Interactive Educational Multimedia, number 4 (April 2002), pp. 40-52
http://www.ub.es/multimedia/iem

Supporting personalization in a web-based course through the definition of role-based access policies

Ignacio Aedo, Susana Montero and Paloma Díaz

Laboratorio DEI. Departamento de Informática
Universidad Carlos III de Madrid

aedo@ia.uc3m.es

smontero@inf.uc3m.es

pdiaz@inf.uc3m.es

Abstract

Role-based access policies model the users domain by means of complex structures where roles, which represent jobs or responsibilities assumed by users, are specialized into more concrete subroles which inherit properties and authorizations from their parents. Such an approach can be applied within the context of educational applications, where different roles are easily identified each of which has different views of the same information items and different capabilities to modify them. Moreover, even though this approach, has only been oriented towards modeling security requirements, it can be extended to support personalized access to the information. In this paper, we describe how to combine the basic principles of RBAC policies and adaptation with a view of providing personalized access to the different types of users of a web-based course. Moreover, we also present Courba, a platform to generate personalized web-based courses using XML to support the definition of access policies.

Keywords

Educational hypermedia; web-based educational applications; role based access policies; XML; adaptation.

1. Introduction

Nowadays there is no doubt about the utility of hypermedia and web technologies to make computer-based educational tools more accessible and, at the same time, more effective.

On the one hand, the combination of a hypertext associative structure with multimedia information items has been proved as a useful educational resource in multiple hypermedia educational applications such as [Benamou and Celentano, 1994; Díaz et al., 1998; Tergan and Lechner, 2000]. While multimedia supplies a greater expressiveness, hypertext provides a geometry that allows information to be freely browsed according to the needs and preferences of users. Moreover, multimedia also introduces the concept of interactivity that increases the system utility by offering the user a wide variety of opportunities to play an active role in the information transmission processes. This latter feature results particularly interesting in educational applications since a passive reception of information does not generate knowledge and, therefore, learning environments should encourage an active user involvement [Quentin-Baxter and Dewhurst, 1992].

On the other hand, moving platform-dependent hypermedia to the web arena has torn down time and distance barriers, opening educational applications to a wider and heterogeneous group of users. The need for educational tools able to deal with different learning styles, that was repeatedly stated [Allinson and Hammond, 1990; Barker, 1993], becomes more evident in such a multi-user environment. Indeed, in the last years several adapted or adaptive hypermedia and web-based applications have been proposed including InterBook [Brusilovsky et al., 1998], AHA! [De Bra and Calvi, 1998], ELM-Art [Weber and Specht, 1997], SmexWEb [Albrecht and Koch, 2000] or Tangow [Carro et al., 2001]. For the shake of readability and, since web-based applications are a particular case of hypermedia, we will use throughout this paper the term hyperdocument to refer to hypermedia applications, whether implemented or not in a web environment.

In order to adapt the hyperdocument, course developers have to build a user model gathering relevant information about the user knowledge, learning style, goals and needs. In predictive statistical models, two approaches are applied for user modeling: content-based and collaborative- based learning [Zukerman and Albrecht, 2001]. Content-based learning uses an individual model based on the user's past behavior while collaborative-based learning applies group based models on the assumption that users behave in a similar way. This user model based on groups can be specified borrowing some concepts from Role Based Access Control (RBAC) policies that have been successfully applied in security modeling [Sandhu et al., 1994; Ferraiolo and Kuhn, 1992; Ferraiolo et al., 1999]. RBAC approaches model users by means of roles, which represent jobs or responsibilities. Multiple -level of hierarchical roles are defined by means of a Directed Acyclic Graph (DAG) structure where general roles are specialized into more specific roles that inherit properties and authorizations from their parents. Each actual user is assigned a number of roles that will determine her operation capabilities.

The work presented in this paper is focused on the combination of the basic principles of RBAC policies and adaptation with a view of providing effective access to the different types of users of a web-based course. Firstly, we extend the typical RBAC users' structure in order to introduce a basic element in an educational environment: the team which represents a group of roles joined to reach a common goal. This extended users' structure makes up the basis for the definition of access rules that specify which information is presented to each user role, in which form and sequence. Moreover, it is presented Courba a web-based platform to develop personalized courses. Finally, we outline some conclusions and future work.

2. *Adapting hypermedia educational systems*

Educational applications should guide the student's learning process through the information space by applying flexible combinations of various sources of information with different models of training, student profiles and concepts [Beltran, 1992] in order to deal with different learning styles. To apply this maxim to educational hypermedia applications we analyze which are the components of such applications and how they can be adapted to meet their user's learning expectations and needs. Moreover, we study how RBAC principles can be applied to adapt the hyperdocuments to different kinds of users.

2.1. Adaptations in a hypermedia application

Hyperdocuments are collections of multimedia documents organized in a hypertext net. Each document or node includes several information pieces, usually referred to as contents. From this definition we can foresee that two are the candidates to be adapted in a hypermedia educational application: the browsing structure (*navigation adaptation*) and the content delivered to the user (*presentation adaptation*).

The browsing structure of a hyperdocument is made up of a series of links defined among the nodes as well as some navigation tools such as visual maps, active indexes, guided tours, marks, footprints and backtracking mechanisms [Nielsen, 1995]. Most adaptive hypermedia tools offer different possibilities to support navigation adaptation including links hiding, sorting and annotation, direct guidance or maps adaptation [Brusilovsky, 2001].

Concerning presentation adaptation, the content of each node or page can be split into two components: the information items presented in the node and their presentation features. Contents can be adapted to the user's knowledge and preferences so that each learner is delivered the information items she needs in each node. Thus, students with different knowledge background will have different views of a same node. Moreover, interaction mechanisms can also be adapted to the learning style and goals. Finally, the presentation features of the contents (e.g. writing style, font, size, color and so on) can also depend on some user characteristics (e.g. age, background, disabilities, language) as well as on the usage platform (e.g. handheld, laptop).

2.2. Role based access control and adaptation

RBAC policies [Sandhu et al., 1994; Ferraiolo and Kuhn, 1992; Ferraiolo et al., 1999] model users by means of roles, which represent jobs or responsibilities. To gather the complexities of most organizations, hierarchies of roles can be defined as a DAG, where general roles are specialized into more specific roles that inherit properties and authorizations from their parents. Each role is granted permission to perform some operations that define the manipulation abilities of a specific kind of user within a specific context. Specific users are assigned a number of roles so that during the application operation each user exercises the abilities specified for the role(s) she belongs to. Such an approach can be applied within the context of educational applications, where a number of roles are easily identified, including the instructor, the learner specialized into novice, intermediate and expert (see Figure 1). Each role has different views of the same course as well as different capabilities to modify its components.

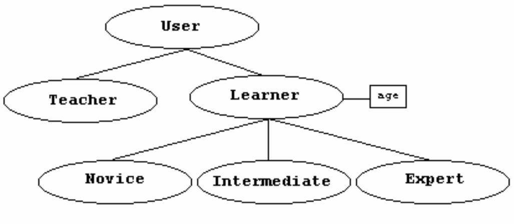


Figure 1: DAG of roles for a course

Even though RBAC is oriented towards modeling security policies, it can be extended to support the adaptive access to the courseware, translating authorizations rules to access rules that will state the contents, navigation facilities and interface of each node for each existing role, using a collaborative-based personalization mechanism. Such rules can be explicitly specified or derived for each role, depending on whether the system is an adapted or adaptive one. The main benefits of this approach are the following:

- Multi-level hierarchies of roles make possible to model complex users structures and reuse general knowledge about some groups of users (parents of the hierarchy) to model more specific subgroups (children). For example, in the DAG of Figure 1, the learners age can be used to adapt the style of presentation of the contents as in CESAR [Aedo et al., 1997] an educational application for helping hearing-impaired children to learn the sign and written languages, where students are classified into three linguistic age groups to decide which kind of information has to be presented: images, signs, text or a combination of media. Since age does not depend on the learner's expertise it is defined at the learner level (see Age assigned to the Learner role) and the access rules derived from this feature will be inherited by the learner children (Novice, Intermediate and Expert).

- Users characteristics and access rules are specified in a gradual and modular way according to the DAG structure, whether using a top-down or a bottom-up approach.

- RBAC policies are really efficient in terms of management since once an access rule is established for a specific role it rarely varies; what changes is the user who holds a role. The lesser the number of assignments to be updated when a rule changes the easier the administration task is and the lesser the potential for having erroneous rules.

3. *Designing personalized web-based courses*

In this section, we present three basic principles applied during the development process of our platform of courses adaptation, called Courba, which were assumed in order to make easier the personalization of web-based courses. These principles are: the separation among the structure, the content and the presentation features; the definition of a useful hyperdocument structure; and, finally, the assumption of an RBAC model to specify the users structure.

3.1. Separation amongst structure, content and presentation features

When the logical structure of the application and the actual contents delivered to the user are merged the resulting hyperdocument becomes harder to maintain as it happens with HTML pages. For this reason, it is highly advisable to clearly separate the structure, made up of information holders (nodes), from the contents to produce cleaner specifications. Moreover, and what is more important for adaptation purposes, this separation makes possible the definition of multi-level access policies where different users obtain different views of the same document [Díaz et al., 2001a]. This latter aspect is illustrated in Figure 2 where a two layers architecture is used to support the hyperdocument access. Information is delivered to the user by the Application Manager module taking into account the users requests as well as the access rules. Thus, users Maria and Eva, who hold respectively the roles Novice and Expert, obtain a distinct result from their requests to visualize page1.html since content2 is an explanation considered unnecessary for an expert student with a good command on the topic treated in that node and, therefore, it is hidden (see the corresponding access rule). Finally, this approach also permits to reuse the same content (content2) in different pages (page1.html and page2.html in the example) without replicating it, improving thus maintenance. Separation between structure and content, proposed in most hypermedia reference models [Tompa 1989; Furuta and Stotts, 1989], has been adopted to support adaptation in educational applications like Tangow [Carro et al., 2001].

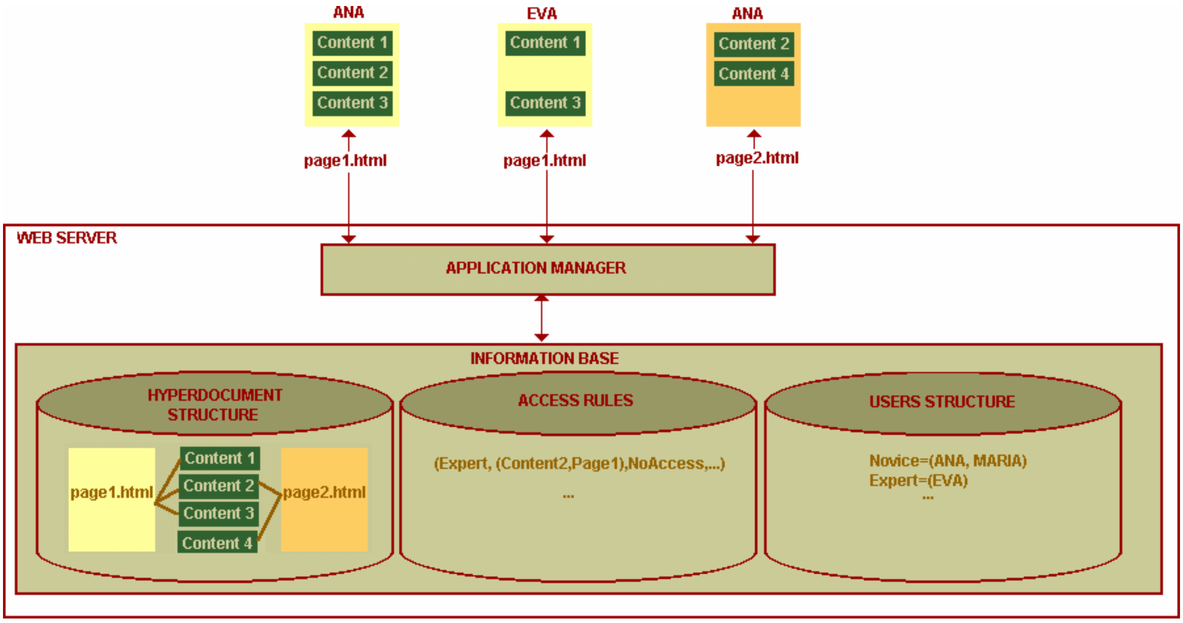


Figure 2: Separating content from structure to support multi-level access policies

Similarly, contents have also to be separated from their presentation features for the same reasons of maintenance, clarity, reusability and adaptation to the user needs. Thus, not only the content but also how it is presented can be dependent on some characteristics of the user or the usage environment with a view of increasing the system accessibility.

3.2. Definition of a useful course structure

Even though the use of hypermedia can be profitable form an educational point of view, there are also some aspects to be taken into account to design a useful hypermedia course. In particular, some empirical studies have shown that freely information browsing is not always the best learning strategy [Sciarone and Meijer, 1993; Aedo et al., 2000]. Indeed, when a learner has a clear learning goal to be met in a specific period of time she tend to prefer some kind of guidance through the information space. Therefore, a basic design concern is to establish which degree of freedom in the hyperstructure navigation will be offered to the users [Hammond, 1993].

In order to help course developers to create a useful structure, Courba provides a template for the learning material represented in Figure 3. This structure defines both the types of learning objects that can be included (the Structural Diagram of figure 3.A) and the kinds of links provided to the users (the Navigation Diagram of figure 3.B). In the figure, the notation of the Ariadne hypermedia design method [Díaz et al., 2001b] is used to represent the learning material structure from both a logical and browsing perspectives.

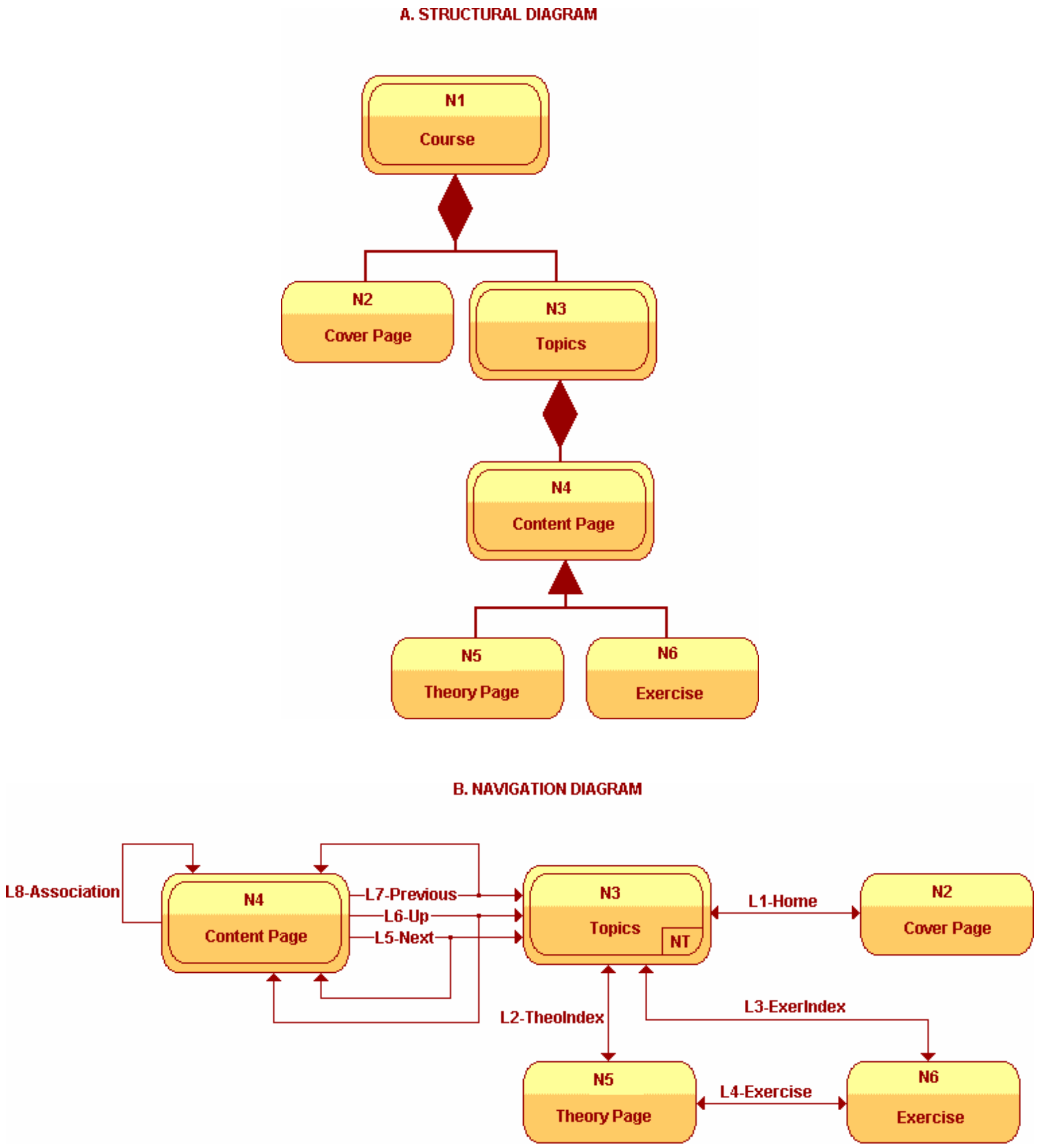


Figure 3: Structural and Navigation Diagram for a Courba course

The logical structure is defined by means of composite nodes (double line rounded boxes) and nodes (single line rounded boxes) both of which are related through structural relationships depicted using UML notation [Booch et al., 1998].

Generalization implies that components inherit the properties, behaviors, links and contents of their parents. As it can be seen in figure 3.A a Courba course it is made up of a series of a cover page and a series of topics, each of which is composed of a number of pages specialized as a theoretical page or an exercise. This structure makes possible to define a consistent user interface, since common elements (e.g. links, contents, headers, footers, background, and so on) are assigned at the composite level and inherited by its components.

Concerning the browsing structure, there are a number of structural links automatically derived by Courba (see links from L1 to L7 in figure 3.B). These links provide a useful navigation guidance to study the whole course following a predefined sequence that can depend on the user. Thus, it can be established that a Novice student requires structural guidance whereas an Expert student does not need it and they can be hidden. In addition, course developers can include associative links (L8 in figure 3.B) to support free browsing in the contents of the node by using the HTML anchor tag. As it can be seen in the figure 3.B, an associative link can be set between any two content pages.

3.3. Use of an RBAC model for hypermedia

An RBAC model, that was proposed for providing safe access to hypermedia applications in [Díaz et al, 2001a], is extended in Courba in order to provide course developers a rational framework to specify the access policy of their courses. Such an access policy is defined in terms of a number of access rules that specify the access category of a subject for a given object and how will the user perceive the object (presentation specifications):

Access rule = (subject, object, access category, presentation specifications)

From this definition, they emerge three elements that need a further explanation: the subject, the object and the access category.

Subjects are those active entities that can execute actions on the objects (e.g. retrieve a node, modify a link or create a new content). According to this model, the subjects are the users of the application, taking for granted that other entities which can activate actions, like programs or scripts, inherit the same access rules that apply to that user activating them. Since an RBAC policy is assumed designers (or course developers in the Courba environment) do not identify specific users but two kinds of abstractions: roles and teams. While a role is an organizational position or job function that appears in the domain of application, a team is just a group of users established whether to represent collaborative group as in [Wang, 1999] or to simplify the administration tasks. Roles and teams support composition mechanisms to be able to deal with complex user structures. To gather the complexities of most organizations, hierarchies of roles can be defined as a DAG [Ferraiolo et al., 1999] where general roles are specialized into more specific roles. This generalization entails inheritance and, therefore, there is a propagation mechanism to establish the characteristics and access rules of the children according to the characteristics and access rules of their parents. Moreover, roles can be aggregated into teams. In this case, the access rule assigned to a team is directly assigned to those members of the team for whom no other rule has been specified or inherited. Figure 4 presents a typical roles and teams structure in Courba. As it can be seen in the figure there are eight roles and two teams. The course is considered as a team of users composed by a coordinator, some instructors, lab groups and learners. An instructor is specialized into lecturer and assistant and the course coordinator is always a lecturer. A lab group is a team of learners working on a specific task, and a learner is specialized according to her background on the course topics into novice, intermediate and expert.

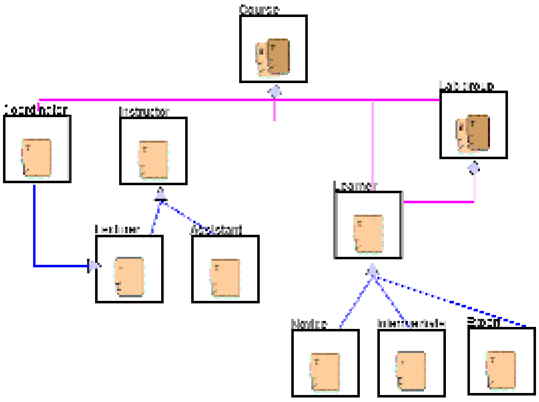


Figure 4: DAG of roles a team in a Courba course (teams are represented with the double face icon whilst roles have a single face icon)

Objects are passive entities that receive the effects of the actions executed by the subjects. All the elements of a hyperdocument are amenable to be considered as objects as far as operations to manipulate them are supported in the hyperdocument. However, in order to maintain a reasonable level of efficiency, both in runtime and during the management process, the object's granularity has to be carefully determined and, for this reason, only nodes (whether simple or composite) and contents are considered as objects. As for the links, they are defined as connections between a source and a target which always resolve into a node or content, and they inherit the rules that apply to the elements making up their definition. For instance, a link is only presented to a user when she can access both its source and its target. An example of object can be the topic "Designing user interfaces" in an HCI course.

Access categories embody the access capabilities for a given hyperdocument or course. Four categories are considered:

- No access: to deny access to a node or content.
- Browsing: to retrieve information whether selecting links or using other mechanisms such as indexes or search engines;
- Personalizing: adds to the Browsing category the ability to include personal elements (such as private contents, nodes or links); and
- Editing: adds to the Personalizing the ability to modify elements of the hyperdocument, that is, to manipulate elements that are accessed by all the hyperdocument users.

For example, for the same object (the "Designing user interfaces" topic) course developers can specify the following two rules to distinguish between the ability to modify the topic contents and the ability to access the topic pages:

(Instructor, Designing user interfaces, Editing,-)
(Learner, Designing user interfaces, Browsing,-)

Access rules can be directly specified by the course developer (*adapted or personalized hyperdocument*) or derived from the users behavior (*adaptive hyperdocument*). Current version of Courba does only deal with personalized courses where access rules are hard coded for the roles so that personalization of the course structure, contents and interface is supported.

4. Courba: a platform to generate personalized web-based courses

Courba is a platform devoted to creating web-based courses that will be accessed following an RBAC policy. With this purpose it provides mechanisms to: establish the contents and structure of the course; determine the roles that will access it; specify the access rules that will be applied during the course operation and assign each user her corresponding role(s). The next subsections we provide some details about Courba implementation and operation.

4.1. Courba architecture and implementation

Courba is made up of a series of modules which make possible to support the personalization of the hypermedia structure, contents and interface to different users roles in a web-based course (see Figure 5). From the course developer point of view, a course will be a well-formed XML document with the desired contents and associative links. This course is automatically paginated and linked by the platform taking into account the access rules. Moreover, when a user asks for a course page to the web server the Application Manager Module will dynamically generate the result including the contents and links specified for that role and applying the corresponding presentation features. Next paragraphs describe the main module s of Courba.

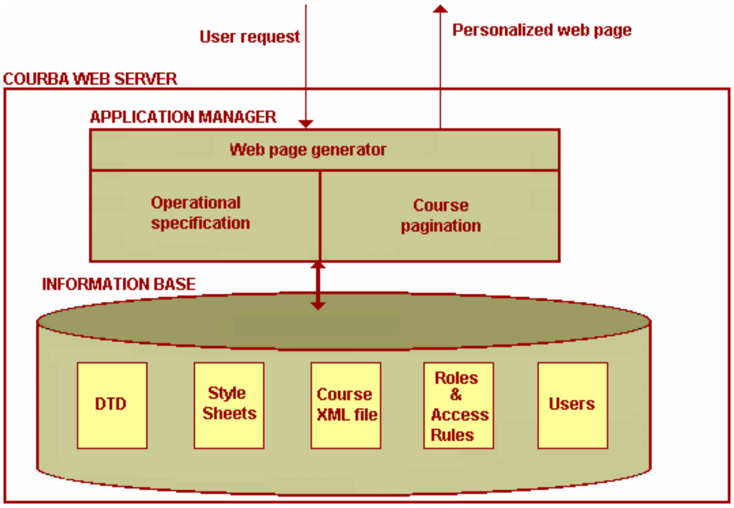


Figure 5: Courba Architecture

The **Information Base** is made up of a number of components that hold information on the course as well as on the users.

First there is a course DTD compliant the logical structure shown in figure 3.A. Table 2 includes part of this DTD where it can be seen that a course has a number of roles, access rules and a set of pages. Each page can be a cover page, index, theory page or exercise. In turn, a theory page has some contents and some operations which can be performed on that page (e.g. go to the parent page).

```
<!ENTITY % accessCategory “NoAccess|Browsing|Personalizing|Editing”>
...
<!ELEMENT course (user_rol+, access_Rule+, page+>
...
<!ELEMENT page (coverPage| index| theory | exercise>
       <!ATTLIST page id CDATA #REQUIRED>
       <!ATTLIST seq CDATA #REQUIRED>
...
<!ELEMENT theory (title_theory, info+, operation*)>
<!ELEMENT title-theory ANY>
       <!ATTLIST title_theory accessCategory (%accessCategory;) “Browsing”>
       <!ATTLIST title_theory code ID #IMPLIED>
<!ELEMENT info (text|image|plugin|operation)+>
       <!ATTLIST info accessCategory (%accessCategory;) “Browsing”>
       <!ATTLIST infocode ID #IMPLIED>
...
<!ELEMENT operation ANY>
       <!ATTLIST operation accessCategory (%accessCategory;) “Browsing”>
       <!ATTLIST operation code ID #IMPLIED>
       <!ATTLIST operation type (%operationType;) #REQUIRED>
       <!ATTLIST operation target CDATA #IMPLIED>
       <!ATTLIST operation view CDATA #IMPLIED>
       <!ATTLIST operation description CDATA #IMPLIED>
...
```

Table 2: Part of a Courba course DTD

The course actual contents are held in an XML document and the information about the presentation features (color, style and so on) is contained in separated style sheets. Finally, there are also maintained the roles that will be considered in the course which are a part of the course DTD as well as the assignment of users to specific roles which is maintained in a separated file. When a user has to be included or removed from a course, only this latter users file is modified and the course while the course DTD, XML document and style sheets remain unchanged. Thus, hyperdocument management becomes easier.

The **Application Manager** is responsible for answering to the user's requests combining the information about the course and the personalization rules. From an implementation point of view, it consists of three modules:

- The *Course pagination* module divides the XML document of the course into a series of pages (or nodes) taking into account the structure of the course represented in figure 3.A.

- The *Operational specification* module executes the manipulation operations that will be supported by the course (see the operation element in Table 2). Current version of Courba supports the navigation operations defined in figure 3.B and it establishes the target of each link (previous, next and parent) according to the role of the user who is accessing the node. Access rules determine which operations will be available for each role. In order to provide more complex operations, such as annotation, the corresponding specification should be included in this module

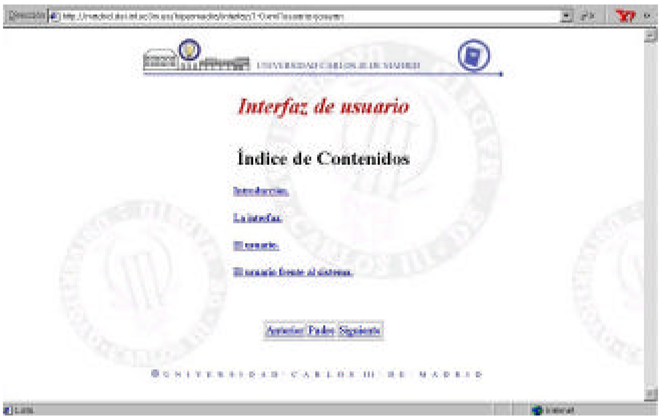
- The *Web page generator* creates the web page corresponding to a users request taking into account the information about the page (structure, contents, interface and operations) as well as the user's role.

The *Application Manager* implementation relies upon an Apache web server to deal with the user requests. In order to increase the course accessibility, information is finally delivered as an HTML page which can be used in any web browser. With this purpose, a Cocoon 1.8.2 module transforms XML documents to HTML using XLS to specify the presentation features. Finally, Apache Jserv 1.1.2.2 is used to execute the operations specified in the *Operational Specification* module.

4.2. Personalized access in a Courba courseware

In order to create a personalized or adapted hypermedia courseware, developers have to create the XML document containing the course components, to specify which roles will be used in their course and which users hold that roles. Finally, they have to establish the access rules for the course components that will be nodes (that is, pages) and contents (that is, each information item – text, image, plug-in or operation-) included in a page. This information will be used by the Application Manager to support personalized access to the courseware and provide each user the information, browsing capabilities and interface which has been established for her role.

The Courba platform here presented has been tested in a course on Hypermedia Design. Figure 6 and 7 show an example of personalized content for two different roles: the novice and the expert learner. Both figures show the dynamic index of topics created by Courba according to the access rules specified for both roles. In figure 6, it can be seen that a novice learner in this subject will have access to four topics that will introduce her into the user interface problems, the interaction styles, the user characteristics to be taken into account and a short introduction to the different interaction models which can be considered to deal with different kinds of users. Once a learner has some experience in user interfaces (see Figure 7), she is also given the opportunity of accessing sections about rules and best practices concerning user interface design and a procedure to design the user interaction based on the hierarchical design of user tasks. In this case, the introduction where basics concepts of HCI are reviewed is hidden. The targets of links Previous ("Anterior" in the figures), Parent ("Padre" in the figures) and Next ("Siguiente" in the figure) are calculated in runtime by the Courba Operational Specification module.


Interfaz de usuario

Índice de Contenidos

Introducción.

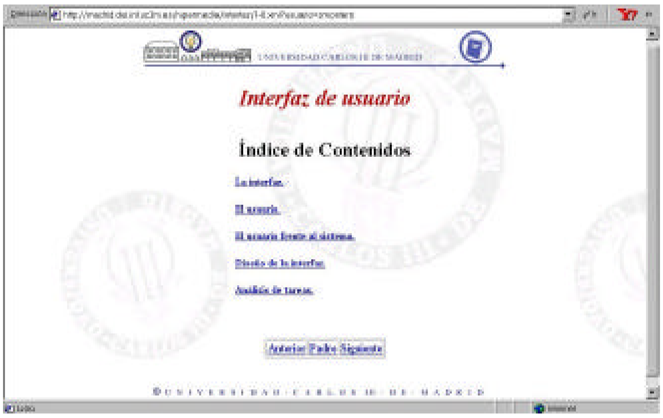
La interfaz.

El usuario.

El usuario frente al sistema.

UNIVERSIDAD CARLOS III DE MADRID
Anterior
Padre
Siguiente
UNIVERSIDAD CARLOS III DE MADRID

Figure 6: Table of contents of the subject “User Interface” for a novice learner


Interfaz de usuario

Índice de Contenidos

La interfaz.

El usuario.

El usuario frente al sistema.

Diseño de la interfaz.

Análisis de tareas.

UNIVERSIDAD CARLOS III DE MADRID
Anterior
Padre
Siguiente
UNIVERSIDAD CARLOS III DE MADRID

Figure 7: Table of contents of the subject “User Interface” for an expert learner

5. Conclusions

In this paper, we have discussed how to combine the basic principles of adaptation for hypermedia applications with RBAC policies in order to support personalized access to a web-based courseware. With this purpose a modular architecture has been proposed where structure, content and presentation features are separated to improve maintenance, reusability and adaptation. Moreover, the assumption of RBAC basic principles makes possible the specification of complex user models, defined by means of multiple level hierarchies that include nested and parallel structures and set up the basis for reutilizing characteristics and access rules of groups of users.

We also presented a platform to support personalization in web courses, called Courba, that has been used in a Hypermedia Design course. One of the main benefits of the proposed approach is that it reduces the maintenance effort, since once roles are identified and their characteristics and access rules are defined they are not likely to change. What changes is the set of users that assume that role and that information is separated from the adaptation rules, so that including, deleting or modifying concrete users does consume few resources.

However, the current version of Courba is quite restrictive and complex to use since no graphical interface is provided for the course developer. We are actually trying to improve the platform to make it more usable as well as to support adaptive courseware, that is, including an intelligent module to dynamically derive the access rules of each role instead of being hardcoded.

Acknowledgements

We'd like to thank Jose Ángel Cruz for his cooperation in the development of Courba. The RBAC model used in Courba is part of a project funded by "Dirección General de Investigación de la Comunidad Autónoma de Madrid y el Fondo Social Europeo(07T/0012/2001)".

6. *References*

[Aedo et al., 1997] Aedo, I., Díaz, P. and Catenazzi, N. A model for hypermedia learning environments based on electronic books Association for Learning Technology Journal (ALT-J), 5(3),. 4-21.

[Aedo et al., 2000] Aedo, I., Díaz, P., Fernández, C., Muñoz, G. and Berlanga, A. Assessing the utility of an interactive electronic book for learning the Pascal programming language. IEEE Transactions on Education **,** 43(4). 403-413.

[Albrecht and Koch, 2000] Albrecht, F., Koch, N. and Tiller, T.. SmexWeb: An Adaptive Web-based Hypermedia Teaching System. Journal of Interactive Learning Research, Special Issue on Intelligent Systems/Tools in Training and Life-Long Learning, 11(3/4), 367-388.

[Allinson and Hammond, 1990] Allinson L. and Hammond N. Learning Support Enviroments: Rationale and Evaluation Computers & Education. 15(1-3) 137-143.

[Barker, 1993] Barker P. Exploring Hypermedia. Kogan Page, London.

[Beltran, 1992] Beltran T. Hypertext, Databases and Computer Aided Instruction: where is the match?. Proceedings of the 3th International Conference on Databases and Experte Systems Applications, 215-220.

[Benamou and Celentano, 1994] Benamou, N. and Celentano, A. Production or Interactive Multimedia Courseware with Mathesis. In Design and Production of Multimedia and Simulation-based Learning Material. Ed. Kluwer Academic Press, 61-82.

[Booch et al., 1998] Booch, G., Jacobson, I. and Rumbaugh, J. The Unified Modeling Language. Addison-Wesley.

[Brusilovsky et al., 1998] Brusilovsky, P., Eklund, J. And Schwarz, E. Web-based education for all: a tool for developing adaptive courseware. Computer Networks and ISDN Systems. 30(1-7), 291-300.

[Brusilovsky, 2001] Brusilovsky, P. Adaptive Hypermedia. User Modeling and User-Adapted Interaction. 11, 87-110,

[Carro et al., 2001] Carro, R. M., Pulido, E. and Rodríguez, P. Improving Web-Site Maintenance with TANGOW by making Page Structure and Contents Independent. In Web Engineering: Managing diversity and complexity of web application development. Murugesan, S. and Deshpande, Y. Eds. Springer LNCS 2016, 325-334.

[De Bra and Calvi, 1998] De Bra, P. and Calvi. L. AHA! An open Adaptive Hypermedia Architecture. The New Review of Hypermedia and Multimedia. 4, 115-149

[Díaz et al., 1998] Díaz P., Aedo I., Torra N., Miranda P. and Martín M. Meeting the needs of teachers and students within the CESAR Training System. British Journal of Educational Technology. 29 (1). 35-46.

[Díaz et al., 2001a] Díaz, P., Aedo, I. and Panetsos, F. Modelling security policies in hypermedia and web-based applications. In Web Engineering: Managing diversity and complexity of web application development. Murugesan, S. and Deshpande, Y. Eds. Springer LNCS 2016, 90-104.

[Díaz et al., 2001b] Díaz, P., Aedo, I., Montero, S.: Ariadne, a development method for hypermedia. Proceedings of Dexa 2001, LCNS 2113, Munich. 764-774.

[Ferraiolo and Kuhn, 1992] Ferraiolo, D. and Kuhn, R.: Role-Based Access Control. Proc. of the 15th National Computer Security Conf. Vol II, 554-563.

[Ferraiolo et al., 1999] Ferraiolo, D.F., Barkley, J.F. and Kuhn, D.R.: A Role-Based Access Control Model and Reference Implementation within a Corporate Intranet. ACM Trans. on Information and Systems Security, 2(1), February, 34-64.

[Furuta and Stotts, 1989] Furuta, R. and Stotts, P. D.. Separating Hypertext Content from Structure in Trellis. Hypertext: State of the Art.

[Hammond, 1993] Hammond N. Learning with Hypertext: Problems, Principles and Prospects. Eds. McKnight, C., Dillon, A. y Richardson, J. "Hypertext a psychological perspective". Ellis Horwood (New York), 51-70.

[Nielsen, 1995] Nielsen, J. Multimedia and Hypertext: the Internet and beyond. Academic Press, Inc.

[Quentin-Baxter and Dewhurst, 1992] Quentin-Baxter M and Dewhurst D. A method for evaluating the eficency of presenting information in a hypermedia enviroment Computers & Education 18 (1) 179-182.

[Sandhu et al., 1994] Shandu , R.S., Coyne, E.J., Feinstein, H.L. and Youman, C.E.: Role-Based Access Control: A Multi-Dimensional View. Proc. of the Tenth Annual Computer Security Applications Conference, Orlando, Florida, 54-62.

[Sciarone and Meijer, 1993] A.G. Sciarone and P.J. Meijer. How free should students be? A Case from call: computer-assisted language learning. Computers & Education. 21 (1-2). 95-101.

[Tergan and Lechner, 2000] Tergan, S.O. and Lechner, M. HyperDisc: A Hypermedia-Information System for Supporting Advanced Studying Design Rationale, Application, and Some Research Results. Journal of Educational Multimedia and Hypermedia.9(3), 207-222.

[Tompa 1989] Tompa, F. A Data Model for Flexible Hypertext Database Systems. ACM Transactions on Information Systems, 7(1). 85-100.

[Wang, 1999] Wang, W. Team-and-role-based organizational context and access control for coooperative hypermedia environments. Proc. of Hypertext'99, Darmstad, 37-46.

[Weber and Specht, 1997] Weber, G. and Specht, M.. User modeling and adaptive navigation support in WWW-based tutoring systems. Proceedings of User Modeling '97. 289-300.

[Zukerman and Albrecht, 2001] Zukerman, I. and Albrecht, D.W. Predictive Statistical Models for User Modeling. User Modeling and User-Adapted Interaction. 11, 5-18.