

Towards a generic conceptual design meta-model for web-based educational applications

SYMEON RETALIS

University of Cyprus
and

ANDREAS PAPASALOUROS AND MANOLIS SKORDALAKIS

National Technical University of Athens

The WWW has turned into a development and run-time environment for large-scale and complex applications among which are the educational applications. There are now justifiable research and development efforts that attempt to formalize the engineering process of such applications in order to achieve better learning effectiveness as well as development efficiency. A lot of work has been done in the field of the instructional designing rather than on hypermedia designing of such applications. Most of the design methods that have been used are general purpose hypermedia design methods which do not allow much in either the conformance of the design models to the latest learning technology standards, or in their seamless translation to standard data formats. This paper presents an attempt for facilitating the conceptual design of web-based educational application by meta-modelling their underlying content and navigational structure. This work is based upon and also extends the OOHDM principles. The Unified Modeling Language (UML) serves the purpose of notation syntax and semantics for this meta-model. The theoretical analysis of the model is accompanied by relevant examples.

Categories and Subject Descriptors: I.7.2 [**Document and Text Processing**]: Document Preparation-*Hypertext/hypermedia*; K.3 [**Computing Milieux**]: Computers and Education

General Terms: Design, Theory

Additional Key Words and Phrases: Web-based educational application, hypermedia design method, conceptual design model

Authors' addresses: Symeon Retalis, Department of Computer Science, The University of Cyprus, P.O. Box 20537CY-1678 Nicosia, Cyprus; email: retal@cs.ucy.ac.cy; Andreas Papasalouros and Manolis Skordalakis, Department of Electrical and Computer Engineering, National Technical University of Athens, 15780 Zografou, Athens, Greece; email: andpapas@softlab.ntua.gr, skordala@softlab.ntua.gr.

1. INTRODUCTION - MOTIVATION

The ubiquity of the Internet and the World Wide Web in all the fields of the new knowledge-based economy is setting new demands and requirements for the engineering of hypermedia applications. The evolution from hand-crafted personal web-pages to the multi-billion market of complex e-learning, e-commerce and e-government applications has happened too fast and is not to be taken lightly while developing such applications. Regretfully, arbitrary, anarchic approaches and ad hoc methodologies that were originally used for web site development, still dominate the current state of application development in the Web environment. It is more than evident that web sites cannot be designed and implemented like they used to be. Instead these trial-and-error approaches, must relinquish to methodic and systematic engineering approaches for hypermedia development (Lowe and Hall [1999], Murugesan [1999]). Therefore the construction of high quality hypermedia applications within specific time and fund limits inflicts the need for development methodologies.

In contrast to generic software engineering, where significant progress has been made the past twenty years, there is still a great deal of work to be done on formalizing process models, and defining methodologies or design methods for hypermedia applications (Lowe and Hall [1999]). In this paper we do not intend to tackle the hypermedia development process as a whole but rather to focus on the design phase by presenting a simple design method, giving emphasis on the conceptual design of web-based educational applications. We present an object oriented meta-model which can serve as a guide or a pattern for constructing a web-based educational application (WbEA) conceptual design.

This model provides sufficient means to create complete, flexible, well-structured and easily maintainable design blueprints of the WbEA. This model expresses an overall structure of the resources that comprise a WbEA as well as their semantic relationships.

The requirements that this meta-model had to meet were (Koper [2001]):

1. *Formalisation*: its notation system must describe a WbEA and its constituents in a formal manner
2. *Completeness*: its notation system must be able to fully describe a WbEA, including all types of its constituents, the relationships among them and their behavior
3. *Reproducibility*: its notation system must describe a WbEA and its constituents in an abstract level so that repeated execution/adoption is possible for specific subject domains
4. *Compatibility*: its notation system must fit in with the available standards and specifications (IMS, IEEE LTSC, SCORM, etc.)
5. *Reusability*: its notation system must make it possible to identify, isolate, decontextualize, exchange and re-use constituents of a WbEA.

The rest of the paper is structured as following: In section 2, an overview of the three steps in the design process are described, followed by an analysis of the meta-model used for the conceptual design in section 3. An instantiation of the meta-model is presented through a case study of a undergraduate course on the subject domain of "Compilers" in section 4. In section 5 relevant work and is presented. Finally, section 6 contains some concluding remarks.

2. CADMOS-D: A HYPERMEDIA DESIGN METHOD

CADMOS-D (design) is a method for the creation of the detailed design of a web-based educational application, which includes structural details of the learning resources, the navigational schema and templates for describing abstractly the graphical user interfaces.

This method follows the principles of the object oriented hypermedia design method (OOHDM) (Schwabe and Rossi [1995], Schwabe and Rossi [1998]), which has provided systematic ways to design generic hypermedia applications and not especially educational ones.

CADMOS-D method, which belongs to a web-based educational application (WbEA) development methodology—named CADMOS— described in Retalis and Skordalakis [2001], proposes a stepwise design process, as shown in Fig. 1: Conceptual Design, Navigational Design and Interface Design. The intermediate products of each step are validated according to guidelines for formative evaluation of the instructional design (checking structural, navigational, aesthetics and functional issues). The whole design process is considered to be iterative, where in each iteration loop the design artefacts are evaluated and the feedback from the evaluation is used for their improvement, until they reach the desirable level.

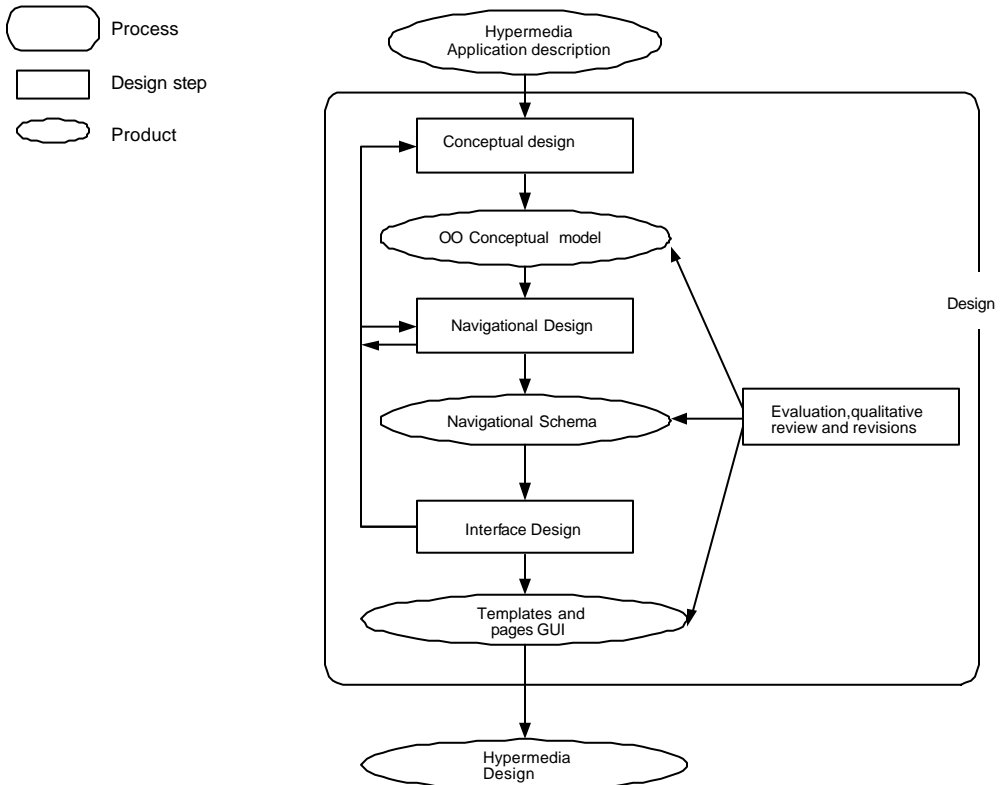


Fig. 1. – The three design steps.

2.1 Conceptual Design

In this step, the description of the WbEA is transformed into a conceptual design following an object-oriented conceptual design meta-model. According to this meta-model, a WbEA is a *mosaic* of learning resources, such as hierarchically arranged sets of pages of an electronic book, web testing resources, on-the-fly pages, site maps, search engines, etc. For each type of resource, the designer must specify the elements that comprise it, such as media elements, active behavior, etc. Active behavior might mean either *static publishing* (i.e. delivery of content through a set of structured linked media) or *dynamic publishing*, meaning that the delivery of content can be made via “on-the-fly”

web pages (e.g. queries to databases) generated in the context of a server-side web application environment (e.g. CGI, PHP, ASP, JSP, ColdFusion) or Learning Management System. In order to serve the needs of the object-oriented data model, the design method has adopted the UML notation. In particular, conceptual design adopts *use case diagrams* and *class diagrams* from the UML repository. Use case diagrams depict the ways that the WbEA is used by external *actors*, or in other words specify the system functional requirements. Class diagrams contain object-oriented constructs such as classes, interfaces, packages (grouping mechanisms), and various relationships between them and aim at specifying in detail the hypermedia application structure at an abstract level. For facilitating the construction of the domain specific class diagrams, CADMOS-D has proposed an abstract object oriented meta-model which is independent of underlying subject domain of the application (i.e. mathematics, geography, etc.) but provides a suitable platform to describe structural and navigational issues of the learning resources.

2.2 Navigational Design

In this step the navigational schema of the WbEA is analytically designed, so that it is clearly specified how web pages are inter-connected with hyperlinks. The data model of the navigational design contains web pages, single and bi-directional hyperlinks. The navigational design provides a way of checking the implementation of all the hyperlinks in the final product. More importantly it facilitates the maintenance of the web site, especially when web pages are added or deleted and hyperlinks to and from them have to be updated. In this way, the well-known problem of 'dangling' links can be avoided. The navigational structures proposed for this kind of design, are well accepted by many hypermedia design approaches, such as HDM (Garzotto et al. [1993], Garzotto et al. [1996]), RMM (Isakowitz et al. [1995], Isakowitz et al. [1997]) and OOHDM (Schwabe and Rossi [1995], Schwabe and Rossi [1998]). More specifically they are: a) *indices* that provide direct access to every indexed node, b) *guided tours* which are linear paths across a number of nodes and c) *indexed guided tours* which combine the two previous structures.

2.3 Interface Design

In this step, the Graphical User Interface (GUI) of the hypermedia application is designed, that is the content, layout and 'look and feel' of the web pages. WbEA interface design is ruled by the principles of the *page metaphor*, a practice taken from multimedia engineering where it has been extensively adopted and used. Page metaphor is used to specify the page components with graphic symbols and deploy them on the screen showing their layout. Therefore, with the use of graphical semantics, the design depicts the page form just as it will be implemented. The data model for the interface design contains six kinds of page components: plain text, multimedia elements, active elements, hyperlinks, frames and forms elements. The designs made are actually re-usable page templates. For instance, if we design the page template of one paragraph of an on-line book in a hypermedia application, then all the other paragraphs of the book might have the same look, using the same components with the same layout, have the same frames etc. During the interface design, except for designing page components and their layout, we define certain metadata on them. All page components can have metadata that describe various aspects of them, like author details, type or format etc. The definition of metadata during the design phase is of paramount importance as it facilitates the management of the page resources and their accessibility and reusability.

3. SPECIFICATION OF THE META-MODEL

In the present section we present an object-oriented meta-model for constructing a WbEA conceptual design. This meta-model aims to help the designers in structuring the web-based learning resources, representing the subject matter in various hypermedia modalities, rendering it clear and accessible to the learning audience as well as and conforming to learning standards and specifications.

The design meta-model proposed in this paper defines an extension of UML (Booch et al. [1999]) suitable for modelling WbeA. UML provides explicit, built-in extension mechanisms in order to render the specific semantics of elements belonging to domains other than typical software systems design, namely stereotypes, and tagged values. Stereotypes are adornments applied to existing UML elements altering their semantics, thus extending the vocabulary of the language. Stereotyped elements can be presented with symbols (or icons) different from the UML standard notation icons, thus extending the visual vocabulary of the language. Tagged values are name-value pairs that provide additional information (in the form of tags) to model elements.

This meta-model provides a guide in constructing a detailed conceptual schema of the WbEA by means of a hierarchy of nodes of information corresponding to certain types of resources. This hierarchical structure of hypermedia content emanates from Dexter model for hypermedia systems (Halasz and Schwartz [1994]) and is also common in other hypermedia application models proposed in Ferreira de Oliveira et al. [2001] and Garzotto et al. [1993]. It also provides details of navigational aspects in terms of links between nodes. A concrete conceptual design model of the WbEA on a specific subject domain that is based on our meta-model should be considered as its instance. The underlying ideas, we had to elaborate, are similar to other educational models, such as the Educational Modelling Language (EML) [<http://eml.ou.nl>]:

1. Classify, or type the learning resources and their constituents of a WbEA in a semantic network
2. Build a containing framework expressing the relationships between the typed resources and their constituents
3. Define the structure for the content and behaviour of the different types of typed resources and their constituents

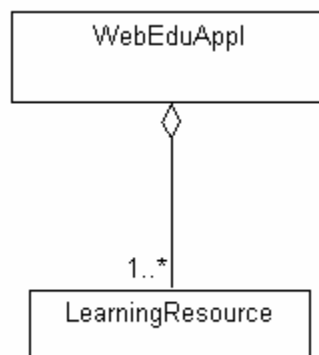


Fig. 2. Web-based learning resources.

The principles of this meta-model are the following:

- A WbEA is composed of learning resources (Fig. 2). Learning resources are organizational units of learning content each with specified learning objectives (Fig. 3).

Each resource contains either a number of web pages or other learning resources hence organizing the WbEA in a hierarchical structure. This structure conforms to learning content packaging standards like IMS [2001] and ADL [2001], so it is possible to automate the process of mapping the model artefacts to one of the existing learning technology standards. While learning resources are conceptual elements that allow the structure of the WbEA, they can have visual presentations by means of *access pages* (see below) that contain links to actual content (content pages, and web testing resources). Web pages include *atomic elements*, which are multimedia elements (audio, video, image or simple text), active elements (applets, ActiveX controls, Flash objects), client scripting code and form elements (Fig. 5). They can also contain navigational (structural) links (e.g. forward, backward, back to the contents, etc.) It should be mentioned that the navigational links implement structural relationships between the pages of the pages of the courseware and are subject matter independent. For each link, its name and the linked web page should be noted. Finally, each learning resource is (optionally) associated with a *metadata* element (LTSC [2001]), containing information about the resource.

The web pages of the learning resources are divided into the following types, as illustrated in Fig. 5:

1. *Access Pages*, whose learning content is minimal, used for navigational purposes since they provide access to web pages with learning material. From one access page the user can also navigate to another Access Page (as often happen with the contents of the hypermedia book). As mentioned above, access pages usually are visual instances of learning resources containing links to the learning resources and other pages contained into the corresponding learning Resource.
2. *Content pages*, which include informative or learning material. These pages are either “hard coded” (e.g. HTML files) or created on-the-fly, as in the case of .asp pages. So a specification at a lower lever must be made.
3. *Web testing resources*, which are web pages that have a unique structure and functionality in the learning process. They contain open or close type of questions. These resources could be created and maintained (graded, updated, etc.) by the assessment tools of a learning technology system. Of course, they could be stand-alone hypermedia applications. Web testing resources are sets of questions. Each question resides in a content page and is associated with a response element, as shown in Fig. 4, which has information needed from the question and test system in order to handle the question. The detailed data model as well as the types of the questions, are derived from the Question and Test Interoperability Standard of the IMS consortium (IMS [2002]).

- A *web page* contains *anchors* which are *realisations of relationships* among the web pages of learning resources. Anchors can be either static navigations links or dynamic links that activate a process that create on-the-fly web pages, as is the case of submitting queries to a database which is followed by the results' dynamic publishing.

- A *web page* includes atomic elements, namely multimedia elements, active elements, links and code in a mark-up language (e.g. HTML). Moreover, each one of these elements contains embedded links which are subject specific and do not depend on the general web-based courseware navigational schema. For some kinds of multimedia elements, the designer must specify the software system (player) that will handle its presentation, such as RealAudio or RealVideo, or a special plug-in. In Fig. 5, the abstract structure of atomic elements, where media elements are illustrated using a class diagram.

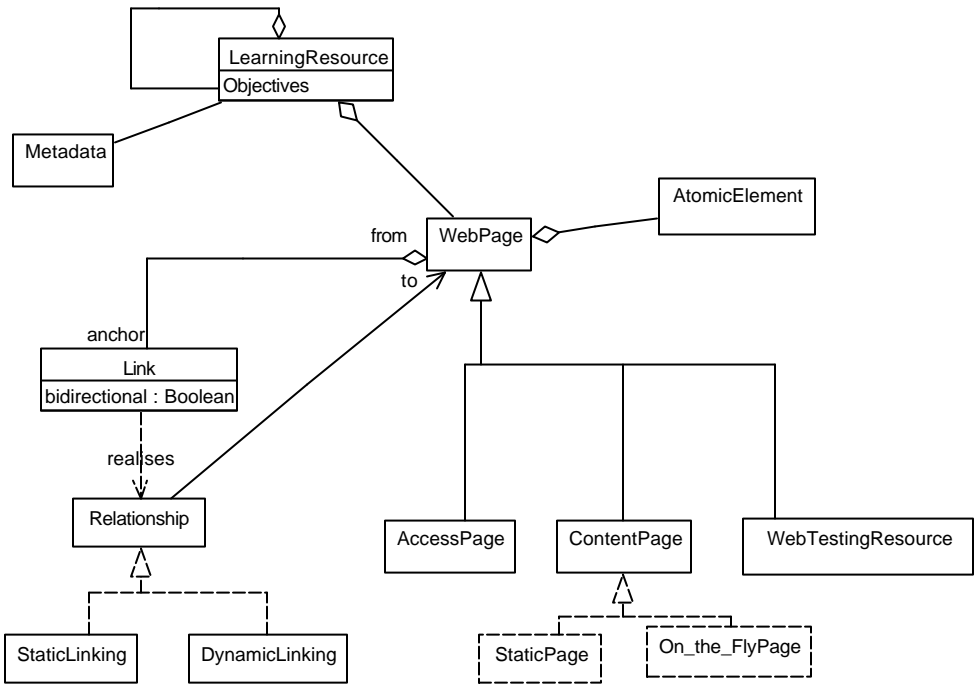


Fig. 3. Learning Resources detailed meta-model.

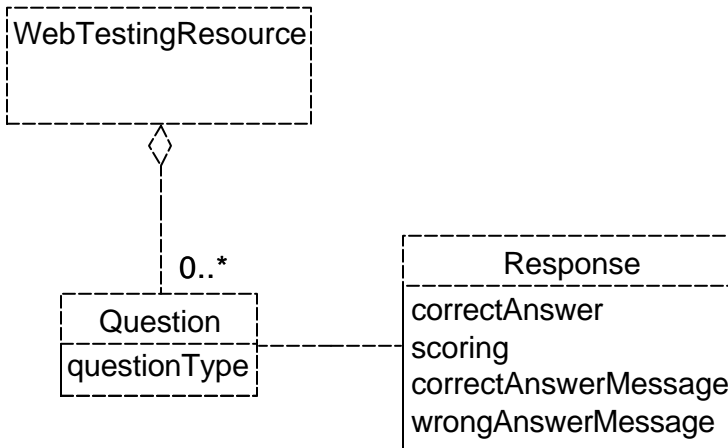


Fig. 4. - Structure of web-testing resources.

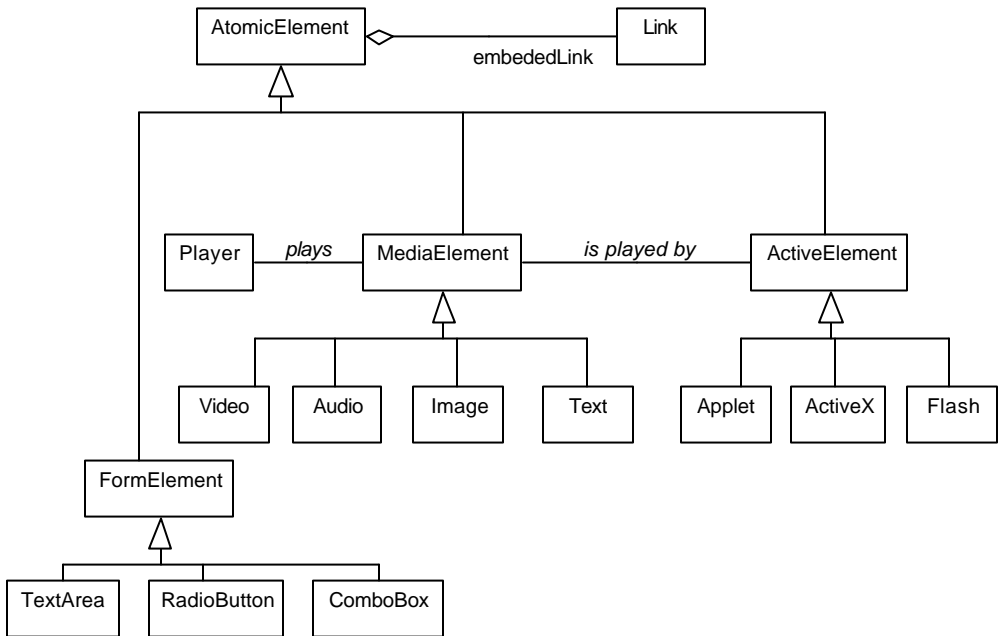


Fig. 5. Structure of the web pages' atomic elements.

- Finally, for specifying the nature and behaviour of the active elements contained in a web page, the designer should give details of the data model of language that will be used for its development as well as the behaviour of the element. For example for a Java applet the Java data model should be used. For a simple CGI script a transition diagram will be enough to show its behaviour and guide the developers in its implementation.

This model was conceived to assist designers in their effort in order that a detailed conceptual design model of their application on a specific subject domain can easily be constructed. So, the designer's main task is to reduce the abstraction level of the generic meta-model presented so far by completely specifying the classes and their relationships of concrete models conforming to it.

As mentioned before, the elements of this meta-conceptual schema, are not domain specific but are themselves hypertext-oriented, much like the concepts of 'component' and 'link' in Dexter model. This approach does not circumvent the principle of separation of concerns in Conceptual and Navigational design steps but rather considers the Conceptual Design step as an early step where the learning material is structured into learning resources in a hierarchical manner. These decisions on the organization of the educational hypertext are further elaborated in the Navigational Design step constructing the Navigational Model. Thus, decisions made in Conceptual and Navigational Design are not orthogonal, but are related with each other, differing on the degree of focus on either organization or navigational issues. This principle of tight coupling between Conceptual, Navigational, Interface Design and Implementation makes elements of the early conceptual model traceable back from, even from implemented WbEA allowing the "reverse engineering" of existing WbEA. Furthermore, even concepts that naturally correspond to the implementation phase such as "Static" and "Dynamic" pages and links appear in the conceptual model but they are actually implemented later, during the implementation phase.

4. AN EXEMPLAR APPLICATION

In this section we present some parts of the detailed conceptual design of the web-based educational application that was developed for the course “An Introduction to Compilers” offered by the Software Engineering Lab of the National Technical University of Athens. This application includes a variety of learning resources such as: a hypermedia didactic book, two case studies, self-assessment web based questionnaires, a collection of past exam papers, a course description, and a study guide. Because of space limitation the illustration of the utilisation of the object-oriented meta-model will be concentrated in one of the case studies. It contains learning material for giving the students a complete example on how to create a compiler for an educational language called “Russel”. This case study has the structure of a hypermedia didactic book. It also contains active elements that are being used by learners to test the validity of already prepared examples as well as their own intermediate products during the development process of a compiler (e.g. lexical analysis). The resources have been developed using HTML and CGI scripts and have been stored on a web server that hosts the WebCT Learning Management System [<http://webct.softlab.ntua.gr>]. The language of the learning material in the current implementation is Greek and so we avoid showing the real interfaces (it will look Greek to you).

In Fig. 6 we present the structure of the WbEA. Four learning resources, each one containing a set of web pages, both access and content pages have been developed.

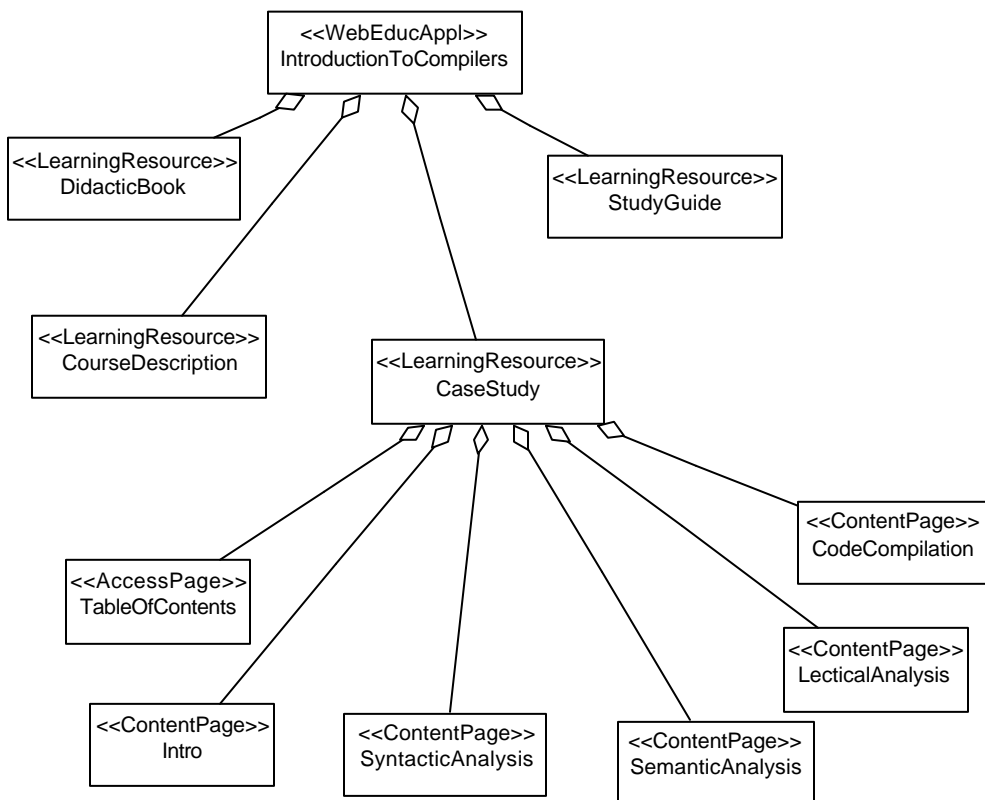


Fig. 6. Structure of the WbEA 'Introduction to Compilers'.

In the following we will concentrate on showing the detailed design of the resource *Case Study*. For simplicity reasons, only four of the actual 27 content pages contained into the current learning resource are shown. An access page named *Table Of Contents* is shown containing links to the content pages of the current learning resource. As Fig. 7 shows, the *Table Of Contents* contains anchors that realise both static and dynamic linking.

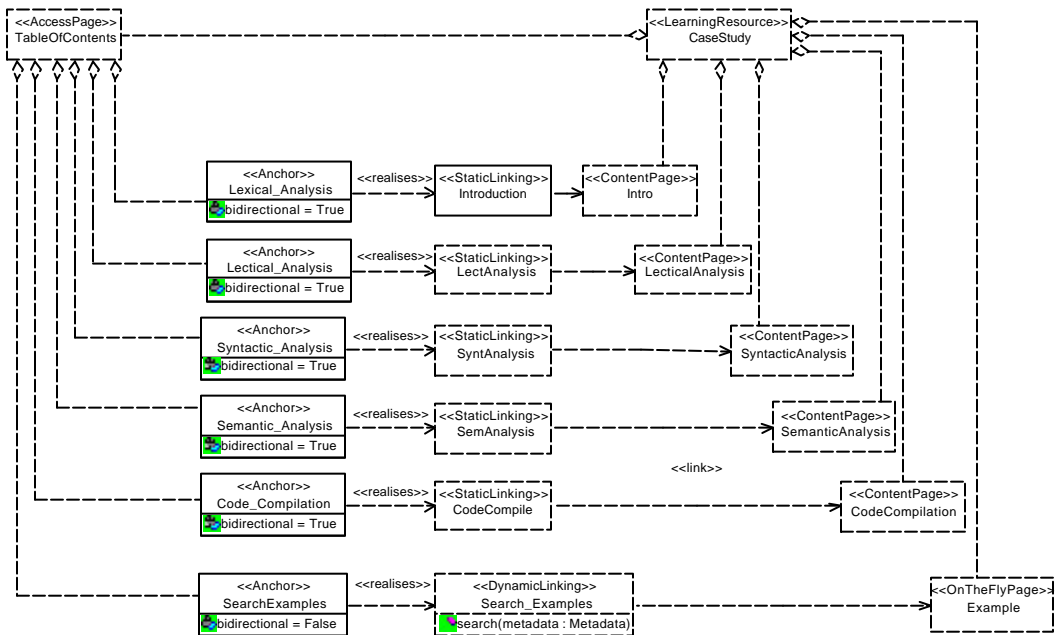


Fig. 7. Structure of the learning resource ‘Case Study’.

Fig. 8 shows a particular page of the resource *Case Study* that has text and a form element, and it is used for validating the intermediate products during the development process of the “Russel” compiler.

Though it is not obvious from the simple previous example, there may exist nesting structures of learning resources of an arbitrary depth. We propose a recursive process of identifying the learning resources of a particular WbEA until the definition of the web pages and their contained atomic elements, which are the finer granularity elements in this model.

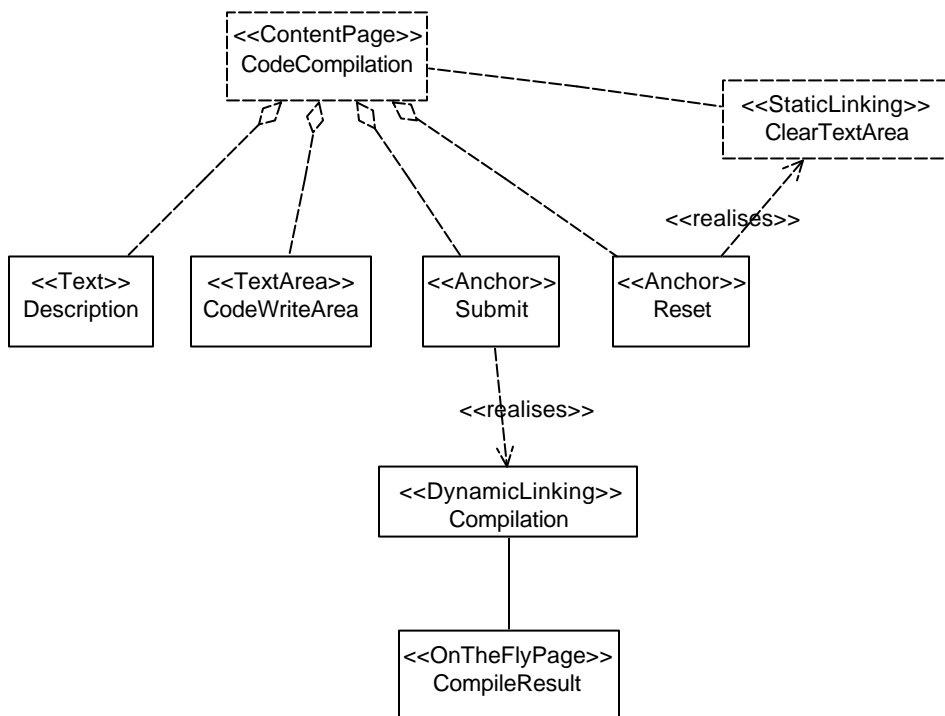


Fig. 8. The structure of a page with a form.

5. RELATION WITH OTHER MODELS AND METHODS

The conceptual meta-model we propose is based on or influenced by previous established models for web and hypermedia engineering.

The Dexter Hypertext Reference Model (Halasz and Schwartz [1994]) proposes three layers of hypertext organization: *Runtime Layer*, defining the dynamic aspects of interacting with hypertext, *Storage Layer*, where the actual nodes and links of hypertext are represented and the *Within Component Layer* where the internal content of hypertext nodes is described. The hierarchical structure of hypertext nodes in the Storage Layer is particularly appropriate in the modeling of WbEA and is adopted in our model.

HDM (Garzotto et al. [1993], Garzotto et al. [1996]) provides a model for high-level hypertext design. Like Dexter, model preserves the hierarchical structure of hypertext nodes but in addition supplies domain specific concepts, namely *Entities* and *Components*, so facilitating the definition of a conceptual *schema* of a hypertext application. The application itself is considered as an implementation of the previous schema and thus the model provides separation between conceptual design and implementation. In the same manner our model provides a conceptual organization of the educational through structuring of learning resources. The implementation of the previous organization in terms of navigation structure, user interface layout and actual content creation is taking place in later steps than the conceptual design, described in this paper.

OOHDM, described in Schwabe and Rossi [1995] and Schwabe and Rossi [1998] clearly proposes the separation between Conceptual, Navigational and User Interface

Design steps in the development of hypermedia applications, suggesting certain types of models for each step. We follow this separation in the process of designing WbEA, though in the present paper we only deal with conceptual design.

Conallen's approach (Conallen [1999]) introduces a UML extension for web application architecture modeling. While web-site modeling is more implementation oriented, introducing the web page as the core modeling element, a purely conceptual approach is adopted for the modeling of the business logic implemented in server side components. Unlike our model, it refers to generic web based applications and not particularly educational ones.

Hennicker and Koch [2001] also extend UML for generic web application modeling. Unlike Conallen [1999], they follow the distinction between conceptual, navigational and static presentation modeling.

Süß et al. [1999] provide a UML extension by means of a meta-model for teachware management. Their meta-model can be separated into two sections, one concerning a conceptual model and one concerning a navigational model, implementing the previous one. They have also developed an XML based language, LM²L that implements their meta-model. User Interface design as well as personalization of teachware is achieved by defining XSL stylesheets and applying them on LM²L files.

EML (Koper [2001]) is not a strict educational hypermedia concept, but it defines a formal, XML based, language for modeling different aspects of the educational process e.g. activities, peoples' roles, content, etc, concerning specific lessons or courses called "units of study". It is a high level model aiming at facilitating instructional design in the context of e-learning.

Finally, in Papasalouros and Retalis [2002] we propose an adaptive extension of our model that can serve both as a design model and a reference model of adaptive educational hypermedia applications. This extension introduces the Object Constraint Language as a means of applying adaptation rules to UML based educational hypertext models. It also utilizes the principles of object oriented design in order to achieve effective User modeling, a very important process in Adaptive Educational Applications.

6. DISCUSSION AND CONCLUDING REMARKS

This paper presents an object oriented meta-model that provide guidelines for creating the conceptual design of a web based educational application. The proposed approach for meta-modeling has a lot of advantages:

- It provides a rich semantic expression for the resources that comprise a WbEA, thus supporting designers in building valid conceptual models of WbEA
- It facilitates the identification of various types of learning resources or their atomic elements that can be re-used and/or easily maintained

Concluding, we should mention that his work is still evolving. There are a number of activities that have been initiated, among which are the following:

- The development of a CASE tool, named CRITON, to support CADMOS-D design method. It combines three design steps in an integrated environment with using the object-oriented meta-modeling language for the conceptual design (Avgeriou et al [2002]).
- To experiment in depth with complex WbEA such as adaptive educational hypermedia applications. In this direction, we are designing a development and run time environment for the creation and delivery of adaptive web based educational applications.
- To introduce an XML-based language to syntactically represent a conceptual design model.

REFERENCES

- ADL 2001. Sharable Content Object Reference Model. <http://www.adlnet.org/>.
- AVGERIOU, P., RETALIS, S., AND SKORDALAKIS, E. 2002. Criton: A Web-based Courseware Design Tool. *Multimedia Tools and Application* (to appear).
- BOOCH, G., RUMBAUGH, J., AND JACOBSON, I. 1999. *The UML User Guide*. Addison-Wesley, Reading, MA.
- CONALLEN, J. 1999. Modelling Web Application Architectures with UML. *Communications of the ACM*, 42, 10, 63-70.
- FERREIRA DE OLIVEIRA, M. C., SANTOS TURINE, M. A., AND MASIERO, P. C. A Statechart-Based Model for Hypermedia Applications. *ACM Transactions on Information Systems*, 19, 28-52.
- GARZOTTO, F., SCHWABE, D., PAOLINI, P. 1993. HDM- A Model Based Approach to Hypermedia Application Design. *ACM Transactions on Information Systems*, 11, 1-26.
- GARZOTTO, F., MAINETTI L., AND PAOLINI, P. 1996. Navigation in Hypermedia Applications: Modeling and Semantics. *Journal of Organizational Computing and Electronic Commerce*. 6, 3 , 211-238.
- HALASZ, F., AND SCHWARTZ, M. 1994. The Dexter Reference Model. *Communications of the ACM*, 37, 2, 30-39.
- HENNICKER, R., KOCH, N. 2001. Systematic Design of Web Applications with UML. In *Unified Modeling Language: Systems Analysis, Design and Development Issues*, K. SIAU AND T. HALPIN, Eds. Idea Group Publishing, Hershey, PA, 1-20.
- LTSC 2001. Draft Standard for Learning Technology Systems Architecture (LTSA), Draft 9. <http://ltsc.ieee.org/>.
- IMS 2002. IMS Question & Test Interoperability Specification- Best Practice and Implementation Guide, version 1.2 – Final release. <http://www.imsproject.org/>.
- IMS 2001. IMS Content Packaging, - Best Practice and Implementation Guide, version 1.1.2 - Final Release. <http://www.imsproject.org/>.
- ISAKOWITZ, T., STOHR, E., AND BALASUBRAMANIAN, P. 1995. RMM, A methodology for structured hypermedia design. *Communications of the ACM*, 38, 8, 34-48.
- ISAKOWITZ, T., KAMIS, A., AND KOUFARIS, M. 1997. Extending the capabilities of RMM: Russian Dolls and Hypertext. In *Proceedings of the Thirtieth Annual Hawaii International Conference on System Sciences (HICSS'97)*, Maui, Hawaii, Jan. 6-10.
- KOPER, R. 2001. Modeling Units of Study from a Pedagogical Perspective: the pedagogical meta-model behind EML. <http://eml.ou.nl>
- LOWE, D., AND HALL, W. 1999. *Hypermedia & the Web, an Engineering Approach*. John Wiley & Sons, West Sussex, England.
- MURUGESAN, S. 1999. Editorial. *ACM SIGWEB Newsletter*, 3,3.
- PAPASALOUIROS, A., AND RETALIS, S., 2002. Ob-AHEM: A UML-enabled model for Adaptive Educational Hypermedia Applications. *Interactive educational Multimedia* , 4, 76-88.
- RETALIS, S., SKORDALAKIS, E. 2001. An Approach for Solving Instructional Problems Using Networked Technologies. *Computers in Human Behavior*, 17, 4, 393-407.
- SCWHABE, D., AND ROSSI, G. 1995. The Object-Oriented Hypermedia Design Model. *Communications of the ACM*, 38, 8, 45-46.
- SCHWABE, D., AND ROSSI, G. 1998. "An Object Oriented Approach to Web-Based Application Design", Theory and Practice of Object Systems 4(4), 1998. Wiley and Sons, New York, ISSN 1074-3224)
- SÜB, C., FREITAG, B., AND BROSSLER P. 1999. Meta-modeling for Web-based Teachware Management. In *Conceptual Modeling: ER '99 Workshops on Evolution and Change in Data Management, Reverse Engineering in Information Systems, and the World Wide Web and Conceptual Modeling*, Paris, France, November 1999, P. CHEN, D. EMBLEY, J. KOULOUMDJIAN, S. LIDDLE, J. RODDICK, Eds. Springer.