A Cooperative Environment for E-Learning Authoring

Manuele Kirsch Pinheiro* — Tiago Lopes Telecken**
Carlos Mario Dal Col Zeve — José Valdeni de Lima****
Nina Edelweiss**

** Institut National de Recherche en Informatique et Automatique (INRIA)*
ZIRST - 655, Avenue de l'Europe - F-38330 Montbonnot Saint Martin
Manuele.Kirsch_Pinheiro@inrialpes.fr

*** Instituto de Informática - Universidade Federal do Rio Grande do Sul*
Caixa Postal 15064, CEP 91591-970, Porto Alegre - RS - Brazil
{teleken, zeve, valdeni, nina}@inf.ufrgs.br

ABSTRACT. This article presents a cooperative environment for e-learning authoring, which allows the cooperation between educators and technical people in order to create material for e-learning courses. Our approach while building this environment was to use mainly web standards, like XML, SMIL, SVG and WebDAV, and open software. We combine some features like awareness and workflow to allow the cooperation between authors, without using for this any complex groupware structure. The result is a simple and low cost environment, ideal for low budget institutions.

RÉSUMÉ. Cet article présente un environnement coopératif pour la création des documents pour l'enseignement par ordinateur. Cet environnement permet la coopération entre des éducateurs et des techniciens pour créer le matériel pour des cours. Notre approche en construisant cet environnement était d'employer des standards web, comme XML, SMIL, SVG et WebDAV, et aussi des logiciels ouverts. Nous combinons dans cet environnement quelques éléments spéciaux, comme notification et workflow*, pour la coopération entre les auteurs, sans qu'il faille nécessairement utiliser un logiciel de travail en groupe complexe. Le résultat est un environnement simple et pas trop cher, idéal pour les institutions à bas budget.*

KEYWORDS: *E-learning authoring, cooperative authoring, awareness, workflow.*

MOTS-CLÉS : *création de documents coopératifs, enseignement par ordinateur, notification,* workflow.

1. Introduction

An important use for the World Wide Web is to support long distance learning, also called e-learning, characterized by the physical distance of the student to the teacher. Among all the existent technologies on the Internet, the Web provides the most capabilities for distance learning (Bos *et al.*, 1995). The web is used to present the courses and to connect students and teachers. Information and the knowledge, which are essential for the instructional process, are accessible and readable. The place, time and rhythm of learning are flexible (Bouras *et al.*, 1999). An important feature of this kind of courses is the more and more frequent use of multimedia in e-learning documents. By using multimedia resources, the authors intent to produce courses as close as possible to regular lectures.

The process of didactic material authoring for e-leaning, what we call in this paper "e-learning authoring", is naturally a cooperative process. To ensure the quality of these documents, the authoring process needs teachers, educators and also technical people, like web designers, programmers and database administrators. E-learning bases its performance on integrated efforts that combine multidisciplinary contributions (Lytras and Pouloudis, 2001). Thus, the e-learning authoring process requires the interaction of a multidisciplinary team during the creation and integration of contents and medias.

To assure the quality of the created documents, some kind of control of the cooperation between the authoring team members must be provided. This may be achieved defining the authoring process workflow (Sizilio and Edelweiss, 2000). A workflow (Casati *et al.*, 95; Workflow Management Coalition, 95) defines clearly all the activities to be executed during the execution of a process, their relationships and coordination, together with the agents responsible for their execution. In contrast to traditional organizations that place their emphasis on tasks to be performed by individuals in relatively static hierarchical organizations structures, a workflow places emphasis on efficient processes performed by teams dynamically formed for specific projects (Scheer, 1994). “Coordination is the act of working together harmoniously” (Malone and Crowston, 1994).

The use of a workflow to control collaborative work has been analyzed in recent researches. LaMarca *et al.* addresses collaborative work using a centralized architecture (LaMarca *et al.*, 1999). Grigori *et al.* propose in COO Project (Godart *et al.*, 1999) a flexible workflow based on different activities status in the management execution system itself (Grigori *et al.*, 2001). The need of flexible workflows to support collaborative work is identified in other researches (Agostini *et al.,* 2000) (Casati *et al.,* 1996). The effectiveness of the cooperation among actors is highly dependent on their awareness of their work (Agostini, 1996).

Due to its distribution, the web became a natural medium for cooperation support. It can be used not only to implement an e-learning course, but also to connect the group involved in the e-learning authoring process. Thus, the web will provide the communication between the team members, which may be physically apart. Collaborative authoring process can be defined as the use of workflow techniques to manage the creation of integrated data sets by more than one author.

This paper proposes an environment to support the e-learning authoring process. This environment is based on the web to connect the authoring team members, and is controlled by a workflow engine. Awareness is added to prevent problems that may not be identified by the workflow engine. Group awareness facilitates the assessment of present state (who is doing what) in a shared workspace. From a group's perspective the proposed environment provides: communication support for exchanging information and accessing shared data as well as documents and data during the authoring process; coordination support for the group; and the cooperative execution support for working on shared tasks (Haak and Wang, 1997).

To implement this environment, a cooperative open environment based on web technologies is also proposed. This open environment uses mainly web standards, like XML, SMIL, SVG and WebDAV, and open softwares. Thus the browsing and download features of the Web are extended with more sophisticated features for document upload, version management, member and group administration and more, with the goal of changing the Web from a primarily passive information repository to an active cooperation tool (Bentley and Appelt, 1997). Besides the workflow engine that guides the authoring process, further cooperative features are integrated to the authoring tools (specially, web authoring), as a lock mechanism to avoid problems due to concurrent access, an awareness support, and an annotation mechanism. Combining these features, the cooperation between authors is provided without using for this any complex groupware structure, but producing only a simple and "easy to use" authoring environment.

The goal of this paper is to discuss the proposed environment and the cooperative features used to allow the cooperation between authors, and how these features are used to reach a simple, yet powerful, cooperative environment for authoring of e-learning documents. This article is divided as follows. First, the complete environment is discussed. Then, the details of the workflow and awareness supports are presented, followed by a discussion about the implementation of this environment, where the cooperative features, like concurrent access control, annotations and awareness, are presented. After that comes a discussion about other e-learning environments. Finally, some conclusions about this cooperative environment are presented.

2. E-learning authoring supported by workflow and awareness

E-learning courses present particular features, partly due to the physical distance between teacher and students, but also because of the available multimedia resources, whose use is very important to diminish this distance. We are rapidly moving toward a world in which knowledge is constructed collaboratively at a distance by multidisciplinary teams, supported with an electronic communications and information infrastructure (Kanfer *et al.*, 2000). Traditional courses are usually planned and presented by the corresponding teacher, who is in charge of the whole teaching/learning process. E-learning courses, to be effective, shall not be constructed only by a unique teacher, responsible for the whole knowledge to be passed. Several teachers (or just specialists in the subject presented in the course) may contribute with the knowledge, thus assuring a better quality for the course. In order to develop attractive and communicative content, the teachers need additional skills, such as the knowledge on graphic design techniques, which generally they do not possess. So, the teachers in charge of the authoring of e-learning courses should be supported by a team with such skills (Fucks *et al.*, 2002). Technical people shall contribute in the course authoring, to insure the quality of the electronic documents, managing multimedia tools, implement communication tools (like chat, and discussion list), and allowing the access to databases where the students’ evolution is kept for analysis. Moreover, as the learning methods used in traditional teaching do not apply to long distance courses, the contribution of specialized persons in education and psychology shall be used. Thus, a multidisciplinary team will perform the e-learning authoring process.

To implement an e-learning course the first thing to do is to define the authoring team, and the person who will act as the team coordinator. The authoring environment proposed here uses a workflow to coordinate this team. A workflow defines clearly all the activities to be executed during a process, their coordination, defining also agents responsible for each one of these activities. An authoring workflow is defined, and then executed by a workflow engine, guiding and controlling the activities performed by all the team members. The team coordinator is responsible for the whole process, and he/she is the only person who may interfere in the workflow execution if necessary. Thus, the whole cooperation is handled under the workflow control. The workflow support is explained in details in the next sub-section.

As in a workflow all the actions to be performed are exactly defined, an awareness protocol is added to the environment to increase the cooperation among the group members, and to avoid problems that may arise during the execution. Although only the team coordinator is allowed to interfere with the workflow definition, all team members need information about the group activities. The awareness support allows authors to be aware of what their colleagues have been doing in the authoring process – thus they may access information about the work their colleagues are executing, as well as information about the activities that have already been done. The team coordinator may interfere in the workflow execution, based on awareness information, to solve unexpected problems. Awareness information will not be the same for all the team members – the team coordinator will need more information to better coordinate the group’s activities. In fact, the awareness information presented to a user (authoring group member) is adapted to the role played by this user in the authoring process, assuring that users receive convenient information to better play their specific roles.

2.1. *Workflow support*

A workflow defines not only all the activities to be executed during a process, but also the execution sequence and synchronization. The workflow defines also the role of the agent that will execute each activity, based on the skills that this agent shall present. Available agents are classified according to the roles they may play in the process. Once a workflow is defined, it may be instantiated one or more times. Each workflow instance corresponds to an independent process. The effective allocation of an agent to an activity is done only the moment this workflow is instantiated.

According to the WfMC (Allen, 2002), a workflow management system is a system that defines, creates and manages the execution of workflows through the use of software, running on one or more workflow engines, which is able to interpret the process definition, interact with workflow participants and, where required, invoke the use of informatics tools and applications. Thus, a workflow management system may be used to control the authoring process, when an authoring workflow is available.

In the proposed environment, the authoring process is divided in two phases, the “pre-authoring phase”, where an authoring workflow is defined, and the “authoring phase”, where this workflow is executed. The pre-authoring of courses consists of the set of activities that support the planning and modeling of hypermedia courses, conforming to appropriate models and methodologies. The authoring team coordinator, based on the composition of the authoring team, performs the pre-authoring phase. Each role and the activities it may perform shall be clearly identified previously, during the team definition. The resulting authoring workflow will keep the necessary commands and instructions for the management of the development of course authoring. The same workflow specification can be instantiated many times, each instance corresponding to a different courses. Figure 1 shows these phases, from the authoring workflow definition until the course production.

During the *pre-authoring phase*, executed by the authoring coordinator, the roles are defined and associated to the activities of the authoring process model. This is done in the following sequence. In a first moment, the authoring workflow is defined (activities and their sequence and synchronization), and the roles (teacher, designer) that will be necessary during the authoring process are created and associated to their access rights (write, read, modify). After this, each process activity is associated to the role(s) that allows the execution of this activity during the authoring phase. Finally, the connection between pre-authoring and authoring phases is done instantiating the workflow, when the agents to execute the activities are chosen among the available agents, according to their roles. This relationship between role and available agents may be a “many to many” relationship, and it should be defined in such way to determine the group composition and the members’ relationships.

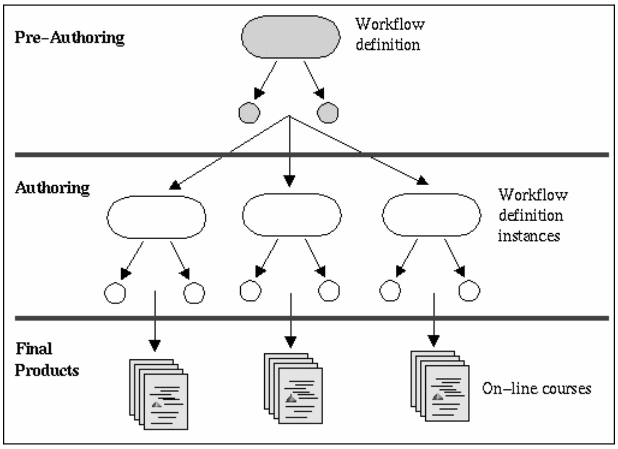


Figure 1. *Authoring process, from the authoring workflow definition until the courses production*

The results of a learning process are mostly a combination of different variables such as didactic style, educational strategy, technology deployment, appropriate selection of content, and personal qualities of teachers and students (Hampel and Keil-Slawik, 2001). Based on a learning approach and its underlying, the teachers describe the learning goals and design their pedagogical strategies, which will result in specific tasks (Santoro *et al.*, 2001). Thus, the workflow definition must consider the learning theory to be adopted by the authors on both, pre-authoring and authoring phases.

During the *authoring phase*, a workflow engine is used to control the creation a course, based on the previously defined authoring workflow. This workflow will guide all the authoring process, while coordinating the cooperation among authors, thus helping the group to efficiently produce their e-learning material, represented by web pages, multimedia resources, and students' controls. An important feature of the authoring process is to define the way the students shall behave during a course – which activities they shall perform alone, the sequence of these activities, which activities the student shall execute in groups. This definition shall be done by the authoring team during the authoring phase, and may also be represented by another workflow – which we call the *execution workflow*. This execution workflow will be instantiated separately for each student. Each one of these workflow instances will execute independently, interacting only when planned. For example, if teachers understand that the better strategy to achieve specific learning goals is the cooperative learning approach, they must define specific procedures to assess both individual students and students groups' performance and attitudes (Santoro *et al.*, 2001). This process can be described in the execution workflow.

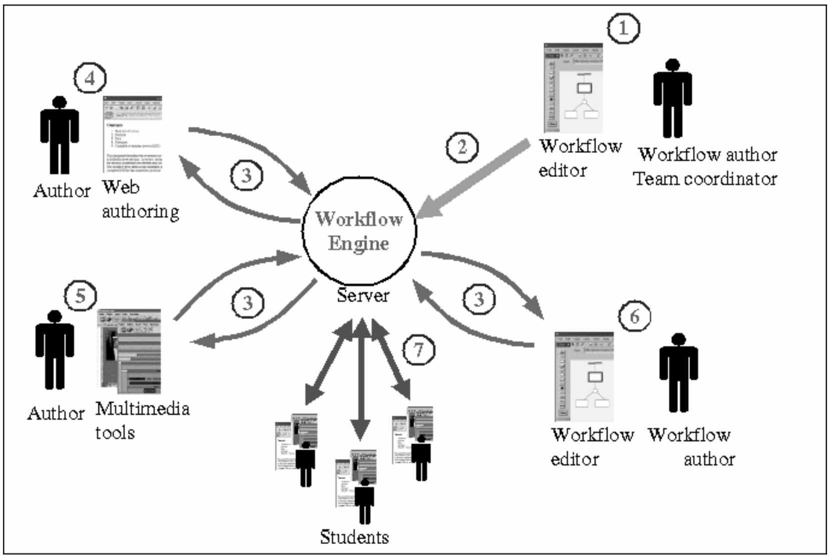


Figure 2. *Environment overview – the interaction between all the environment elements*

To implement this environment, the following five basic elements are necessary: a pre-authoring tool to define the workflow for the e-learning authoring process, a workflow engine that controls the authoring process, a web authoring tool, multimedia tools, and a set of functions supporting cooperative work (presented in this article). Those elements interact as shown in the Figure 2. In a first moment (1), the coordinator of the e-learning authoring process (team coordinator) specifies the workflow using a workflow editor. He then submits this workflow description to the server (2), where a workflow engine will use this description to control the authoring phase. After that, the course authors will be able to follow this process, by interacting with the workflow engine (3), while they are using the tools to create the course's web pages (4), the multimedia resources (5), and also to define the execution workflow (6), to be later executed by the students (7).

2.2. *Awareness support*

The work executed within a group produces better results when there is a harmonic interaction between the group members. This harmonic interaction depends on the understanding among these group members. This understanding is based on the reduction of the "uncertainty", which refers to absence of information, and "equivocability", which refers the ambiguity and the existence of conflicting interpretations in the work's subject. When reducing the "uncertainty" and the "equivocability", the group increases the understanding and then can reach a harmonic interaction, producing, then, better results (Dias and Borges, 1999).

To reach this understanding, the group needs: communication among the participants; coordination of theirs activities; a "group memory", which records the group's common knowledge, such as the interaction between the participants and the products developed by them; and the group needs awareness (Dias and Borges, 1999). Awareness is the knowledge on group activities, including past, presents and future activities. It's also the knowledge about the group itself and its overall status (Kirsch-Pinheiro *et al.,* 2001). It represents an understanding of the state of a system, including past activities, present status, and future options (Sohlenkamp, 1998).

In a cooperative work, decision and success depend on the integration of the efforts from different group members. When it happens, it is import that each member knows the work of his/her companions (Fucks and Assis, 2001). It is important to have information about the evolution of shared data, users' actions, etc., to improve individual contributions. These contributions may be improved by understanding the activities of the whole group, and that's why awareness is important in the development of collaborative activities (Preguiça *et al.*, 2000).

Providing computer-based mechanism for supporting awareness has been shown to be of central importance to the design of collaboration support systems. The reason for this is that being aware of the colleagues and their activities, is very important to make the work more natural and fluid (Gutwin and Greenberg 98, 99). Moreover, keep user update about what is happening to the product will reduce the risk for double-work and integration problems (Farschian 2001). It is the awareness support that makes possible the transformation of irregular interactions in consistent interactions over time (Dias and Borges, 1999). Awareness information forms a context for individual work by comparing it with the activities of the rest of group. It helps users to synchronize their work, coordinating themselves around their context. Without this context, the individuals are unable to measure their own work with respect to the group's goals and progress (Fucks and Assis, 2001).

Therefore, based on all the arguments presented above, the need of awareness support in cooperative environments is evident. This awareness support will lead the group to the understanding needed to reach its goals with the desired quality and efficiency.

Analyzing several systems with awareness support, Kirsch-Pinheiro *et al.* (Kirsch-Pinheiro *et al.,* 2001) identified some important characteristics needed to provide such support. These characteristics are organized in 6 questions (what, when, where, who, how, how much), each one identifying crucial aspects for the awareness support in cooperative systems. Among these questions, the question "when" tells us when the cooperative activities executed by the group are produced and when the awareness information produced by those activities are presented to the final user. Depending on when each one of those activities happens, it may be more or less important for the group. An activity may be produced in four situations, concerning the "when" question: (1) it may be produced in the past, when the activity began and finished in a past interval, and its results may not be valid or the same anymore; (2) in a continuous past, when the activity as been produced since a past moment until now (it can mean that somebody is working on it until now, or that the events' results are still valid); (3) in the present time, when there is somebody producing it right now (like moving a piece on a diagram); and (4) in the future, representing future options for the group (like deadlines and meetings).

We use to call "past awareness" the awareness support for activities produced in the past. They represent an important knowledge about the work process and it is very useful in asynchronous environments. In asynchronous environments, users usually collaborate accessing and modifying shared information without immediate knowledge about other users' actions. Although the absence of immediate knowledge, the overall information about the evolution of the collaborative activities, such as the evolution of shared data and the users' actions, may improve each user's contribution (Preguiça *et al.*, 2000).

The proposed e-learning environment is naturally an asynchronous environment, like the Web itself. So, it will need a past awareness support to keep the team members informed about their colleagues' activities and about the authoring process. Different collaborative activities, applications and users will demand different awareness information (Preguiça *et al.*, 2000). For example, a scheduler application may actively notify user of new appointments, while a collaborative writing system may simply maintain a modification log. In the writing applications, it is important that each user takes notice of updates performed in shared document. For this, information about updates should be automatically collected, maintained and, then, displayed to the users, using different user interfaces metaphors. That means that the awareness information relative to the evolution of shared documents should be tightly integrated (Preguiça *et al.*, 2000).

Thus, like the writing tool in the example above, our authoring environment will also need to give awareness information about the evolution of the course's documents. So, due to the fact that the proposed environment has a well-defined work process, defined by the authoring workflow, we decided to adopt activity-based awareness, which is based on the group's activities and organizational process. This model is capable of providing higher amounts of organizational awareness, that is, awareness of the overall organization of work, such as the interconnections between tasks (Farschian, 2001).

Therefore, the awareness support in the proposed environment is concentrated in the group's activities and in the work process defined by the authoring workflow. This support collects and presents the information about the activities executed during the authoring phase. Due to this information collection, the awareness support will also contribute with the group memory. Based on the group memory, the awareness support and the workflow engine can cooperate during the authoring phase. By consulting the information kept in the group memory, the awareness support may follow and present to the users the information about the work process and its progress. The workflow engine, by its turn, may consult the information about the users' actions collected by the awareness support, and base some decisions on it. Besides this cooperation between awareness support and the workflow engine during the authoring phase, the pre-authoring phase is also concerned with the awareness support. It is during the pre-authoring phase that the team's coordinator will define the roles that will be played by the team members. When defining those roles, the coordinator will also define what awareness information each role will need to be better executing its work.

Indeed, roles and awareness information have a very close relationship. Users need awareness information to better play their roles, but not all roles will need the same information. The awareness information presented to a user should be adequate to the user's role and preferences. A user playing a sample author role in our environment, for example, will need less information than the team's coordinator. A coordinator should be informed about the overall work progress. He/she needs a global view of the group activities, the defined tasks and deadlines, to take his/her decisions and guide the team's effort (Kirsch-Pinheiro *et al.*, 2001). If the coordinator has an adequate awareness mechanism, which presents relevant information, the coordination task will be easier (Borges and Pino, 99).

In the proposed environment, we adopt profiles to describe roles' needs and author's preferences. Those profiles indicate what activities (and within what period) are interesting for a role or a user. Thus, by manipulating their profiles during the authoring phase, team members may indicate of what activities he/she wants to receive awareness information. At anytime during the authoring process, he/she will be able to change his/her profiles, adapting it to his/her needs.

Together with the awareness support, users working cooperatively will need to communicate with their colleagues. Persons should communicate in order to coordinate their work efforts and cooperate on an objective (Fucks and Assis, 2001). Even the hierarchical, strict formal bureaucracy is entwined with officially invisible, informal cooperative work (Sohlenkamp *et al.*, 2000). In fact, good part of the work that have place in an office is based on informal communication (Smith, 99). Communication helps people to be aware of their colleagues' activities and also supplies a context for their work activities. It cooperates with awareness support to improve the team's understanding and the work quality.

For the proposed environment, we concentrate ourselves mainly in asynchronous communication, since the environment has asynchronous characteristics. We propose the use of annotations integrated with the authoring environment, and non-integrated tools, like e-mail and discussion lists. We also suggest that the group should use some synchronous tools too, like instant messages or conference systems, to take profit of those moments where there are many participants on-line. Those participants may use those opportunities to known each other, discuss about their work, and about the group's activities, thus improving their informal contact.

2.3. *Examples*

2.3.1. *An Example of an e-learning course development*

In this section, we present an example of a team using the proposed environment to create an hyperdocuments course. The creation of the course is done in the following phases: first, the team's coordinator, with the help of other team's members, will define the authoring process (that is, the authoring workflow), them he/she will submit the workflow description to the workflow engine. Second, the team's members will create an instance of this authoring workflow, and follow it to produce the course material. This material will be used in the course execution with the students. The first phase is the pre-authoring phase and the second, the authoring phase, and just after this, we will have the execution phase.

In the pre-authoring phase, the team's coordinator, together with some colleagues, will define the authoring process. This small group needs multiples skill to define this process. For example, our coordinator may be the teacher responsible for the discipline, he may aggregate forces from a designer, an educator and maybe other teachers. Together, they can consider aspects such as navigation, the course design, adequate learning theories, and also discuss what media elements they should use and how. Thus, this small group can define an authoring process more adapted to the hyperdocuments course. This authoring process is represented by the authoring workflow, created with the workflow editor integrated in the proposed environment (Section 3.1).

Once the authoring workflow is defined, the description file created for the workflow editor for this is deposed in the workflow engine. This workflow is used by the team to guide the authoring process, and it can also be used by other teams to create similar hyperdocument courses. The authoring process begins when this authoring workflow is instantiated. By instantiating this workflow, our coordinator will be able to assign the roles to the team's members, deadlines to the workflow activities, and activities to the roles. Once this initial assignment is done, the authoring process can start.

During this authoring process, the team members follow the instructions and activities defined in the authoring workflow. In our example, this team is composed by the small group that participates of the pre-authoring phase, increased by some other hyperdocument teachers, and also technical staff, engaged to built media, scripts, and other resources. Together, they will build the hyperdocument course material by interacting and cooperating through the proposed environment. Through this environment, they can use resources like awareness support, annotations, multimedia and Web editors, and access control with the WebDAV lock capabilities. They can also use disconnected resources, like chats.

While cooperating in the proposed environment, the team members can concentrate themselves in the course production, because the workflow will guide them, telling which activities to do, when and what should be their results. Depending on what has been defined in the pre-authoring phase, during the authoring phase the workflow may indicate the course structure, what media types (text, images, videos) are used in each unit of the course, when it is more appropriate do a group meeting, when a content revision should be done, etc. The team can concentrate itself in the cooperation, building the course content, communicating, discussing, being aware of the colleague's activities…

Thus, the team will be able to create the hyperdocument course: all the material (web pages, images, videos, dynamic resources, like scripts, etc) produced during this authoring process is kept in a central server, where they will be available for the students, during the course execution. This execution can also be defined through an execution workflow. Teachers and educators that belong to the team, during the authoring phase, can define this workflow. This workflow, also built with the workflow editor present in the proposed environment, is based on several aspects, such as the learning theory adopted and the course goals, and it is essentially the result of the understanding of team members of what is important and essential for the course execution.

2.3.2. *An example of an authoring process workflow*

In this section we present a little example for illustrating specifically the authoring process of a hyperdocument e-learning course, showed in figure 3. Here, the authoring process, that was defined in the pre-authoring phase, is organized with a team composed by two professors (each editing a chapter) and one reviewer. The workflow is started and each professor makes the edition of his/her chapter. When the professors finish their works, the reviewer can begin the revision and finally, after this, the course information is ready in the server.

With the awareness support monitoring all the authoring process, the reviewer can initiate his/her work based on the development of the prior editions. If the awareness informs that part of a text composed by one of the professors is ready, the reviewer can review this part before the professor finishes all the text edition. This procedure allows anticipating some work without modifying the workflow occurrence, giving the workflow more flexibility.

When the authoring phase is completed the course can be executed by the students, and the students' actions can be shown to the professors also through the awareness system.

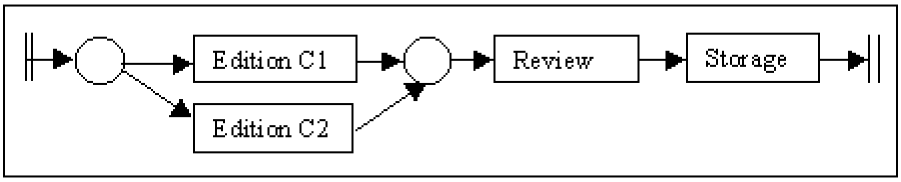


Figure 3. *Simple authoring workflow*

3. An open E-learning authoring environment

The web has been largely used for training and e-leaning purposes. This use is growing even more with the rise of new Web technologies, like XML, SVG and SMIL. Because of this, we present here an implementation of the e-learning authoring environment presented in the last section, using web technologies as the environment infrastructure. This allows profiting of the technologies, standards and available products for the Web also in the authoring process. This allows a better content production support. Modern web technologies like XML (XML, 2002), SMIL (Michel, 2001), SVG (Lilley, 2002) and WebDAV (WebDAV, 2001), and open software, like Amaya (Vatton, 2001) and Annotea (Kahan *et al.*, 2001), both created by the W3C (W3C, 2002), are used in this environment. The result is a low cost environment for e-learning authoring.

In order to allow the cooperation between the authors in a authoring phase, we provided in this environment means to divide the process in small pieces, to enhance their productivity by exposing their work to group review, to take profit of colleagues' expertise, and also to discuss, exchange ideas, etc. This is achieved by the following cooperative features: (1) a workflow engine, which controls the authoring phase, making the authors' live easier, since they may concentrate themselves mainly in the course's content production, and not in administrative tasks; (2) an awareness protocol, that allows the authors to be aware of what their colleagues have been doing in the authoring process; (3) a lock mechanism to avoid problems due the concurrent access, as the "lost update problem", very common in the Internet; and (4) an annotation schema, that allows the users to exchange comments, questions, points of views, etc, about the documents in construction.

The creation of the web pages that will compose the course's content is made using the software *Amaya*. Amaya is a browser/authoring tool developed by W3C that allows to publishing documents on the Web. It is a complete web browsing and authoring environment and comes equipped with a "WYSIWYG style" interface, similar to that of the most popular commercial browsers (Vatton, 2001). As it is used by W3C to demonstrate and test new developments in Web technologies, Amaya includes very modern features, like XML and CSS. Thus, using Amaya, users can generate pages in HTML and XHTML, generate CSS style sheets, MathML expressions and SVG vector graphics. Doing so, the users can build for their e-learning courses powerful web pages that can be viewed thanks to any modern web browser, although those pages have been constructed using a very lightweight tool. We enriched Amaya with a workflow editor and cooperative features, like awareness, both described in the following sub-sections. We also use, for communications purposes during the authoring phase, the *Annotea* client (Kahan *et al.*, 2001) implemented by Amaya. Annotea is an application for collaborative annotations based on state of art technologies, like RDF and XPointer. We chose in this environment not to adopt a complex groupware system, but only to use a few cooperative functions, remaining closer to simple applications design, working with open and worldwide known protocols and technologies. Therefore, by using a small tool that the users already knows, rather than a new complex tool, we expect to not make many changes in the users' habitual environment. This should cause a smaller impact on the users, making the environment adoption easier for them. In this section we present those features that form this environment and make possible the cooperation between the authors in the e-learning authoring process.

As the importance of multimedia for e-learning is unquestionable, besides the web authoring activity, the use of multimedia resources in this environment is also supported. There is no way to support e-learning without including multimedia in the courses material. There are many tools to create medias, like images, sounds and videos, and thousands of medias already available over the Internet. But, just putting all medias together in the same document will not necessarily enhance the course's quality. Those medias should be synchronized to compose a good presentation of the document and to reflect over time the content used by teachers, like demonstrations and examples. In this environment we concentrate our work in the use of multimedia synchronization tools, which give the authors the possibility to coordinate and construct rich multimedia presentations.

One of the recent standards for *multimedia synchronization* is the SMIL language. SMIL (Synchronized Multimedia Integration Language) (Michel, 2001) is a XML based language proposed by W3C that permits the integration of different media types, like streaming audio and video, images and text. SMIL enables simple authoring of interactive audiovisual presentations. With SMIL, it is possible also to augment existing languages with synchronization functionality. The result is languages like XHTML+SMIL, which includes a subset of the SMIL 2.0 specification, providing support for animation, content control, media objects, timing and synchronization, and transition effects to XHTML and CSS elements (Newman *et al.*, 2001).

We propose in our environment the use of multimedia authoring tools that support SMIL and SMIL+XHTML documents. These documents could be linked in normal web documents, such as the documents created by Amaya, and rendered by web browsers, like Microsoft Internet Explorer 5.5, or by its plug-ins, like Real Player. By consequence, the results of these multimedia synchronization tools can be added to the documents created by Amaya, producing richer hypermedia documents for the e-learning courses.

All the produced multimedia and web resources are kept in a central server, where the workflow engine processes the authoring tasks and constraints. By keeping all those resources in one server, we guarantee that all the produced e-learning material will be available for all authors involved in the authoring process, which will be able to use the available material and collaboratively interact to enhance its content.

Figure 4 gives an overview about our environment for e-learning authoring.

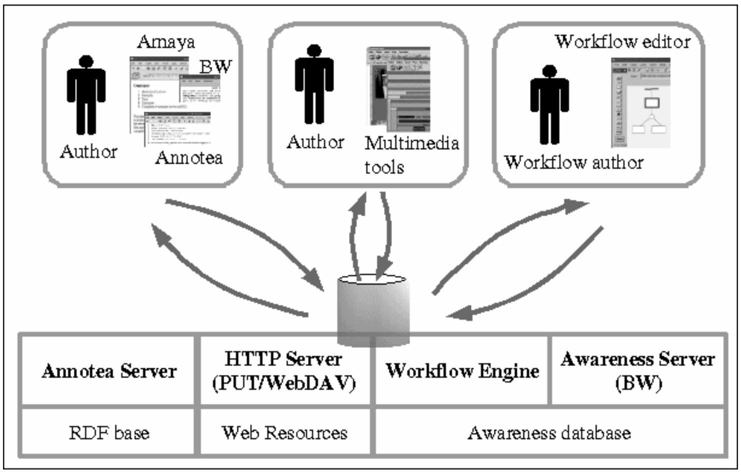
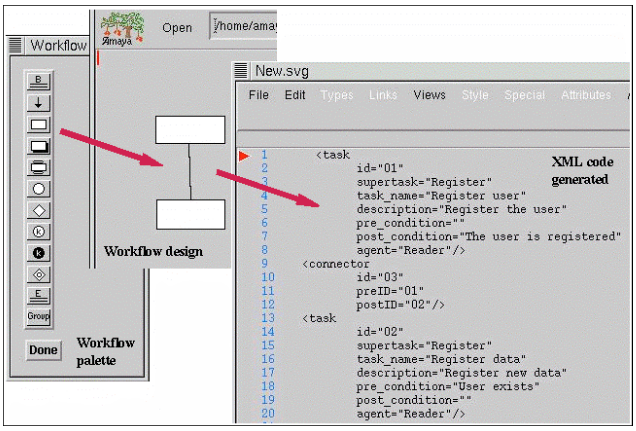


Figure 4. *The e-learning authoring environment structure*

3.1. *Workflow editor*

To help an author during a workflow specification, a graphical editor was built to define a workflow. It is used in two different phases: (1) in the pre-authoring phase by the authoring group coordinator, to define the authoring workflow, and (2) during the authoring phase, by the authors to define the execution workflow.

The graphical workflow editor uses Amaya and the SVG support present on it to build graphical symbols, which represent the workflow tasks, super-tasks, decisions, etc. The used modeling technique is based on the WIDE methodology (Workflow Development Methodology) (WIDE, 2002). Using this editor, the workflow author is able to define the workflow using graphical figures and their attributes. At the end of a workflow definition, the editor generates the description of the workflow mapped to a XML language specially defined for this purpose. The resulting XML code contains the commands, instructions and parameters for the workflow engine manager to control and coordinate the whole process (authoring process, or student executing a course). Figure 5 shows the palette used by this editor to produce the workflow definition and the XML code generated by it.



```
    <task
          id="01"
          supertask="Register"
          task_name="Register user"
          description="Register the user"
          pre_condition=""
          post_condition="The user is registered"
          agent="Reader"/>
  <connector
          id="03"
          preID="01"
          postID="02"/>
  <task
          id="02"
          supertask="Register"
          task_name="Register data"
          description="Register new data"
          pre_condition="User exists"
          post_condition=""
          agent="Reader"/>
```

Figure 5. *Workflow editor and the XML code produced*

3.2. *Workflow engine*

Once the authoring workflow is defined, the workflow author (team coordinator) should submit the produced workflow specification to the workflow engine. This workflow engine, hosted in the environment server, is responsible of the control of the authoring process, based on the given description. It is based on an interface composed by CGIs and Servelets, that allows the environment tools, like Amaya, to interact with this engine and follow the authoring process.

By interpreting this workflow specification in XML, the workflow engine determines which activities, resources and constrains the workflow has to control. Besides this description, the workflow engine may use awareness information to better control the authoring process, as discussed in Section 2. The workflow engine may explore the database kept by the awareness protocol server to know, for example, the progress of a document, recorded through awareness events like document “saves”. This database works like a memory of the group activities. The workflow engine can use it periodically to base some of its decisions. The workflow engine also can access the documents’ annotations and the concurrency control information, defined by WebDAV protocol, as well as any other resources defined in the implementation of workflow engine.

On the other hand, the authors in the authoring phase will be able to interact with the workflow engine by its CGIs interfaces. Using these interfaces, the authors may request a task, following the workflow sequence and constrains, and also interactively supply information about the realization of the task to the workflow engine (for example, by clicking on specific URLs when the task is completed). The same interaction can be done with the execution workflow instances, created for each one of the students that follow the course.

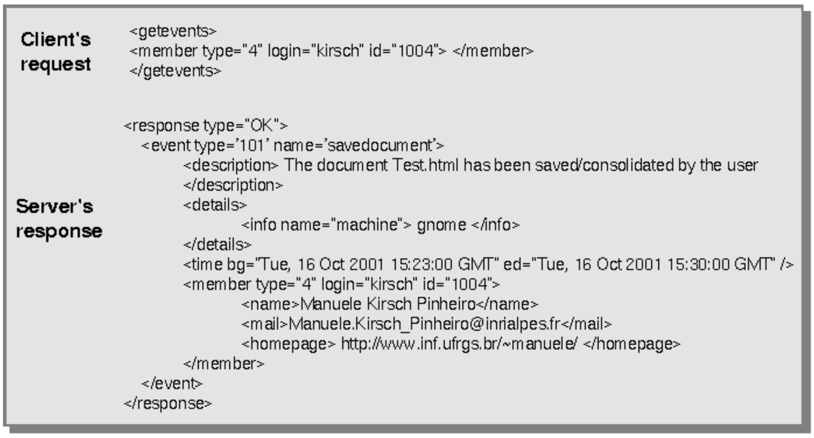
3.3. *Awareness protocol*

To provide the awareness support, a small awareness protocol, based on a XML structure, was defined. This awareness protocol allows the exchange of awareness information between many clients applications used by the group members, and an awareness server, allowing the diffusion of the awareness information in the complete authoring environment. Awareness information includes information about the users (team members), their preferences and roles, additionally to the activities produced by them.

The user produces the awareness information on the clients' applications, which format and pass this information to the server using this awareness protocol. The server is responsible of the control of this information, storing it in a reliable database. This database, as we said in Section 3.2, works like the group memory. The workflow engine consults this database to follow some users' activities, basing some decisions on it. For example, if an author has been absence for a long period, the workflow engine may decide to give his/her tasks to another author. This awareness server is not a general purposes server. It should be designed to deal with a specific set of activities from a user group. In our case, we designed it to deal with a user group in an e-learning authoring activity. It means that the structure of the database manipulated by the awareness server is previously known, which facilitates the workflow engine interventions.

Moreover, this awareness protocol is a connection-oriented protocol based on the request/response paradigm: a client sends requests in XML format to the server, which also answers in XML format. It includes 17 methods for manipulating the awareness information and for control purposes. Using those methods a client may inform the server about the beginning, conclusion or cancellation of an activity, set or change the user's information, and request the awareness information available according the user's profiles. An user's profile specifies the user's preferences about which activities, among the group activities, he should be notified, by indicating what activities are interesting for the user, and the time interval in which they are interesting. In our implementation, we combine three types of profiles: the profiles associated to a team's member, to a role, or both (a team's member playing a role). By using those profiles, users may indicate their global preferences and their preferences when playing a specific role. The team's coordinator may, by defining profiles for the roles during the pre-authoring phase, define what information should also be presented to a user playing a role in the authoring phase. By considering those profiles, the awareness server will be able to filter the information available in its database, showing only the information that will be more useful for the users.

So, by exchanging these information through this awareness protocol, authors in the e-learning authoring environment can inform their colleagues about their own activities and also be informed about their colleagues' activities. Figure 6 shows an example of a request of the awareness protocol, which is represented in a XML syntax: In that particular request, the client asks for awareness information, and the server answers with the events produced in the environment that accord with the user's profiles.



Client's request

```
<getevents>
<member type="4" login="kirsch" id="1004"> </member>
</getevents>
```

Server's response

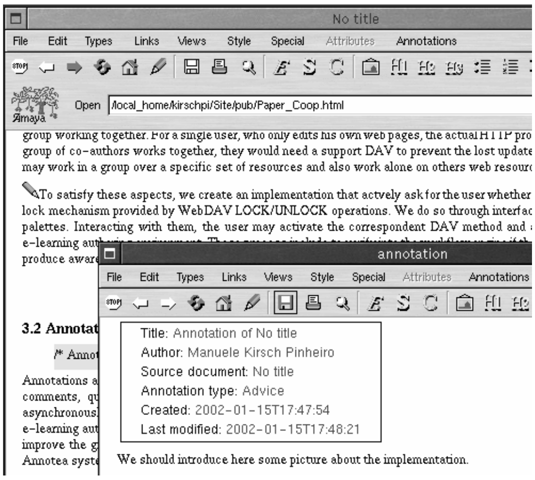
```
<response type="OK">
   <event type='101' name='savedocument'>
         <description> The document Test.html has been saved/consolidated by the user
         </description>
         <details>
                  <info name="machine"> gnome </info>
         </details>
         <time bg="Tue, 16 Oct 2001 15:23:00 GMT" ed="Tue, 16 Oct 2001 15:30:00 GMT" />
         <member type="4" login="kirsch" id="1004">
                  <name>Manuele Kirsch Pinheiro</name>
                  <mail>Manuele.Kirsch_Pinheiro@inrialpes.fr</mail>
                  <homepage> http://www.inf.ufrgs.br/~manuele/ </homepage>
         </member>
   </event>
</response>
```

Figure 6. *XML syntax used in the awareness protocol*

3.4. *Annotations*

Annotations act as a communication resource. They allow the co-authors of a document to exchange experiences, opinions, comments, questions, etc, and to enrich the cooperation process. Annotations give to the group the possibility to asynchronously discuss about the work they are performing together, leading them to a better document. We chose in our e-learning authoring environment to use an annotations resource to take profit of that important communication resource to improve the group capabilities during the e-learning process. The W3C's Annotea system is the used annotation resource.

Annotea is a web-based shared annotation system based on general-purpose open RDF infrastructure, where annotations are modeled as a class of metadata (Kahan *et al.*, 2001). It treats annotations as external statements made by an author about a Web document. Annotea stores these annotations on central RDF bases, in one or more annotation servers. As Amaya implements also an Annotea client since 4.x versions, our users can easily use this resource during the e-learning process without changing the application. They can add their annotations directly on the documents they are constructing or on the documents their colleagues are working on. By using annotations, the co-authors in the e-learning process may exchange ideas, comments, questions, finally, they may discuss about the documents and cooperate. The result is a better e-learning material, since this cooperation process will enrich the documents content.


No title
File Edit Types Links Views Style Special Attributes Annotations
Amaya
Open
annotation
File Edit Types Links Views Style Special Attributes Annotations

/local_home/kirschpi/Site/pub/Paper_Coop.html

group working together. For a single user, who only edits his own web pages, the actual HTTP pro
group of co-authors works together, they would need a support DAV to prevent the lost update
may work in a group over a specific set of resources and also work alone on others web resour

To satisfy these aspects, we create an implementation that actvely ask for the user whether
lock mechanism provided by WebDAV LOCK/UNLOCK operations. We do so through interfac
palettes. Interacting with them, the user may activate the correspondent DAV method and
e-learning aut
produce aware

3.2 Annotat

/* Annot

Annotations a
comments, qu
asynchronous
e-learning aut
improve the g
Annotea syste

Title: Annotation of No title
Author: Manuele Kirsch Pinheiro
Source document: No title
Annotation type: Advice
Created: 2002-01-15T17:47:54
Last modified: 2002-01-15T17:48:21

We should introduce here some picture about the implementation.

Figure 7. *An annotated web document in Amaya*

Moreover, the annotations form an important communication channel, not only among teachers during the authoring process, assisted by the proposed environment, but also between teachers and students reading the courses, during the execution workflow, since Annotea is also supported by other applications, like Mozilla. Thus, Annotea may be used in both, authoring and execution phases. In the authoring phase, it is an important authors communications media, as we discussed before. It can also be a powerful pedagogical tool, allowing the students' feedback about the course, during the execution phase, which is not addressed in this paper. Figure 7 shows an example of an annotated document in Amaya.

3.5. *The lost update problem*

When many users work in the authoring of a single web page, a common problem is the "lost update problem", when a user's update is overwritten by another user's update. The Figure 8 shows how it can be produced: (1) first, the users take the original version of a document from the web server; (2) after, both users make their changes in the document separately, but (3) one of them submit his own version before the other, so (4) when this later submit his version, he will overwrite his colleague's version.

The lost update problem is a typical one due to concurrent access, and in cooperative environments over the web, it is an actual issue, because important user's contributions can be lost. To deal with this aspect, we introduce in the Amaya editor a lock mechanism based on the Internet protocol designed for authoring purposes, the WebDAV. The WebDAV (WWW Distributed Authoring and

Versioning) is a set of extensions to the HTTP protocol, which allows users to collaboratively edit and manage files on remote web servers (WebDAV, 2001). It aims to extend the HTTP protocol to give place to an open architecture at protocol level, to develop new distributed authoring tools in the web, specially emphasizing the collaborative authoring of web pages (WebDAV, 2001; Goland *et al.*, 99).

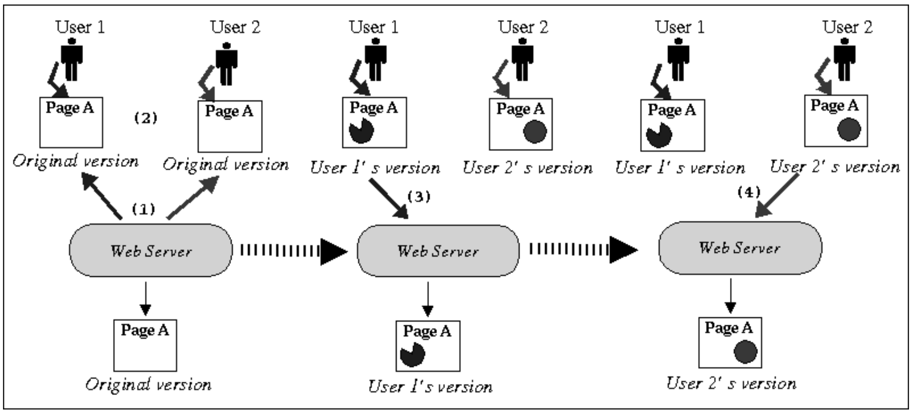


Figure 8. *An example of "lost update problem" situation*

Among all WebDAV features, the lock mechanism is the only treated by us. This lock mechanism defined in WebDAV has been design to prevent the resource overwriting (that is, to prevent the lost update problem) using the operations LOCK and UNLOCK. These operations can be used to create or refresh a lock, and to remove a lock, respectively. The WebDAV protocol defines, at this moment, just one lock type, write lock, for writing purposes. It controls the write access to a resource by limiting HTTP writing operations, like PUT, POST and DELETE. This means that only someone who possesses the lock will be able to execute those operations.

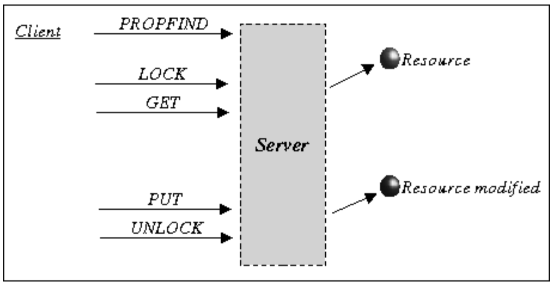


Figure 9. *A client/server interaction using HTTP/WebDAV requests*

In spite of the LOCK/UNLOCK operations, a WebDAV client may use the PROPFIND operation defined in the protocol to request the resources properties. It may ask the server what lock scope and type it supports through a property called “supportlock”. So, using this characteristic, called “lock capabilities discovery”, the client may discovery whether the server support lock and what kind of lock it supports. Also, the client may discovery whether a resource is locked and who owns the lock through the “lock discovery property” (Goland *et al.*, 99). The Figure 9 shows an example from interaction between client and server: (1) the client discoveries if the server supports locks by using the lock capabilities discovery; (2) since the server supports the desired lock, the client lock the resource and them request it from the server; (3) after change the resource locally, the client updates the resource in the server using HTTP PUT operation, and them (4) unlocks the resource, making it available to writing purposes for other clients.

Despite its operations, the WebDAV protocol cannot guarantee that the lost update problem will never be produced again, car it must be compatible with old clients, that not understand WebDAV operations, and with servers that do not support locks. By consequence, the protocol cannot force the sequence LOCK/GET/PUT/UNLOCK, which could prevent the lost update problem (Goland *et al.*, 99). So, in a heterogeneous environment, where common HTTP clients and WebDAV clients live together, concurrent access still may cause the lost of update problem. This problem can only be avoided in a homogeneous environment, with only WebDAV clients and servers with lock capabilities, where the sequence LOCK/GET/PUT/UNLOCK could be respected. This is the case of the e-learning authoring environment proposed by us. We use in this environment a central web server with lock capabilities, where all web resources used or produced by the group are stocked, and, as client, we improve Amaya with WebDAV lock capabilities. Doing so, we approach the homogeneous environment desired and prevent the lost update situation.

We introduce in the Amaya editor the support to the WebDAV lock and properties discovery capabilities. Thus, considering that users using Amaya may work individually with some resources, like their own web pages, and work cooperatively during the authoring process, we introduce those WebDAV capabilities through interface elements, like a menu. Interacting with them, the user may activate the correspondent DAV method and all process needed for our e-learning authoring environment. These processes include to verify into the workflow engine if that activity is allowed and to produce awareness information for this. Thus, we integrate this feature to the entire environment, providing it with a concurrent access control feature compliant with the Internet standards.

4. Related Tools

The authoring environment presented here obviously is not the only environment for e-learning purposes. There are others, like sTeam (Hample and Keil-Slawik, 2001), AulaNet (Fucks and Assis, 2001), LearningSpace (LearningSpace, 2002) and many others.

In their majority, the e-learning environments available are concerned mainly in the course execution, not in the authority process. That is the case of the sTeam environment that combines the idea of room-based virtual world with basic document-management functionalities (Hampel and Keil-Slawik, 2001). Some of them integrate both, authoring and execution, but they are often complex systems not designed to support cooperation. AulaNet (Fucks and Assis, 2001), for example, defines instructor, coordinator and co-teacher roles, but it concentrates mainly in the course execution, assuming that the authoring is conducted mainly off-line.

This approach has two points that we do not approve in our environment: the complexity and the absence of cooperative features. While building this e-learning authoring environment, we searched a simple, yet powerful, environment that minimizes the authors’ adoption impact, but still allows the cooperation between teachers, designers and other professionals involved in the course’s authoring process. Indeed, this authoring process definition and its control by a workflow engine, adopted in our environment, is an important advance, because it may improve the cooperation efficiency. Also, features like annotations and awareness give to the authors the sense of “group”, making of them a real team.

There are other environments for e-learning authoring/execution that include many cooperative features, like message boards and conference tools. But, often those systems are based on big groupware systems, or use some proprietary tools, like databases or languages. That is the case of LearningSpace system (LearningSpace, 2002), proposed by Lotus. This approach has an important disadvantage: the cost. In many situations, small and middle size institutions have not enough budgets to invest in software licenses and professionals training to implement an e-learning authoring solution like this. So, we adopt a simpler approach: we use only Internet standards and open software, like Amaya, Annotea and Apache web server (Apache, 2002). Doing so, we reduce the environment cost and make its results, the courses, compliant with the web standards. By consequence, these courses can be viewed in any modern browser. They do not demand any special environment to be executed by the students.

Besides that, there are other features, especially synchronous and asynchronous communication, that are not directly included in our environment, but there are present in many others. There are available in the Internet many systems for asynchronous and synchronous communications. The email system and the Annotea system, for instance, allow the asynchronous communication among users, while instant messagers like ICQ (ICQ.com, 2002) and Jabber (Jabber, 2002), conference systems like NetMeeting (NetMeeting, 2002) and CUSeeMe (CUSeeMe, 2002), among others, implement synchronous systems. Due to this plurality of systems, we chose to use directly in our system only one asynchronous system, Annotea, and leave the users free to choose what further systems to use for communication purposes. That is, we leave the team free to choose the better communication tool for them.

5. Conclusions

In this article we discuss the cooperative features used by us to build a complete authoring environment for e-learning. Through these features we reach a cooperative authoring environment, without the use of any complex groupware system, neither expensive tools. We only introduced in our environment some features that make possible to a group of the authors to produce cooperatively e-learning materials. So, those authors do not need to change their usual authoring environment, but only to learn about those new features available, helping to improve the environment adoption.

Inside this cooperative authoring environment, authors may produce cooperatively e-learning materials, by defining and following the authoring process under the control of the workflow. Doing so, they produce e-learning materials for their course. The content of the produced course is created and controlled only by the authors themselves. We strongly believe that the work in a group, composed by educators and technical people and the presence of a workflow describing the authoring process will help authors to produce a better course content.

Therefore, this cooperative e-learning authoring environment has been produced using mainly open tools and standards. This allows obtaining low cost systems applicable in almost any real world situation and deployable at a large scale. Besides this, it takes advantage of modern technologies brought through the latest web standards. We have shown, for example, how XML has been used in the Awareness Protocol and in the workflow definition, and how WebDAV's lock capabilities have been used, but the environment also uses other technologies, as XHTML, MathML and SVG in the web pages created by Amaya and SMIL in the multimedia resources. The use of all those up-to-date technologies results in the production of standard web resources for the e-learning courses made in this environment. By consequence, those resulting resources may be largely distributed and presented by any available modern browser.

Acknowledgements

This research is supported by CNPq (Brazilian National Council for Scientific and Technological Development) and INRIA (French National Institute for Research in Computer Science and Control). Authors would like to thanks Opera Research team and W3C team at INRIA Rhone-Alpes, France, and SIGMA team at II/UFRGS, Brazil.

6. References

Agostini, A., De Michelis, G. A Light Workflow Management System Using Simple Process Models. *Computer Supported Cooperative Work: The Journal of Collaborative Computing*, Kluwer Academic Publishers, Vol. 9, No. 3-4, August 2000, pp. 335-363.

Agostini A., De Michelis G., Grasso M.A., Prinz W., Syri A. Contexts, Work Processes and Workspaces. *Computer Supported Cooperative Work: The Journal of Collaborative Computing*, Kluwer Academic Publishers, Vol. 5, No. 2-3, 1996, pp. 223-250.

Allen, R., *Workflow: An Introduction*. Available in: http://www.wfmc.org/standards/docs/Workflow_An_Introduction.pdf. Access: Jan 2002.

Apache HTTP Server Project. Available in http://httpd.apache.org/. Access in: Jan 2002.

Bentley, R.; Appelt, W. Designing a System for Cooperative Work on the World-Wide Web: Experiences with the BSCW System. *Proceedings* of the 30th Hawaii International Conference on System Sciences, Maui, Hawaii, 1997.

Borges, M.R.S.; Pino, J.A. Awareness Mechanisms for Coordination in Asynchronous CSCW. In: *Proceedings* of 9th Workshop on Information Technologies and Systems – WITS'1999. Charlotte, North Carolina, 1999.

Bos, E. S.. Kikstra, A.; Morgan, C. M. Multiple levels to use the Web as a Learning Tool. *Proceedings of ED-TELECOM 96*, pages 31-36 – Boston, Mass USA, June 17-21, 1995.

Bouras, C.; Gkama, A.; Tsiatsos, T. Distributed Learning Environment using Advanced Services over the Internet. *Proceedings of IASTED International Conference Internet and Multimedia Systens and Applications,* October 18-21, Nassau, Grand Bahamas, 1999.

Casati, F., Ceri, S., Percini, B., Pozzi, G., Conceptual Modeling of Workflows, Proceedings of The Object-Oriented and Entity-Relationship Conference, Gold Coast/Australia: 1995. pp. 341-354.

Casati F., Ceri S., Pernici B., Pozzi G. Workflow Evolution. *Proceedings of the 15th International Conference on Conceptual Modeling, ER'96,* Cottbus, Germany. Springer Verlag, 1996.

CUSeeMe. Available in: http://www.cuseemeworld.com/. Access in: Jan 2002.

Dias, M.S., Borges, M.R.S., Development of groupware systems with the COPSE infrastructure, Proceedings of International Workshop on Groupware, IEEE Computer Society, Cancun, Mexico, September 1999, pp. 278-285.

Farschian, B. A. Integrating geographically distributed development teams through increased product awareness. *Information System Journal*, Vol. 26, No. 3, 2001. Elsevier Science, Netherlands. pp. 123-141.

Fucks, H.; Assis, R. Facilitating perception on virtual learning based environments. *Journal of Systems and Information Technologies*, Vol. 5, No. 1. Edith Cowan University, 2001. pp. 93-113.

Fucks, H.; Gerosa, M.A.; Lucena, C.J.P The development and application of distance learning course on the Internet. *Journal of Open and Distance Learning*, Vol. 17, No. 1, Feb. 2002.

Godart, C., Charoy, F., Perrin, O. and Skaf-Molli, H.: Cooperative Workflows to Coordinate Asynchronous Cooperative Applications in a Simple Way. ICPADS 2000: 409-416.

Goland, Y., Whitehead, E., Faizi, A., Jensen, D., HTTP Extensions for Distributed Authoring – WebDAV, RFC 2518, IETF, 1999. Available in: http://andrew2.andrew.cmu.edu/rfc/rfc2518.html. Access: Sept. 2001.

Grigori, D., Charoy, F. and Godart, C.: Anticipation to Enhance Flexibility of Workflow Execution. *Proceedings of the 12th Int. Conf. On Database and Expert Systems Applications - DEXA 2001*, Munich, Germany, 2001. pp. 264-273.

Gutwin, C., Greenberg, S. Effects of awareness support on groupware usability. *Proceedings* of CHI'98 – Conference on Human Factors in Computing Systems. 1998. ACM Press. pp. 511-518.

Gutwin, C., Greenberg, S. A framework of awareness for small groups in shared-workspace groupware, Department of Computer Science, University of Saskatchewan, Canada, 1999, Technical Report 99-1. Available in: http://www.cpcs.ucalgary.ca/papers/1999/99-AwarenessTheory/html/theory-tr991.html. Access: Sept. 1999.

Haak, J. M.; Wang, W. Flexible Support for Business Process: Extending Cooperative Hypermedia with Process Support. *GROUP '97*, Phoenix, Arizona, USA, 1997.

Hampel, T.; Keil-Slawik, R. ` team: Structuring information in team-distributed knowledge management in cooperative learning environments. In *Journal of Education Resources in Computing*, Vol. 1, No. 2, 2001. ACM Press.

ICQ.com. Available in: http://web.icq.com/. Access in: Jan 2002.

Jabber. Available in: http://www.jabber.org/. Access in: Jan 2002.

Kahan, J., Koivunen, M.R., Prud'hommeaux, E., Swick, R., Annotea: An Open RDF Infrastructure for Shared Web Annotations, *WWW10 International Conference*, Hong Kong, 2001. Available in: http://www.w3.org/2001/ Annotea/Papers/www10/annotea-www10.html. Access: Aug. 2001.

Kanfer, A. G. *et al.* Modeling Distributed Knowledge Process in next Generation Multidisciplinary Alliances. *AIWoRC´00 Academia-Industry Working Conference on Research Challenges*, April 27-29, Buffalo, New York, 2000.

Kirsch-Pinheiro, M., Lima, J.V., Borges, M.R.S., Awareness em sistemas de groupware, *4 Jornadas Iberoamericanas de Ingenieria de Requisitos y Ambientes de Software*, Santo Domingo, Costa Rica, 2001. Available in: http://www.inf.ufrgs.br/~manuele/awarenessV2.zip. Access: Mar, 2001. (In Portuguese).

LaMarca, A., *et al.*, Taking the Work out of Workflow: Mechanisms for Document-Centred Collaboration, *Proc. ECSCW 99*, Copenhagen, 1999.

LearningSpace. Available in http://www.lotus.com/home.nsf/welcome/learnspace. Access in: Jan 2002.

Lilley, Chris. W3C Scalable Vector Graphics (SVG). Available in: http://www:w3.org/Graphics/SVG/. Access in: Jan 2002.

Lytras, M. D.; Pouloudi, A. E-learning: Just a waste of time. Proceedings of the Seventh Americas Conference on Information Systems – AMCIS 2001, [CD Proceedings], pp. 216-222, Boston, Massachusetts, USA, August 3-5, 2001.

Malone, T. W.; Crowston, K. The Insterdisciplinary Study of Coordenation. *ACM Computering Surveys*. 1994

Michel, T. *Synchronized Multimedia*. Available in http://www.w3.org/ AudioVideo/. Access: Aug, 2001.

NetMeeting. Available in: http://www.microsoft.com/windows/netmeeting/. Access in: Jan 2002.

Newman, D., Schmitz, P., Patterson, A., XHTML+SMIL Profile, W3C Working Draft, 2001. Available in http://www.w3.org/TR/XHTMLplusSMIL/. Access: Aug, 2001.

Preguiça, N.; Martins, J. L.; Domingues, H.; Duarte, S. Data management support for asunchronous groupware. *Proceedings* of CSCW'00. December 2-6, 2000. Philadelphia, USA. ACM Press. P. 69-78.

Santoro, F. M.; Santos, N.; Borges, M.R.S. Students' assessment in computer-supported cooperative project-based learning environments. *Proceedings* of EDMEDIA'01 – Word Conference on Educational Multimedia, Hypermedia & Telecommunications.2001.

Scheer, A. W. *Business Process Reengineering*. (2nd edition). Springer-Verlag, new York, 1994.

Sizilio, G.R.M.A.; Edelweiss, N. Workflow for the Authoring of Long Distance Learning Courses. *Proceedings* of the ICECE´2000 – Int. Conf. On Engineering and Computer Education, Aug. 27-30, 2000, São Paulo, Brazil.

Smith, I., Toolkits for multimedia awareness, *Proceedings* of CHI, 1996. Available in: http://www.acm.org/sigchi/chi96/proceedings/doctoral/Smith/ies_txt.html. Access: July 1999.

Sohlenkamp, M.; Mambrey, P.; Prinz, W.; Fuchs, L.; Syri, A.; Pankoke-Babatz, U.; Klöckner, K.; Kolvenbach, S. Supporting the distributed German government with POLITeam. In *Multimedia Tools and Applications,* Vol. 12, No. 2, 2000. Kluwer Academicc Publishers, Netherlands. P. 39-58.

Sohlenkamp, Markus., Supporting group awareness in multi-user environment through perceptualization, Paderborn, Fachbereich Mathematik-Informatik der Universität – Gesamthochschule, 1998. Available in: http://orgwis.gmd.de/projects/POLITeam/ poliawac/ms-diss/. Acess: Jan, 2001.

Vatton, I., *Welcome to Amaya*. Available in http://www.w3.org/Amaya/. Access: Aug, 2001.

W3C. The World Wide Web Consortium: Leading the Web to its Full Potential… Available in http://www.w3.org/. Access: Jan. 2002.

WebDAV *Welcome to WebDAV Resources*. Available in: http://www.webdav.org/. Access: Ago, 2001.

WIDE. The WIDE Workflow Modeling Language. Available in: http://dis.sema.es/projects/WIDE/Documents/4080-2.pdf. Access: Jan. 2002.

Workflow Management Coalition. The Workflow Reference Model. Brussels, Jan. 95. Available in: http://www.wfmc.org/standards/docs/tc003v11.pdf. Access: Jan. 2002.

XML. *Extensible Markup Language (XML)*. Available in http://www.w3.org/XML/. Access: Jan. 2002.