Learner-centred adaptation of interactive courseware

Margherita Antona\textsuperscript{1}
Anthony Savidis\textsuperscript{1}
Zacharias Protogeros\textsuperscript{1}
Constantine Stephanidis\textsuperscript{1,2}

\textsuperscript{1} Institute of Computer Science  
Foundation for Research and Technology - Hellas  
Science and Technology Park of Crete  
GR-71110, Heraklion, Crete, Greece  
cs@ics.forth.gr

\textsuperscript{2} Department of Computer Science  
University of Crete

Abstract. This paper focuses on the notion of content and interaction adaptation in web-based courseware. The adopted perspective is that adaptation should be learner-centred, i.e., it should continuously deliver, at any time during the use of the system, the most appropriate interactive learning content to each individual learner. To this purpose, the paper briefly reviews the most important adaptation techniques developed and applied in the context of interactive learning technologies, with particular emphasis on those more relevant in the context of web-based courseware. Subsequently, it discusses the two main sources of knowledge necessary for achieving adaptation, namely the learner model and the course model, and on relevant standard proposals for modelling. Finally, the paper discusses some important aspects of a proposed architectural framework suitable for supporting content and interaction adaptation in web-based courseware.

1 Introduction

In recent years, the concept of adaptation has been investigated in the broad perspective of providing built-in accessibility and high interaction quality in applications and services in the emerging Information Society [Stephanidis, 2000; Stephanidis, 2001a]. From this point of view, adaptation is intended as a proactive strategy for coping with the numerous dimensions of diversity which are intrinsic to the Information Society, and become evident when considering the broad range of user characteristics, the changing nature of human activities, the variety of contexts of use, the increasing availability and diversification of information and knowledge sources and services, the proliferation of technological platforms, etc. Adaptation characterises software products that automatically adapt themselves according to the individual attributes of users (e.g., mental / motor / sensory characteristics, preferences), and to the particular context of use (e.g., hardware and software platform,
environment of use). In this respect, adaptation concerns the interactive behaviour of applications and services as well as the content.

Long distance computer-based instruction, and in particular web-based instruction, is progressively acquiring a fundamental role in shaping new inclusive forms of education in the emerging Information Society, and is attracting considerable attention at both government and industrial level [see, e.g., Commission of the European Communities, 2000a; Commission of the European Communities, 2000b; Kerrey et al., 2000; Hodgins, 2000]. The advent of the World Wide Web and of multimedia technologies has marked a turning point in the way computer-based education is conceived, designed, delivered and experienced. The WWW enables the wide distribution of educational material in which content is presented using a wide variety of media (hypertext, graphics, animation, audio, video, and so on). Web-based learning involves an increasing variety of target learner groups, including, for example, adults, children in secondary education, people in geographically dispersed areas, (re-)trainees, disabled and homebound, etc., in an increasingly wider variety of educational activities, including pre-university and university courses delivered on the web, training in industry, life-long learning, learning on demand, and many others. As a consequence, distance learning is evolving from a form of education in which the learner and the teacher are physically separated, to the provision of “whatever educational opportunities to anybody, anywhere and at anytime, characterised by a great diversity in practice” [Spodick, 1995, p.1].

The role of technology, in this context, is not only limited to providing more efficient, simple and cost-effective solutions for overcoming distance and time in education, but also to create and make widely available new forms of learning, exploiting the capability of dynamic electronic content to be manipulated, structured, retrieved and presented in a variety of ways. In this respect, one of the most promising characteristics of Web-based learning is the concrete possibility of providing adaptation and individualisation of learning resources on the basis of learner’s characteristics such as, for example, learning attitudes and strategies, background knowledge, and purposes of learning. One of the main peculiarities of the Web in terms of interactivity, when compared to more traditional computing environments, such as windowing systems, is the difficulty in drawing a clear line between the content and the interface. The specification of any single Web page (in effect an instance of a hypertext document) mixes together what is traditionally considered information content with interactive behaviour of the information content itself. It follows, therefore, that adaptation of web-based courseware implies both interface and content adaptation.

This paper reports on a preliminary investigation of some of the most important issues related to interaction and content adaptation in web-based courseware. The adopted perspective is that adaptation should aim to be learner-centred in the first place, i.e., it should respond to the educational and interaction needs, requirements and preferences of individual learners. The following section provides a brief overview of adaptation techniques developed or applied in learning technologies, with particular emphasis on those more relevant in the context of web-based courseware. The two subsequent sections focus on the two main sources of
knowledge necessary for achieving adaptation, namely the learner model and the course model. Subsequently, current relevant standard proposals are briefly overviewed. Finally, some important aspects of an architectural framework suitable for supporting content and interaction adaptation in web-based courseware are presented, focusing in particular on the required reasoning component and delivery mechanism.

2. Adaptation in web-based courseware

In the application domain of education, adaptation has been investigated in a number of recent research efforts, mainly falling into the category of Intelligent Tutoring Systems and Adaptive Hypermedia. The goal of Intelligent Tutoring Systems is to use the knowledge about domains, learners, and teaching strategies to support flexible individualized learning and tutoring [Brusilovsky, 1994; Brusilovsky, 1999, Murray, 1999]. Adaptive hypermedia systems apply different forms of user models to adapt the content and the links of hypermedia pages to the user, and have found in education one of their main applications [Brusilovsky, 1996; De Bra, 1999].

In these efforts, several adaptation techniques have been developed, the most relevant of which are curriculum sequencing, adaptive navigation, and adaptive presentation. The goal of curriculum sequencing [Vassileva, 1997; Stern and Park Woolf, 1998; Stern and Park Woolf, 2000] is to provide the learner with the most suitable individually planned sequence of knowledge units and learning tasks (examples, questions, problems, etc). Sequencing can be active or passive. Active sequencing implies a learning goal (a subset of domain concepts or topics to be mastered), and aims to provide the optimal individual path to achieve such a goal. Goals can be fixed or adjustable. High-level sequencing or knowledge sequencing determines the next learning subgoal (e.g., next concept, set of concepts, topic, lesson). Low-level sequencing or task sequencing determines the next learning task (e.g., problem, example, test) within the current subgoal. Active sequencing can be driven by the learner's knowledge and preferences on the type and media of learning material. Passive sequencing is a reactive technique that does not depend on an active learning goal. It starts when the learner is not able to solve a problem or answer a question (questions) correctly, and aims to offer learning material suitable for filling the learner's knowledge gap [Brusilovsky, 1999].

In the context of Web-based courseware, curriculum sequencing is a particularly important issue, given the vast amount of educational material that can be provided on the Web, and the risk that the learner gets “lost in hyperspace”.

Adaptive navigation [Calvi & De Bra, 1997; Weber and Specht, 1997] aims to guide the learner through the system by customising the link structure. Adaptive navigation determines the level of guidance and freedom granted to the student within the system, and attempts to give the learner a spatial representation of the hypermedia environment. Adaptive navigation can be considered as a generalization of curriculum sequencing in a hypermedia context [Brusilovsky, 1999]. In the context of the web, where hypermedia is a basic organizational paradigm, adaptive navigation can be used very naturally and efficiently. Several techniques
can be adopted for implementing adaptive navigation, such as adaptive guidance, link hiding, link annotation and link sorting. Direct guidance implies that the system informs the learner about the available links which point to the optimal next step to follow, established on the basis of learner’s current knowledge and learning goals. Hiding or disabling links is another frequently used technique that provides the learner with a representation of path followed through the system.

The goal of adaptive presentation [Brusilovsky et al., 1998] is to adapt the content of a hypermedia page to the user’s goals, knowledge and other information stored in the user model. The “unit of presentation” is a page (or node). Pages are not static, but adaptively generated or assembled from constituent pieces. Content components may be words, phrases, paragraphs, images, video clips, etc, constituting discrete units of information about a concept. Consequently, the granularity of the adaptation may vary from word replacement to the substituting of pages or the application of different media. Content may be customised to contain additional information, pre-requisite information or comparative explanations. As a result, each learner may potentially be provided with an individually tailored course that is different to the course displayed for all other learners. Adaptive presentation is an important issue in web-base courseware, where the “same” page has to suit the learning requirements and purposes of very different learners.

In Human-Computer Interaction, self-adaptation of a system’s interactive behaviour have been proposed as a framework for providing accessibility and high interaction quality to all potential users, on the basis of each user’s individual abilities, requirements and preferences, as well as of the context in which interaction takes place and of the adopted interaction technology [Stephanidis, 2001b]. Under this perspective, adaptation implies the capability, on the part of the system, of capturing and representing knowledge concerning alternative instantiations suitable for different users, contexts, purposes, etc, as well as of reasoning about those alternatives to arrive at adaptation decisions [Savidis et al, 2001]. Furthermore, adaptation implies the capability of assembling, coherently presenting, and managing at run-time the appropriate alternatives for the current user(s), purpose(s) of use and context [Savidis and Stephanidis, 2001]. The self-adapting behaviour of a system is based on knowledge concerning: (i) the available alternatives, (ii) the characteristics of users, the purpose and context of use, etc, and (iii) the optimal matching between alternatives and characteristics. As a consequence, a self-adapting system should include appropriate sources of knowledge, in the form of models, as well as appropriate reasoning mechanism, for deriving adaptation decisions related to what content the user should be presented with and through which interface. User interface adaptation can be classified along different dimensions, including, but not limited to, the policy employed in providing user control over adaptations, the knowledge utilised by the system to decide upon required adaptations, the decision making process utilised in mapping existing knowledge to adaptation decisions, etc.

From the point of view of web-based courseware, all the types of adaptation mentioned above are relevant. In a learner-centred perspective, content should adapt, as a minimum
requirement, to the learner's previous knowledge, individual progress, learning goals and learning style. Content adaptation should be applied each time a particular course topic is to be entered, so that the learner is initially presented with the appropriate material, as well during learning sessions, so that the subsequently presented material is selected on the basis of learner specific requirements at run-time. Adaptation to the learner's navigation style and preferences should also be considered. This implies that both course structure and learning sessions are flexible and can be assembled on the fly and modified at run-time as necessary, and that the learners' characteristics, the educational material, the course structure and the course delivery process are appropriately modelled, so that the best match between their characteristics can be established at any point in time during the use of the system.

Interaction, on the other hand, should be adapted, as a minimum requirement, to the learner's computing experience. A common pitfall in many currently available educational environments is that they provide the same interface for both novice and experienced users, even though their knowledge of the system may differ at the beginning and the end of the course. Consequently, when the user interface is oriented to experienced learners, it is too complex for the novice, and, conversely, when it is simple enough to be used by novice learners, it is usually inadequate for more advanced learners [Specht and Weber, 1996]. This implies that the interaction requirements of learners, as well as the interaction characteristics of the system and of its content, need to be modelled.

3. Learner's model

A learner model contains explicitly modelled assumptions that represent the characteristics of the student that are relevant to the system, and in particular to the type of adaptation that it is desired to achieve [Conlan and Wade, 2000].

Several techniques are commonly used for learners' modelling in Intelligent Tutoring Systems. One of the simplest is the use of fixed stereotypes. Learners are categorized, and adaptation is based on such a categorisation. Category assignment can be based on questionnaires or tests prior to the use of the system. The overlay model is also widely used in the educational domain. In this case, the model is constructed on a concept-by-concept basis and updated progressively. This allows, amongst other things, for a flexible model of the student's knowledge for each different topic [Brusilovsky, 1994; Brusilovsky, 1996]. For this purpose, the knowledge domain must be modularised into specific topics or concepts, and the complexity of the model depends on the granularity of the structure of the domain knowledge, as well as on the granularity of the estimation of the learner's knowledge, performed by examining the sections the learner has read and by evaluating the knowledge acquired through tests (either automatically or by traditional testing methods). Stereotypes and overlay techniques can be combined. Learners may be categorised by stereotype as an initial phase. Subsequently, the model is gradually modified as the overlay model is built from information acquired from the student's interaction with the system [Conlan and Wade, 2000].
A number of sources of information may be used to construct a learner model. The system acquires data about the user and infers learner characteristics from this data. The validity of the assumptions depends on the technique used to acquire the information. Automatic modelling may be unreliable, and therefore collaborative and cooperative modelling is frequently chosen. The learner is asked to describe pertinent characteristics directly, by filling out questionnaires and forms. Indirect feedback can be acquired from the results of exercises or problem solving tasks [Conlan and Wade, 2000].

The properties chosen to represent the user should be pertinent to the potential adaptation by the system. Learner characteristics frequently modelled are the user's previous knowledge, goals and objectives, cognitive style, learning style, maturity, general ability, confidence, motivation, preferences and background [Specht and Weber, 1996].

The domain knowledge of a student can be utilized to adapt both the content and the interface. Students' knowledge of the subject evolves continuously, and systems need to adapt to the changing knowledge state of a user. Depending on the knowledge of the learner the presented content can be restricted or expanded. Additional explanations can be given to students with more sophisticated knowledge while novices are confronted only with the basic information. The knowledge state of a student can be relevant for a number of adaptation types, such as, for example, the selection of the next task (task sequencing), the number of tasks (mastery learning) and the next learning intervention, and the adaptation of questions and explanations.

Learners’ goals can be distinguished into high-level and low-level goals. A high-level goal is stable during a session or across sessions (e.g., the acquisition of knowledge in the subject domain). Low-level goals are subject to changing often and quickly (also within a single session). Low-level goals, like learning something about a certain domain concept or solving a problem are often important for short-term adaptation purposes (e.g., displaying or bringing to focus relevant parts of content). Information about a user's goals and interests is also an important source of information for choosing the form of presentation of selected content [Specht and Weber, 1996].

The provision of learner-centered adaptation in web-based courseware should capture all information considered relevant regarding the engagement of learners in electronic training courses. In order to ensure wide applicability and reusability, the model should be general enough to address different categories of knowledge domains, learning purposes and training processes. For example, it should be possible to create learner profiles for formal classroom education, professional training, informal learning, etc. This implies the need of differentiating the type of information to be modeled according to its relevance for the specific case. For instance, in classroom-based training, evaluation records may need to be maintained, while in professional training, organizational-role specific information will need to be managed. To achieve such a flexibility, the learner model should be based on meta-data, and encompass appropriate abstract attributes, enabling concrete learner models to be instantiated for different knowledge domains and training processes. It is important that meta-data take into
account currently proposed standards. As intensive use of modeling is required, models should be easy to create, consult and up-date. A possible technique for model information input is the provision of alternative templates for specific domains. Query facilities for existing profiles should also be provided.

For interaction adaptation, knowledge about relevant learners’ characteristics is also necessary. These may include sensory, motor and cognitive abilities, expertise in the use of computers, of the web, and of the particular system, knowledge about specific interface components, colour and media preferences, preferences in interaction and navigation styles, etc.

4. Course model

Besides information about learners, an appropriate model of the data to be delivered in a course is necessary, as well as of the delivery process. Content information, constituting the “raw material” which is to be appropriately selected and dynamically assembled in a learner-adapted course, should be logically separated from the course information, providing semantic information regarding the overall course structure. This logical split enables a course to be delivered in different forms, by controlling various semantic parameters, such as level of detail, type of explanation, and complexity of examples. These models should describe all aspects of how the content should be used and whom it is designed for. In this section, the first model will be referred to as content model, while the second will be referred to as process model.

The content model identifies: (i) the elementary constituent units of content; and (ii) the structure in which elementary constituents are organised. In recent standardisation efforts, units of content are referred to as ‘learning objects’ [IEEE-LTSC, 2000a]. Such a unit can be a paragraph or other piece of text, an image, a video clip, etc. The unit internal structure should not be relevant for the model. Units may be static (stored) or may be generated by an external application (like a natural language generation module) [De Bra, 1999].

The knowledge domain can be described in the content model in terms of concepts (or terms) and relationships between concepts. Concepts may be composite, and can be used for defining prerequisites providing meaningful paths through the information. When concept A is a prerequisite for B it means that the user should visit (pages about) A before B. Other types of concept relationships can be defined as well. An example is inhibitor relationships: A inhibits B means that after a learner has visited concept A, it is no longer necessary that concept B is visited [De Bra, 1999].

The granularity at which the content is stored determines the level of content adaptation that may be achieved, and richer metadata sets are required to adequately describe finer-grained content. Extra level of detail in the metadata can be required, for example, to distinguish similar, but different units content. For example, two pieces of content can both describe the same concept. One may be a theoretical description, while the second may be an application example. The model should offer a clear way to differentiate these two objects [Conlan and
Wade, 2000]. Reuse potential is also closely related to the granularity of the content description. Coarse-grained content may be too domain specific to be easily reusable.

In learner-centred adaptation of web-based courseware, the notion of learning object is particularly important. Learning objects should link directly to content, and encompass course-related semantic attributes. As content can manifest itself in different forms, learning objects should not have a singular mapping to content resource items (i.e., content entities). Rather they should be abstract, i.e., capable of linking to different content items. Additionally, they should not be physically attached to content. This can be accomplished by associating learning objects to content via resource identifiers, as opposed to embedding content within learning objects. Such a logical split also support a greater degree of reusability. Learning objects do not need to be restricted to simple elements such as paragraph items, but may range from multimedia content, more comprehensive instructional content, to external software and relevant tools. In a wider sense, learning objects in web-based courseware could include, and are not limited to, passive items, active items, collaboration gateways, personal contacts, and social events. Alternative templates can be supplied for these metadata categories. Some typical attributes of learning objects could include, for example: type, Author, Owner, Terms of distribution, Format, Teaching / instruction style, Grade level, Mastery level, Prerequisites, etc. For interaction adaptation, learning objects should also include information concerning their interaction characteristics (e.g. color, font, interface objects, required interaction techniques), to be matched with learners’ interaction requirements and preferences.

Current proposed standard specification can be considered as a comprehensive modelling of learning objects, aiming to be domain-free. In this respect, it constitutes a universal template. However, due to its intended generality, such a specification offers a taxonomy of semantic attributes which do not allow for domain-specific concepts, something which could constitute serious constrain domain-oriented adaptations. Therefore, domain specific templates should also be defined.

The structure of a course may vary depending on many factors. For instance, educational material for a course may be structured as a hypermedia book, a tutorial, a training session, a seminar, etc. Course structures are usually specified on the basis of a hierarchical organisation model, in which the most primitive constituent components are the learning objects.

From the point of view of adaptation, the process model constitutes the main backbone for matching the content to be provided to the learners’ characteristics. Different templates can be provided for different type of courses. An important requirement on course templates is that they cater for alternatives to be made available to different users in different situations, and provide information for selecting amongst alternatives, i.e., they define multiple courses in one hierarchical structure, depending on the learner’s level, purpose of learning, background, acquired prerequisites, etc. Learning objects’ templates and course templates should appropriately link, in order to enable the implementation of facilities such as, for example,
search, evaluation, selection, and acquisition of learning objects, sharing across different courses, semantic manipulation in creating personalised learner-centred lessons, and documentation and recognition of completion for learning / performance objectives associated with learning objects. The process model should also provide information for course management at run-time (e.g., start date, expected duration, end date, real duration, delays causes, course identification, sessions, plan, enrolled learners, tutors, course topic index, completion status, feedback from learners).

5. Relevant standard proposals

There are a number of ongoing standardisation efforts in the field of modelling for educational technologies. Most of them concern the architecture, learner modelling and learning objects’ modelling.

The Dublin Core [Dublin Core, 1999] is a metadata element set for content description, intended to facilitate discovery of electronic resources. Originally conceived for author-generated description of Web resources, it has attracted the attention of formal resource description communities such as museums, libraries, government agencies, and commercial organisations. It provides a simple structure which lends itself well to educational resources and, by attempting to unify other data content standards, it increases the possibility of semantic interoperability across disciplines. The 15 element Dublin Core metadata set was finalised in December 1996. Dublin Core have recently established a working group to look specifically at educational metadata issues.

The SO-IEC JTC1 SC36 Information Technology for Learning and Education is the ISO working group on setting up standards for learning technologies. Currently, the working group studies previous work on standardisation, and establishes links and liaisons, together with an appropriate technical agenda. The group is still in its “infancy”, while from recent publications it is clear that it will likely focus on the filtering and acceptance of currently existing standardisation work, especially that of IEEE LTSC, rather than producing specifications from scratch.

The IMS Global Learning Consortium, Inc. (IMS) is developing and promoting open specifications for facilitating online distributed learning activities such as locating and using educational content, tracking learner progress, reporting learner performance, and exchanging student records between administrative systems. IMS has two key goals, namely defining the technical specifications for interoperability of applications and services in distributed learning, and supporting the incorporation of the IMS specifications into products and services world-wide. IMS endeavours to promote the widespread adoption of specifications that will allow distributed learning environments and content from multiple authors to interoperate. IMS profiles specification detail learner information such as qualifications, institutions attended and learner preferences containing technical, physical and cognitive details [IMS, 2001a]. The IMS Consortium is also developing the Content Packaging Information Model [IMS, 2001b] that describes a self-standing package of learning resources.
Such a model is defined as describing data structures that are used to provide interoperability of Internet based content with content creation tools, learning management systems, and runtime environments. Standardized sets of structures can be used to exchange content. These structures provide the basis for standardized data bindings that allow software developers and implementers to create instructional materials that interoperate across authoring tools, learning management systems and runtime environments that have been developed independently by various software developers.

The IEEE LTSC working groups is developing technical Standards, Recommended Practices, and Guides for software components, tools, technologies and design methods that facilitate the development, deployment, maintenance and interoperation of computer implementations of education and training components and systems. LTSC has been chartered by the IEEE Computer Society Standards Activity Board. Many of the standards developed by LTSC will be advanced as international standards by ISO/IEC JTC1/SC36 - Information Technology for Learning, Education, and Training.

PAPI [IEEE-LTSC, 2001b] is the IEEE Public and Private Information Specification that is a standard format for the representation and communication of learners’ profiles. The purpose of the specification is to allow the creation of records that can be communicated between educational systems over the lifetime of a learner. The profile information for a learner is divided into four areas: (i) – personal information which is for private consumption such as the student's name, address and Social Security number; (ii) preference information which may be for public consumption, such as the technology available to the student, the learning style of the student, physical limitations or disabilities, and which collected with the cooperation of the student, i.e., it is negotiated; (iii) performance information which is for consumption by technology, and consists of the observable behaviour of the student including, for example, grades, reports, logs, etc; and (iv) portfolio information which is for consumption by humans, such as the student's accomplishments and works. The PAPI specification also incorporates the Dublin Core metadata element set. The information used to construct the user profile is inferred by the system, directly input by the user or is constructed by the user and system in collaboration. PAPI also intends to address the privacy and security issues involved in the storage and communication of user profile information. The IMS profile specification and the PAPI codify similar information, especially as far as technical and physical preferences are concerned.

The IEEE-LTSC Learning Object Model (LOM) metadata maps directly to the Dublin Core metadata element set. The proposed model is a structured model of metadata to describe learning objects. The Learning Object itself is defined as any entity that can be used or referenced during technology supported learning [IEE-LTSC, 2000a]. The Learning Object Model attempts to create a framework that specifies the syntax and semantics of the metadata required to describe the attributes of a Learning Object. This metadata standard allows educators to search, evaluate, acquire and utilise Learning Objects. It is also intended that this standard will facilitate the sharing and reuse of Learning Objects. LOM defines over
seventy optional and mandatory elements ranging from technical descriptions of the Learning Object to pedagogical considerations. The current LOM model includes the following nine elements, some of which can occur multiple times:
1. General: title, category/entry, language, description keywords, coverage, aggregation level.
4. Technical: format, size (bytes), location, requirements (installation, platforms), duration.
5. Pedagogical: interactivity type, learning resource type, interactivity level, semantic density, intended end user role, learning context, age range, typical learning time, description of how the object is to be used.
6. Rights (cost, copyright, conditions of use).
7. Relation (kind, target resource).
8. Annotation (person, date, description/comment).

The Relation and Classification elements and sub-elements are especially relevant from the perspective of supporting content adaptation, as they provide semantic relationships among learning objects. The Classification element provides the principal mechanism for extending the LOM model by allowing it to reference taxonomy and describing associated path sub-elements. Thus, classification provides for multiple alternative descriptions of a LO within the context and meaning of several taxonomies. Recently, a standard architecture has also been proposed [IEEE-LTSC, 2000c].

6. Learning systems architecture

This section discusses a proposed architecture for learner-based content and interaction adaptation in a courseware environment comprising both authoring and delivering facilities, and in which concrete learning object instances are physically distributed. Adaptation is considered to be based on learner information, course information, learning objects, and process information. A high-level representation of an architecture for the core of such a system is depicted in Figure 1.

The architecture encompasses modules implementing the Learners and the Course models, which are created and stored by course authors on the basis of available templates and meta-data.

For adaptation to take place, additional components are required, namely the Adaptation Logic component and the Delivery mechanism.
The adaptation logic provides the reasoning mechanism required for accomplishing both interaction and course adaptation, on the basis of the relevant characteristics captured in the models. The reasoning process will take place both at the beginning of and during each learning session, and drives to decisions concerning the learning objects which should be displayed and how they should be assembled. In other words, the adaptation logic provides a decision mechanism for dynamically selecting, among the multiple instances of course components and learning objects available, the one(s) appropriate for presentation at a given point in a learning session of a specific user. In this respect, the adaptation logic combines the information provided in all models in a coherent whole. Suitable implementation techniques include production systems, decision trees and declarative logic programming. Although general adaptation rules can be defined, it is unlikely that all adaptation cases for a variety of learners’ characteristics and courses can be captured at a general level. Therefore, adaptation rules should be editable, i.e., course authors should be allowed (and supported in) defining their own adaptation logic on the basis of the knowledge domains, learners and purposes their course is designed for.

Apart from deciding which learning objects should be supplied to learners, the physical form of such objects should be retrieved and delivered appropriately. Delivery may imply the presence of mechanisms to assemble a physical structure that is not only “acceptable”, but also high-quality. For instance, assume that the learning objects correspond to independent paragraphs in a document, while the necessary physical data are supplied in different locations, and may contain both text and images. The assembly of those items into a single document should be carried out in a way leading to a high-quality visual document structure. Additionally, the delivery platform should be able to handle the retrieval and display of such physical data (i.e., text and images in their particular format). The content assembly process can be based, for example, on domain-specific HTML templates.
As already mentioned, in web-based learning systems, a clear separation between the content and the user interface is not possible. The reason is that content is also interactive, supplying all kinds of interaction objects and techniques which are normally classified as belonging to the interfacing layer. Therefore, two types of user interface can be identified: the delivery interface, and the content interface. The former provides some standard facilities such as history control, navigation, and communication, while the latter provides content-specific interaction methods, which depend on the physical delivery of a learning object (i.e. embedded in form). Consequently, to accomplish user interface adaptation, both levels of interfacing need to be affected. The delivery interface can realise adaptation according to information on learner’s interaction requirements and preferences. Regarding the content interface, the meta-data specification for learning objects need to attach interaction-related attributes to the physical resource identification, thus supporting the selection of the most appropriate physical forms. While the delivery interface is chosen at the beginning of a session, the content interface is chosen at run-time (if interaction alternatives exists) each time new content is retrieved.

6. Summary and Conclusions

This paper has briefly reviewed the most important adaptation techniques developed and applied in the context of interactive learning technologies, with particular emphasis on those more relevant in the context of web-based courseware, and has discussed the two main sources of knowledge necessary for achieving adaptation, namely the learner model and the course model, as well as the relevant standard proposals for modelling.

The adopted perspective is that adaptation should be learner-centred, i.e., it should continuously deliver, at any time during the use of the system, the most appropriate interactive learning content to each individual learner. Adaptation has been considered at both the content and the interaction level, as they are inseparable in a web environment. A first implication of this is that content adaptation should take place not only on the basis of the educational and information characteristics of content units, but also on the basis of the interaction properties of such units. Secondly, this implies that content should be dynamically assembled in a way that produces high quality results in terms of both appropriateness to the learner’s knowledge status and learning goals, and appropriateness to the learner’s interaction requirements and preferences. As a consequence, the adaptation of the interaction should be investigated with respect to its relationships to the adaptation of educational content and content delivery method, so as to lead to a global systematic approach toward learner-centred adaptation. This implies identifying the required modelling attribute in the learners’, educational content and delivery process models, as well as their interrelationships. While these models and meta-data should build on existing standard proposals as well as on previous work in the fields of Intelligent Tutoring Systems, Adaptive
Hypermedia and Human-Computer Interaction, they also need to be general and flexible enough to cater for a variety of knowledge domains and educational purposes, and at the same time to be specific enough to support meaningful adaptation in each application. Furthermore, models should be simple enough to allow extensive modelling on the part of course authors and teachers, and should support the easy definition of alternatives for course structures, educational material units and interaction behaviour characteristics of such units. Finally, the paper discusses some important aspects of a proposed architectural framework suitable for supporting content and interaction adaptation in web-based courseware. Important constituents of such an architecture are the adaptation logic and the delivery mechanism. The adaptation logic is required for reasoning on the available alternatives and select the optimal material to be presented to learners according to the set parameters. Given the wide variety of possible applications and knowledge domains, it is likely that adaptation rules will vary, and as a consequence, it should also be possible to define different rules to cater for different cases. The delivery mechanism role is to assemble in a coherent unit the material chosen for presentation, and offer it to the learner.

A wide range of research issues emerges from the above analysis. To mention only some examples, the granularity at which content should be broken down and analysed constitutes a very important aspect, likely to highly affect the adaptation process and the necessary information. Appropriate granularity levels are likely to vary for different knowledge domains and media, and different modelling solutions are required to be supported. Another important issue concerns the source of knowledge for continuous learner modelling during system use. It is likely that completely automated assessment of the user progresses is not a viable solution, and that human assessment is required at some point in the process. However, on-line monitoring of dynamic user characteristics, which may change in the time-frame of a session, is also required. Therefore, learner characteristics should be carefully investigated with respect to when and how their values and history need to be up-dated.

References


