the fact that specialized teams will be able to act upon the environment's different components and frameworks.

In this paper it was presented the AulaNet 2.0 architecture, the difficulties identified in its maintenance and improvement, and some reasons that led to the proposal of the eLabora, which is a component-based framework approach for the management of group work.

As future work we intend to improve the use of software components, mainly through the use of the object-oriented paradigm. Another direction is the definition of the relationships (in the form of contracts) between the communication, cooperation and coordination frameworks. If these contracts are well defined, it is believed that integration of the new components with respect to both eLabora as well as to the "sub-frameworks" would be facilitated.

Through April 2001, more than 4,100 AulaNet installers had been distributed. By proposing a new work environment that is similar to a person's learning environment, we intend to encourage the convergence of the work and learning environments. This new environment can satisfy the needs of continued learning and help in training modern workers.

References

[1] Fuks, H. Groupware Technologies for Education in AulaNet. Computer Applications in Engineering Education. New York. Vol. 8. Issue 3 & 4, pp 170-177, December 2000. Online ISSN: 1099-0542 Print ISSN: 1061-3773 Wiley-InterScience.

[2] Lucena, C. J. P., Fuks, H. Professores e Aprendizes na Web: A Educação na Era da Internet. Rio de Janeiro: Ed. Clube do Futuro, October 2000. ISBN: 8588011018 (in Portuguese).

[3] Coleman, D., & Khanna, R. Groupware: Technology and applications. USA: Prentice Hall, Inc, 1995.

[4] Fuks, H., Laufer, C., Choren, R., & Bblois, M. Communication, Coordination and Cooperation in Distance Education. In Proceedings of V AMCIS'99 - Americas Conference on Information Systems, Association for Information Systems (AIS). Milwaukee, USA. pp. 130-132, August 1999.

[5] Szyperski, C. Component Software – Beyond Object-Oriented Programming. New York: ACM Press, 1999. ISBN 0201178885.

[6] Fuks, H. Aprendizagem e Trabalho Cooperativo no Ambiente AulaNet. Revista Brasileira de Informática na Educação, SBC. No. 6, Abril 2000. ISSN 1414-5685. (in Portuguese)

[7] Schrage, M. No more teams! Mastering the dynamics of creative collaboration. Currency Doubleday. New York, USA. 1995.

[8] Harasim, L., Hiltz, S. R., Teles, L., & Turoff, M. Learning networks: A field guide to teaching and online learning. 3rd ed. USA: MIT Press, 1997.

[9] Siyan, K. S. & Weaver, J. L. Inside Java. New Riders Publishing, 1997.

[10] Blois, M., Choren, R., Laufer, C., Ferraz, F. & Fuks, H. Scriba - A Tool for Developing Java Based Web Applications. In Proceedings of WebNet'99 World Conference of the WWW, Internet & Intranet. Honolulu, USA. vol 1. pp 112-118, November 1999.

[11] Goodman, D. & Eich, B. Javascript Bible. USA: Hungry Minds, Inc, 2001. ISBN: 0764533428.

[12] Goodman, D. Dynamic Html : The Definitive Reference. USA: O'Reilly & Associates, 1998. ISBN: 1565924940.

[13] Hamilton, G., Cattell, R. & Fisher, M. JDBC Database Access with Java – A Tutorial and Annotated Reference. USA: Addison-Wesley, 1997.

[14] JSP Technology. 2001 <http://www.java.sun.com/products/jsp> (Accessed 16/04/2001).

[15] Cousineau, G. & Mauny, M. The Functional Approach to Programming. Cambridge University Press, English edition, December 1998. ISBN: 0521576814.

[16] Pressman, R. Software Engineering: A Practitioner's Approach. 5th Edition. McGraw-Hill Higher Education, 2000. ISBN: 0073655783.

[17] Torres, V., Lucena, C. J. P. & Fuks, H. ContentNet: A Framework for the Interoperability of Educational Content Using Standard IMS. Computers & Education Journal submitted.

[18] Bachman, F. et alli. Volume II: Technical Concepts of Component-Based Software Engineering. Carnegie Mellon University. 2000. <http://www.sei.cmu.edu/publications/documents/00.reports/00tr008/00tr008title.html> (Accessed 04/16/2001).

[19] Mattsson, M., Bosch, J. Framework as components: A Classification of Framework Evolution. Workshop on Programming Environment Research. Ronneby, Sweden. 1998.