We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,300

Open access books available

172,000

International authors and editors

190M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Personalized Learning Path Delivery

Hend Madhour and Maia Wentland Forte
*Business School of the university of Lausanne
 Switzerland*

1. Introduction

Getting the “right” set of pedagogical documents or proper information is a challenge that a learner can hardly overcome in an open environment. Not only is the form and content of online material very heterogeneous but it almost impossible for a user, and even more so for a novice, to discriminate between the plethora of available documents the best suited ones. Unless some kind of personalization is performed to better adapt the proposed material to the user’s needs.

In this chapter, we will first examine different manners of performing such personalization in the framework of the today’s richest open source information environment, the Web. Then, we will concentrate on adaptive educational systems and dig more into the reference models these systems are built on. After showing the limitations of such models in an educational context, we will present a new reference model that we claim brings an elegant solution to these limitations, namely the Lausanne model.

2. State of the Art

In this section, we present approaches, techniques and tools used to provide personalization in the today’s richest open source information environment, the Web.

2.1 Personalization approaches

Personalization consists in adapting the behaviour of the system according to some specific information related to an individual user. We have classified these approaches into three types, depending on how information is collected:

2.1.1 Individual Vs Collaborative

Personalization advocates the user's individuality. To adapt the system behaviour to the users' needs, the system collects specific information about the user herself (interest, preferences, age, etc.) and/or about her interactions with the system (interaction history). Two approaches, illustrating each of these, can be highlighted:

(i) Individual : Building a user profile, or individual model, that contains information about what the user likes/dislikes and making use of this profile to predict/adapt future interactions (Ken, 1995).

(ii) Collaborative: Using the active user profile and the ones of others that share some common interests. This approach is called *collaborative filtering* and is used by (Resnick et al., 1994).

2.1.2 Reactive Vs Proactive

Some personalization systems need explicit interactions that takes the form of a special request or a feedback. Those systems are called reactive systems. We mention Entree (Burke et al., 1996), Detorecs (Fesenmaier et al., 2003), ExpertClerk (Shimazu, 2002) as examples of those systems.

Other personalization systems learn about user preferences and give recommendations using this information if given by the user. In fact, giving such information is not compulsory to run those systems. Besides, users choose to follow or ignore generated recommendations. Those systems are called proactive systems. We mention Amazon.com, CDNOW (CDNOW, 2008), GoupLens (Resnick et al., 1994), MovieLens (Miller et al., 2003) and Ringo (Shardanand & Maes, 1995) as examples of those systems.

2.1.3 User Vs Item oriented

Sources of the information needed to perform personalization are of two types:

- Information about the user: past feedback, behaviour, demographic information (age, sex, origin, education)
- Information about the item: content description, domain/product ontology

Most user oriented systems (Ken, 1995) ; (Ghani & Fano, 2002) are based on user's behaviour (past bought or rated items). Few ones (Krulwich, 1997) exploit demographic information because such information is difficult to collect.

2.2 Personalization techniques

2.2.1 Content filtering

Two types of information are used to generate recommendations (Mladenec, 1996); (Krulwich & Burkey, 1997); (Ken, 1995) :

- User profile based on item's content description: it is used to predict unseen items rate.
- Rated items analysis

Systems using this technique tend to recommend items that have big similarities with previously seen items.

2.2.2 Social techniques

Collaborative filtering

This technique is an alternative to content filtering. In fact, it enhances the users collaboration by helping each others to filter items. The users rate *consumed* items. Rates given by users sharing the same interests are used to recommend suitable items. Based on a matrix item/user, the system recommends by prediction adequate items.

Genetic algorithms

In the social context, artificial intelligence can take advantage of biological observations of certain species to imitate their experience. This trend, called evolutionary computation, uses

genetic algorithms on one hand and swarm intelligence techniques on the other hand. The ACO algorithm belongs to this trend. In a simplified reality, the ants start moving randomly. Then, when they find food, they come back to their colony, secreting in their way chemical substances called pheromones. If other ants find this path, there is a high probability that they follow the path marked with pheromones reinforcing it when coming back.

Consequently, the more a path is visited, the more it will be reinforced. Conversely, because the pheromone evaporates, the less reinforced paths slowly vanish leading all ants to follow the shortest one.

Applied to distance learning, such algorithms require to model learner's behaviour, task that for the time being, has not been very efficiently accomplished. For example, (Semet, 2003) models the learner's memory evaporation by a mathematical function depending on a parameter assessed within seconds. This leaves us a bit perplexed because we do not have the means to demonstrate the validity of this assertion.

2.3 Personalization tools

2.3.1 User Modelling

Overlay model

Popular user models are overlay user models; these share the same representation as domain models and are used to represent a user knowledge / interest in a resource space. These models are vectors of attributes (measures of interest or knowledge), one for each concept in the domain model. These are updated from user navigation in the domain model. However, the definition of the domain models and of the users' models is done manually.

As examples of systems using the overlay model, we mention Orimuhs (Encarnacao & Stork, 1996), Push (Espinoza & Hook, 1996), Hypertutor (Gutierrez et al. , 1996), ELM-Art (Weber & Brusilovsky, 2001), Hynescosum (Vassileva, 1996) and ADAPTS (Brusilovsky & Cooper, 1999).

User profile

It represents cognitive styles, intentions, learning styles and preferences. It is entirely user oriented because information is provided by the user itself. For this reason, result suitability depends on the information provided by him.

As examples of systems using the user profile, we mention EPK (Timm & Rosewitz, 1998) and SmexWeb (Albrecht et al., 1998).

Stereotype model

It takes the form of a couple (stereotype, value) or a Boolean value that indicates either or not user belongs to a specific stereotype. This model is simpler but less powerful than the overlay model.

2.3.2 Updating user model

Acquiring the user model can be done explicitly via interviews and questionnaires or implicitly via information inference based on available user information (by learning or direct observation). Implicit acquisition needs to automate the manner to fill user model attributes. This automation petrifies user model reliability. Besides, specific attributes such as preferences and proficiencies can not be deduced by the system. That is why it is important to let the user participate in acquiring his user model. This way is called cooperative modelling.

2.3.3 Sharing user model

In an open environment, mobility between different learning environment is a need. It is important to have a unique user model that is updated during the curriculum.

A first initiative consists in defining specifications and standards used by all learning environment. The current specification of PAPI (Papi, 2008) splits the learner information into 6 areas: personal information and preference information, performance, relations, portfolio and security. IMS/LIP (Lip, 2008) is a specification that describes the learner's characteristics to personalize the content. Conversely, LIP divides the learner information into 9 areas: interest, affiliation, QCL (Qualifications, Certifications and licenses), activity, goal, identification, competency, relationship, security key, transcript and accessibility.

This specification is much more detailed than PAPI and provides almost a complete users' profile. But none of them differentiate the access rights allowing the user to modify all areas.

Another initiative is to adopt a web service such as UMOWS (User Model Web Service) (Bielikova & Kuruc, 2005) that update a centralized user model after each learning environment call.

3. Personalized education systems

When moving from traditional learning to educational e-systems, students get increasingly involved in their learning process; technological systems (mainly Internet) are the new vectors used to disseminate knowledge between and provide feedback amongst the learning process actors, i.e. the pedagogues, the tutors and the learners. The use of IT in education covers a wide range of very different activities: authoring, course management, web sites conception, communication, simulations, learning environments, and much more. Its contribution to the process goes all the way from being a simple on-line help or support (current traditional or blended learning) to that of sophisticated main device (distance learning). Because the *one-size-fits-all* paradigm cannot be applied to individualized learning, adaptability is becoming a must. Hence, courseware is meant to be tailored according to the learner's needs. Two main families of computerized applications aspire to offer this adaptability: Intelligent Tutoring Systems (ITS) (Brusilovsky, 1992) and Adaptive Hypermedia Systems (AHS) (Brusilovsky, 1996).

3.1 Intelligent tutoring systems

Intelligent Tutoring systems (Brusilovsky, 1992) rely on curriculum sequencing mechanisms to provide the student with a path through the learning material. An adaptability algorithm computes this so-called *personalized path* corresponding to the course construction, the *curriculum sequencing* (Shang et al., 2001).

The process is twofold:

- Find the relevant topics and select the most satisfactory one
- Construct dynamically page contents based on the tutor decision for what the learner should study next.

ITS usually provide an evaluation of the learner's level of mastery of the domain concepts through an answer analysis and error feedback process that eventually allows the system to update the user's model. This process is called *Intelligent solution analysis* (Serengul & Smith-Atakan, 1998).

Finally, the learner may need some help during the solving process from displaying a hint to executing the next step for him. This intelligent help is provided by *an interactive problem solving support* (Serengul & Smith-Atakan, 1998).

VC-Prolog-Tutor (Peylo et al., 1999), SQL-Tutor (Mitrovic, 2003), German Tutor (Heift & Nicholson, 2001), ActiveMath (Melis et al., 2001) and ELM-ART (Weber & Brusilovsky, 2001) are some of the examples of Intelligent Tutoring Systems.

3.2 Adaptive hypermedia systems

Adaptive Hypermedia (AH) (Brusilovsky, 1996) was born as a trial to combine intelligent tutoring systems and educational hypermedia. As in ITS, adaptive education hypermedia focus on the learner, while at the same time it has been greatly influenced by adaptive navigation support in educational hypermedia (Brusilovsky, 1996). In fact, adaptability implies the integration of a student model in the system in the framework of a curriculum which sequence depends on pedagogical objectives, user's needs and motivation.

AH provides two major functionalities :

- The Adaptive presentation is a functionality that helps the hypermedia take advantage of information included in the student's model of a connected ITS. Two distinct methods allow to perform this feature: the comparative explanation method used in LISP-Critic (Gonschorek & Herzog, 1995) and the explanation variants method used in LISP-Critic (Gonschorek & Herzog, 1995), Anatom-Tutor (Beaumont, 1994) and Sypros (Fischer et al., 1990). The first one scaffolds upon previously acquired knowledge, while the second organizes the domain knowledge by topics and levels of mastery.
- The Adaptive navigation support aims at helping users to find their paths in hyperspace by adapting the navigation and displayed functionalities to the goals, knowledge, and other explicit characteristics of an individual user.

In adaptable education systems, the tutor is involved as an expert. Some research (Hernandez & Noguez, 2005) tries to model the tutor's expertise in order to automate the learning process as much as possible. Modelling this expertise is a restrictive process because we postulate that it is impossible to formalize all of the tutor's know-how. Moreover, when modelling the learner, the tutor intervention in the learning process should be taken into account. Besides, other actors' interventions in the system implementation must be clarified such as those of the author and the instructor.

3.2.1 Reference models

A reference model allows to collect the specifications and best practices to provide documentation and guidelines for a community of practice. It must also be precise enough to show the know how and broad enough to be malleable. In fact, standardize consists in establishing definitions and specifications of syntactic and semantic rules, and descriptions of environments. A standard must be neither prescriptive nor exclusive.

The concept of reference model was discussed at the level of hypermedia well ahead of educational objects, while a major part of hypermedia are for educational purposes.

For this reason, we propose to present existing reference models for adaptive hypermedia systems.

Dexter

The Dexter model (Halasz & Schwartz, 1990) describes the structures needed to define the links between information items. It was designed to clarify the concepts in existing hypertext systems.

Two levels are presented in this model: the atomic component and the component compound. The components are composed of collections of other components (atomic or composite) and links that connect the components to their children. These components are treated as a single component.

In addition, components (atomic or composite) can be connected by means of anchors. An anchor is part of the component, it can be a fragment of text, graphics, etc.

A component (atomic or composite) contains three main parts: content, semantic attributes, and presentation specifications.

Finally, two types of relationships exist in this model: the links via anchors and the relationships between components and their children.

Amsterdam

The Amsterdam model (Hardman et al., 1994) is an extension of the Dexter model and the CMIF (CWI Multimedia Interchange Format) model (Hardman et al., 1993). The latter takes into account the synchronization of blocks of information. It essentially supports multimedia while the model only supports Dexter hypertext because of the lack of synchronization information. Amsterdam addresses the need to have a model for hypermedia. The Amsterdam model adds to Dexter model the notion of channels that are in fact predefined presentation specifications. The channels are abstract output devices that are used to define overall characteristics of a certain type of media such as the volume for an audio channel. For an atomic component, specifications presentation is enriched by the name and length of the channel. Composite component was extended to include sync arcs, anchors referencing anchors descendants and a start time for each son.

Dortmund

The Dortmund Family of Hypermedia Models (DFHM) (Tochtermann & Dittrich, 1996) is a family of interrelated models rather than a single model with a specification of a data type. The concepts introduced in these models are:

- Link structures are a set of links that connect parts of the hyper document. Several links can be assigned to the same document. This can be very useful in the expression of different contexts for different users.
- Views are defined like the databases views to the extent that the information that is not interesting or relevant to a user is hidden.
- File is a container of nodes, links and other files. The records support the links between records, documents and documents beyond those folders.

AHAM

The model AHAM (Adaptive Hypermedia Application Model) (De Bra et al., 1999) is a reference model for adaptive hypermedia. In order to adapt, AHAM offers three models: the domain model, the user model and the adaptation model.

The domain model represents author view concerning the application domain. It describes the structure of an adaptive hypertext system as a finite set of component concepts (concepts and relations between concepts).

The concept is an abstract representation of a node of information in the application domain. Two types of concept are defined: atomic and composite. An atomic concept is a piece of

information when a composite concept is a sequence composed of concepts that can be composite (an abstract concept) or atomic (page).

The relationship between concepts must be transformed into relations between atomic concepts because only the pages can be displayed. This transformation is done by adjusting the engine. Several types of relationships are considered in the model AHAM: hyperlink, prerequisites, inhibiting.

The user model contains information that the system records about the user. It is based on user knowledge on the concepts. For each user, the system maintains a table where it stores the attributes describing the knowledge for each concept in the domain. Two attributes are present at least: (i) the value of knowledge that indicates the extent of knowledge of the user on a concept and (ii) the attribute *read* that indicates whether the user read something about the concept. This attribute can be either boolean or a list of access time.

The adaptation model contains the rules to be applied to teaching in the previous models for the purpose of adaptation. AHAM uses a rule language adaptation that looks like SQL (Simple Query Language) except that it does not contain the FROM part because it will always refer to the user model and the application domain. For example, to update the knowledge of a concept whose page is P, the following rule is followed:

C: Select P. Access where P.ready = true

A: Update P.knowledge: = "well known"

The condition C states that access to a page can only be done if the attribute *ready* is true. If you have accessed to P, the A states that the attribute *knowledge* is given to well known. The adaptation engine performs a number of tasks when a user accesses the system: It retrieves the corresponding user model. All attributes for all concepts are retrieved.

The other attributes used in teaching rules but not included in the user model are initialized to a default value. For example if the attribute *ready to know* is not in the user model, it is set to true for all concepts.

The engine determines the corresponding concept C based on the user model and teaching rules. It generates an HTML page by following the presentation specifications and updates the attributes of the user model.

Munich

The Munich model (Koch & Wirsing, 2002) is based on the Dexter model and extends it for an adaptation purpose. Indeed, added to the navigational relationships (links), there are also conceptual relationships (part of, prerequisite of, in the same page as a variant of). Like AHAM, a user model and a model of adaptation are required. The Munich model does not take into account aspects such as typical multimedia synchronization. It aims to construct specific views to the user, as it is the case in the DFHM model. It differs from other reference models previously described by its object-oriented approach and therefore the use of UML (Unified Modelling Language) modelling technique.

Limitations

Although the existing adaptive hypermedia systems have a potential in providing suitable learning resources, they remain relatively closed environments. In addition, the existing reference models tend to be generic and consequently do not address issues related to specific systems such as learning systems. More specifically, existing reference models suffer from several limitations

One of its fundamental limitations is the fact that it considers only two actors in the learning process: the author and the user. We think that this is too reductionist since more than two

roles are involved in this complex process, even though only two persons can play these different roles : from the teaching side, the designer, the tutor (and even the system); from the learning side, the learner:

- The role of *designer* includes that of author of pedagogical documents - called the author -, the author of a course which usually is a *teacher* who might not be the author of any of the proposed documents but who organizes the sequence by adapting and reusing existing material, the *pedagogue* who helps generating the sequence by defining contextually adequate pedagogical rules. All of these are called the *author* in the AHAM model.
- The tutor who supervises and facilitates the learning process
- The learner who is meant to benefit from the adaptive system.

Besides, the domain is composed of concepts connected by relationships following a restrictive vocabulary. On the one hand, there is no typology of concepts on which the domain model is based. On the other hand, there is no way to describe elements other than the concepts. Moreover, in the context of an open environment, the used restrictive vocabulary is a problem because it forces various sources to use the same vocabulary. This seems difficult even between two different designers. Finally, the concepts do not have a description formalism facilitating their discovery.

Finally, the user model takes into account only the information about the resource. It is often an overlay model. The current standards such as IEEE / PAPI and IMS / LIP are not used.

The adaptation model is generic and allows to the author to define the pedagogical rules. It remains dependent on the author, who in addition to learning resources must take time to design rules that take into account the specificities of each user. This requires a big effort. In addition, the relevance of the generated course depends strongly on these rules.

Because of all these limitations we propose a new reference model, called the Lausanne Model, extending AHAM, in order to meet our needs.

4. Lausanne Model

4.1 Domain model

In the context of a learning environment, the Domain Model focuses on pedagogical material content and aims at describing it by representing its entities and their relationships in a standardized manner. After discussing the granularity issue, we go in some details into the indexation and annotation problems and then proceed to illustrate our MLR solution.

4.1.1 Entities

The Learning community has come up with a number of ways to depict hypermedia each proposing different manners of tackling issues such as the level of granularity or the description scheme.

Granularity degree

By analysing some of the best-known models (Wagner, 2002);(Duval,2001);(Dodds,2001);(Barrit,1999);(L'atelier,2003),we see emerge, amongst others, the three following generic granularity levels:

Asset: is a document lowest level of granularity. As such, assets can be pictures, illustrations, diagrams, audio and video, animations, as well as text fragments

The Learning information is a group of assets expressing the same meaning. For example, a figure associated with its comment is learning information.

The Learning Object represents the semantic structure (or network) in which learning information is grouped. It is associated to a context and is described with a specific formalism.

If we consider a level of granularity to be adequate when the element is small enough to allow for flexible and integrative reuse and big enough to make sense by itself" then a learning object, as defined here above, could be considered as a potential good candidate. We call unit a coherent learning entity of adequate level of granularity.

Description

Two methods are available to facilitate the search and retrieval of educational resources, i.e. indexation and annotation.

Indexation

It consists in describing a document by giving values to a number of predefined fields (often specifications and standards). This information, called metadata, is then stored and linked to the educational resource it describes. To-date the two major standards are LOM (Learning Object Meta-data) (LOM, 2002) and DC (Dublin Core,). The DC is intended to describe any document while the LOM specializes in educational documents. Several other initiatives exist, some of which are adaptations (profiles) of LOM and were designed to reduce its complexity. We can mention: LOMFR (LOMFR, 2006) and ManuEl (De la Passadière & Jarrot, 2004). Recently (MLR, 2005), an initiative under the name of MLR (Metadata for Learning Resources) has been launched to develop a standard metadata addressing LOM drawbacks and offering new features.

Annotation

The annotation is "a commentary on an object as the commentator said it was noticeably distinguishable from the object itself and the reader interprets as noticeably distinguishable from the object itself" (Baldonado et al., 2000).

(Huart et al., 1996) found a correlation between the different methods of annotation and their semantics (or objective).

Although annotation can provide useful semantic information, as long as this information is not properly stored, it remains unsharable. Conversely, segmentation, looked upon as an annotation instance, could contribute to remedy to this shortcoming. In the case of \cite{Wentland}, a detailed segmentation methodology enables to highlight the semantics of any given educational resource by disjoining it in as many presentation chains as concepts¹ addressed in it. Each presentation chain is considered to have the smallest contextually pertinent level of granularity in the framework of the implicated hypermedia. Although relations between presentation chains are preserved after segmentation has taken place they are lost when indexed individually. We here below put forward an example of how we have solved this issue.

Let us consider the context of segmentation in which it can be claimed that the highest level of granularity of a document is the document itself, while the smallest level of granularity is any of its identified presentation chain seen as a learning object per se. We have adapted MLR to describe the relationships between learning objects capitalizing on the semantic

¹ A Concept is a semantic element explicitly defined in the text. Its definition is composed of either already identified concepts or of prerequisites defined elsewhere. It is characterized by a presentation order, a label, a gender, a type, a complexity degree and content.

information obtained when segmenting the document. We distinguish four types of metadata (Figure 1):

- Navigational metadata describes the relationships between presentation chains (MLR:Description:Relation). Navigation is done through a graph of resource anchors called Learning Object Network (Madhour et al., 2006). A Learning Object Network (LON) is an overlay model to represent relationships between learning object as a graph}, the navigational metadata being associated to the anchors.
- Conceptual metadata that comprises all the attributes of a concept other than those belonging to navigational metadata. It is stored, when it exists, in the Description category (MLR:Description:Description).
- Descriptive metadata that gives a global description (expressed in XML \cite{XML}) of the resource. It is associated directly with the resources in one and same file which is uploaded in a Learning Object Repository (LOR).
- Adaptation metadata that aims at personalizing the navigation in a given domain taking into account the environmental constraints (MLR:Contextualization), the security issues (MLR:Access).

4.1.2 Relationship

Pursuing with our previous example, this field is deduced automatically as relational information has been defined during segmentation. Since segmentation is, by far, not the only way to produce learning objects, we have examined the literature and concluded that the other existing relationships are not compatible with one another. This has prompted us to try and define generic relationships that could support any kind of useful semantic links. Ultimately, the Lausanne Domain Model is composed of a set of indexed entities called learning objects related to one another by any pertinent link. The indexation scheme describes:

- The environment in which the learner evolves
- The relations between learning objects
- The Annotation results if a segmentation is performed

Bearing in mind that we aim at being able to deliver a suitable personalized path to a learner, we now need to proceed to explain XUM (eXtended User Model) which is the Lausanne User Model. After considering two main standards in user profile - IEEE/PAPI and IMS/LIP, we propose a mapping from XUM to PAPI and LIP and vice-versa thus favouring the user mobility from one environment to another. We then show how XUM can be mapped to some MLR attributes in order to easily retrieve suitable learning objects and finalize this presentation by considering the responsibility of each actor in the learning process.

Fig. 1. MLR

4.2 User Model

To achieve our goal, we need to gather as much information as possible about the learner to derive and determine a number of specific useful characteristics: a User Model. We have studied two standards of a user model: IEEE/PAPI and IMS/LIP .

The current specification of PAPI splits the learner information into 6 areas: individual information and preference information, performance, relations, portfolio and security. IMS/LIP describes the learner's characteristics to personalize the content. LIP divides the learner information into 9 areas: interest, affiliation, QCL (Qualifications, Certifications and licenses), activity, goal, identification, competency, relationship, security key, transcript and accessibility. Much more detailed than PAPI, LIP nevertheless entitles the learner to modify the attributes of his user model (like PAPI). Because we believe that the responsibility of the learning process should not be entirely delegated to the learner, we think that this possibility should be shared and restricted. We propose therefore to split the learner's attributes into four categories depending on the modification rights: machine driven, learner driven, system driven and tutor driven. For any specific learner to be able to easily retrieve adequate learning objects, and therefore for the system to provide a personalized learning path, we propose to map XUM with some MLR attributes (as a metadata example).

In the following, we describe each category of user model attributes and its mapping with IMS/LIP and IEEE/PAPI on one hand and with MLR fields (as an example of metadata) when necessary on the other hand.

4.2.1 Machine Driven modification category

The Machine driven modification category contains the system properties attributes (such as memory size and processor speed) as well as the learning constraints (such as the delivery mode and accessibility). We think that these attributes are necessary to prevent the system to include in a personalized path elements that cannot be supported by the learner's machine. Security is a learner driven field that will be dealt with later on.

The System properties descriptor indicates the memory size, the power of the processor, the network characteristics (bit rate, etc.). It can be mapped to the technical requirements field (MLR:Access:Technical Requirements).

The Delivery mode defines the document format, i.e. video, image or text as well as the font and font size (for partially-sighted persons for example). It can be mapped to adaptability field (MLR:Access:Adaptability) because it defines the delivery mode (has auditory, has visual).

Accessibility refers to language, disabilities, and preferences. It can be entirely mapped to the accessibility field of IMS/LIP but in so far the Preferences field of IEEE/PAPI is concerned it can be mapped only to XUM:Accessibility:Preferences.

4.2.2 Learner Driven modification category

The Learner driven modification category includes all information that we think a learner can provide such as security, demographic data, interest, affiliation general goal and stereotype.

Security refers here to the learner's security credentials, such as passwords, challenge/responses, private keys and public keys. It can be mapped entirely to the security field of IEEE/PAPI and IMS/LIP.

Affiliation relates to membership of professional organisations. Only the correspondence with the affiliation field of IMS/LIP is identified.

Goal: The learner can only define his general goal. All sub-goals are determined by the instructor. In fact, in the case of traditional learning, the student can choose within the curriculum, the courses he wishes to attend but he can never go down to select chapters and sections. The course organisation must remain the instructor's responsibility. Only the correspondence with the goal field of IMS/LIP is identified.

This field is of paramount importance as it is the basis of the adaptation process and our aim is to deliver a well-suited personalized learning path (a set of suitable interconnected units).

Demographic data corresponds to all personal information relevant to learning such as age, gender, name, address, role, etc. The identification field in the case of IMS/LIP consists of both biographic and demographic data. This biographic data will be mapped to the stereotype field which will be described later. The Personal field in the case of IEEE/PAPI can be mapped entirely to this field.

It can be mapped to the audience field (MLR:Contextualization:audience).

Interest contains information about hobbies and recreational activities. Only the correspondence with the goal field of IMS/LIP is identified. It can be mapped to the subject field (MLR:Description:Description:Subject)

4.2.3 System Driven modification category

The System driven modification category includes the interaction history, portfolio and proficiencies.

Interaction history: For each user and item there is an annotation that indicates the state of a learning item: read, unread, knowledge (or proficiency), waiting (for learning a special prerequisite).

Relationship (IMS/LIP) and relations fields can be mapped to this field. The only difference is that those relationships have not specific terminology.

Portfolio It is a collection of a learner's accomplishments and works that is intended to illustrate and confirm abilities and achievements.

It can be mapped entirely to the transcript field of IMS/LIP and the portfolio field of IEEE/PAPI.

We are now in a position to describe how we use the information stored in the XUM to retrieve suitable learning objects: we first get a set of possible candidates from which we choose the elements to be ultimately retained to be included in the learning path to be proposed to the learner.

4.3 Adaptation model

Based on an active user model (individual adaptation) and on other users' profiles (social adaptation), the Adaptation model describes how the adaptation is performed.

The individual adaptation process filters items based on the user's needs as they are mainly recorded in the fields of the machine driven modification category. It is used when calculating the next unit to be visited. The Social adaptation process aims at providing a personalized learning path based on the experience of other learners provided they share a similar knowledge level (proficiencies) and the same interests.

Our algorithm includes both processes and aims at building a suitable learning path for a specific learner.

To generate this personalized learning path, we propose to use the Ant Colony optimization algorithm (ACO) (Dorigo et al. , 1999) that has shown its worth when applied to routing problems (connected and disconnected mode), as well as dynamic problems such as travelling salesman problem, coloured graphs, and others.

In our case, we try to use XUM as a way to model the learner behaviour. Our algorithm takes as input an instance of XUM denoted X and the current goal of X . It works based on a set of interconnected units where each unit has a routing table associated a fitness function that guides learners to select the best step to do next.

4.3.1 Learner classification

Although students have heterogeneous knowledge level, it is possible to create temporary groups taking into account only the prerequisite associated with the unit or the goal. Group (x) function returns as result the group to which x belongs.

4.3.2 The routing table

Associated to each unit, the routing table gives an idea about the possible neighbours and probabilities that the learner might move toward each of these neighbours. These probabilities depend on the group the learner belongs to as well as on the pheromones spread. The pheromones spread is deferred because it depends on the result generated by the learner following his goal evaluation once his path comes to an end. This result is given by Evaluate(Goal) and takes the form of a number θ with $0 \leq \theta \leq 1$.

The pheromone τ is in our case the average of θ s evaluated for each member of the defined group.

The pheromones spread of a unit is:

$$\tau_i = \frac{i}{N} \tau \quad (1)$$

With N the number of units.

4.3.3 The fitness function

The fitness function allows to assign probabilities to the units likely to be visited by the learner. The highest probability is that relating to the most suitable unit. It takes the following form:

$$f(a_{u_1, u_2}, x) = \alpha \cdot I(u_2, x) (\beta \cdot \tau_{u_2} - \gamma \cdot W) \quad (2)$$

where:

a_{u_1, u_2} Is the entrance u_2 of u_1 's routing table.

x Is the learner.

I is the individualization factor, it is a function of the environment and the learner characteristics (his preference for example).

τ_i Is the pheromone filled earlier by learners of the same group.

W is the relative degree of complexity.

α , β and γ are weights to be determined according to the needed results. They are parameterized by a human being.

To further clarify our train of thoughts, we demonstrate here below how I is calculated with U being an indexed unit of the network and X being an instance of XUM for a learner X (Figure 2). According to mapping rules defined previously, we can deduct that U suits x. So $I = 1$.

Fig. 2. Example of mapping between XUM and MLR

4.3.4 Path Delivery

Learning path delivery algorithm
Procedure Parcours(X, Goal) Begin AdequatePath=null; Update(X); Ucourante=Source(Group(X), Goal); While (Ucourante \$<>\$ SubGoal(Goal)) Do\ A=ReadRoutingTable(Groupe(X)); P=calculateFitness(A,X); UNext=Choice(P); AdequatePath= AdequatePath U UNext; GoTo(UNext); Update(X); EndWhile Evaluate(Goal); PutPheromone(AdequatePath); UpdateRoutingTable(Group(X)); End

The basic assumption is that each student has a set of information describing him. He has a specific goal that he shares or not with other students. Learning is asynchronous because everyone has a different learning pace and different time constraints.

The learner moves from unit to unit in order to achieve his goal.

Because the goal is not a unit in itself, but knowledge to be mastered, the goal unit is not defined before hand. It must be determined dynamically according to the learner's progress.

The SubGoal function determines the unit to be reached in order to achieve the goal. The fact remains that all learners belonging to the same group have the same starting point called Source that is determined by the function Source (group, Goal).

The purpose of our algorithm is to build the suitable learning path for a learner x . The proposed algorithm follows the meta heuristic pseudo code of ACO in the case of deferred spread of pheromones. We believe that the Ant colony algorithm may provide an optimal solution to the problem of learning path delivery if we define properly the fitness function .

5. Conclusion

In this article, we have described the Lausanne Reference Model, designed for learning object systems. The Domain Model is described as a set of indexed learning objects. We have used the Phoenix tool (Fernandes et al. 2005) that allows segmenting on the fly any hypermedia document and dynamically build a pedagogical network of presentation chains that we have in turn indexed with MLR.

The User Model, baptized XUM, is based on the user profile two main standards, the IEEE/PAPI and IMS/LIP. In order to validate the usability of XUM, we have shown that it can be mapped with some MLR attributes (as a metadata example) facilitating the retrieval of suitable learning objects based on XUM and MLR. The Adaptation Model is based on the ACO algorithm which has the advantage of benefiting from the social dimension and provides an optimal learning path.

The Lausanne Model differs from other existing models like AHAM, Munich, Amsterdam and Dexter mainly in that:

- It considers learning issues such as granularity level, description formalism, quality, intellectual property. To be noted that the two latter are raised in the context of an open environment which is not in the scope of this article.
- Learning objects are organized in a network where links are pertinent but not limited to a particular ontology.
- It enhances user mobility from one environment to another.
- It considers both individual and social adaptation.

Future work will consist in simulating the implementation of the algorithm in different situations to determine the best weights of the fitness function. Besides, the integration of the Lausanne Model based system in a learning environment remains to be done. This system, applied in a lifelong learning process, could bring a real added-value if coupled to a knowledge portfolio.

6. References

- AFNOR. Disponible sur internet à l'adresse:
<http://www.educnet.education.fr/articles/lom-fr.htm>, 2006.
- Albrecht, F. ; Koch, N. & Tiller, T. (1998). Making web-based training more effective, *Proceedings of ABIS-98 : workshop on adaptivity and user modeling in interactive software systems*
- Brusilovsky, P. & Cooper, D. (1999). Adapts : Adaptive hypermedia for a web-based performance support system, *Proceedings of the 2nd Workshop on Adaptive Systems and User Modeling on the WWW*, pp. 41-47, Canada.
- Baldonado, M.; Cousins, S.; Gwizdka, J. & Paepcke, A. (2000). Notable : At the intersection of annotations and handled technologies, *HUC*, pp. 100-113, Bristol.
- Beaumont, I. (1994). User modeling in the interactive anatomy tutoring system ANATOMTUTOR, *User Models and User Adapted Interaction*, 4 (1), 21-45.
- Burke, R.D. ; Hammond, K.J. & Young, B.C. (1996). Knowledge-based navigation of complex information spaces. *AAAI/IAAI*, Vol. 1, pp. 462-468.
- Bielikova, M. & Kuruc, J. (2005). Sharing user models for adaptive hypermedia applications. *In ISDA '05 : Proceedings of the 5th International Conference on Intelligent Systems Design and Applications*, pp. 506-513, Washington, DC, USA.
- Barrit, C. ; Lewis, D. & Wieseler, W. (1999) Cisco systems reusable information object strategy version 3.0. Technical report, CISCO, November 1999. Retrieved from <http://www.cisco.com/warp/public/779/ibs/solutions/learning/whitepapers/elcisco rio.pdf>.
- Brusilovsky, P. (1992) A framework for intelligent knowledge sequencing and task sequencing. *In ITS '92 : Proceedings of the Second International Conference on Intelligent Tutoring Systems*, pp. 499-506, London, UK, 1992. Springer-Verlag.
- Brusilovsky, P. (1996) Methods and techniques of adaptive hypermedia. *User Modeling and User Adapted Interaction*, 6(2-3) , pp. 87-129.
- CDNOW. Disponible sur internet à l'adresse <http://www.cdnnow.com>, 2008.
- De Bra, P.; Houben, G. & Wu, H. (1999) Aham : a dexter-based reference model for adaptive hypermedia. *In HYPERTEXT '99 : Proceedings of the tenth ACM Conference on Hypertext and hypermedia : returning to our diverse roots*, pp.147-156, New York, NY, USA.
- Dorigo, M. ; Di Caro, G. & Gambardella, L.M. (1999). Ant algorithms for discrete optimization. *Artif. Life*, 5(2), pp.137-172.
- Duval, E.; Forte, E. ; Cardinaels, K. ; Verhoeven, B.; Van Durm, R.; Hendrikx, K. ; Wentland Forte, M.; Macowicz, M.; Warkentyne, K. & Haenni, F. (2001). The ariadne knowledge pool system. *Communications of the ACM*, 44(5).
- Dublin Core Meta data initiative. Disponible sur internet à l'adresse <http://dublincore.org/>, 2008.
- De La Passardière, B. & Jarraud, P. (2004). Manuel, un profil d'application de lom pour c@mpuscience. *STICEF - Sciences et Technologies de l'Information et de la Communication pour l'Education et la Formation*.
- Dodds, P. (2001). Advanced distributed learning sharable content object reference model version 1.2, the scorm content aggregation model.
- Espinoza, F. & C. Höök. (1996). An interactive www interface to an adaptive information system. *In Proceedings of User modelling'96 conference*, Kailua-Kona, Hawaii.

- Encarnacao, M. & Stork, A. (1996). An integrated approach to usercentered interface adaptation. technical report WSI-96-10 University of Tübingen.
- Fernandes,E. ; Madhour, H.; Miniaoui, S.& Wentland, M. (2005). Phoenix : An XML based tool for Elearning environment, *Proc. EdMedia05*, Montréal, Canada, 3439-3446.
- Fischer,G.; Mastaglio,T. ; Reeves, B. & Rieman, J. (1990) Minimalist explanations in knowledge-based systems, *Proc. 23rd Annual Hawaii International Conference on System Sciences*, Kailua-Kona, HI, pp. 309-317.
- Friesen, N. ; Roberts, N. & Fisher, S. (2002). Cancore : Learning object metadata.
- Fesenmaier, D. R.; Ricci, F.; Schaumlechner,E. ; Wöber, K. & Zanellai, C. (2003). Dietorecs : Travel advisory for multiple decision styles. *In Proceedings of Enter conference*, Helsinki, Finland.
- Ghani, R. & Fano, A. (2002). Building recommender systems using a knowledge base of product semantics. Technical report, Accenture technology labs.
- Gonschorek,M. & Herzog, C. (1995). Using hypertext for an adaptive helpsystem in an intelligent tutoring system, *Proc. 7th World Conference on Artificial Intelligence in education*, Washington, DC, pp. 274-281.
- Gutierrez,J. ; Pérez,T.; Usandizaga,I. & Lopistéguy, P.(1996). Hypertutor : Adapting hypermedia systems to the user. *In Proceedings of User modelling'96 conference*, Kailua-Kona, Hawaii.
- Hardman,L. D.; Bulterman,C.A. & Van Rossum, G. (1993). Authoring multimedia in the cmif environment. *In CHI '93 : INTERACT '93 and CHI '93 conference companion on Human factors in computing systems*, pp. 101-102, New York, NY, USA.
- Hardman,L. D.; Bulterman,C.A. & Van Rossum, G. (1994). The amsterdam hypermedia model : adding time and context to the dexter model. *Commun. ACM*, 37(2) , pp. 50-62.
- Heift, T. & Nicholson, D. (2001). Web delivery of adaptive and interactive language tutoring. *International Journal of Artificial Intelligence in Education*, 12(4), pp. 310-324.
- Halasz, F. & Schwartz, M. (1990). The dexter hypertext reference model. *In Proc. NIST Hypertext Standardization Workshop*, pp. 95-133, Gaithersburg.
- Hernandez ,J. & Noguez, J. (2005). Affective Behavior in Intelligent Tutoring Systems for Virtual Laboratories, *Proc. 7th World Conference on Artificial Intelligence in Education*, Amsterdam, Netherlands, 51-56
- Huart , P.(1996). Definition d'un poste de lecture active de documents électroniques. PhD thesis, IRIT, Toulouse.
- ISO. Mlr draft for a standard , <http://mdlet.jtc1sc36.org/doc/sc36wg4n0145.pdf>, 2005.
- Krulwich, B. & Burkey, C.(1997). The infoFinder agent : Learning user interests through heuristic phrase extraction. *IEEE Expert : Intelligent Systems and Their Applications*, 12(5) , pp. 22-27.
- Ken, L. (1995). NewsWeeder : learning to filter netnews. *In Proceedings of the 12th International Conference on Machine Learning*, pp. 331-339. Morgan Kaufmann publishers Inc. : San Mateo, CA, USA.
- Krulwich, B. (1997). Lifestyle Finder : Intelligent user profiling using largescale demographic data. *AI Magazine*, 18, pp. 37-45.
- Koch, N. & Wirsing, M.(2002). The munich reference model for adaptive hypermedia applications. *In AH '02 : Proceedings of the Second International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, pp. 213-222, London, UK, Springer-Verlag.

- L'Allier, J.J. (2003). Frame of reference : Netg's map to the products, their structure and core beliefs. *Journal of Digital Information The Use and Abuse of Reusable Learning Objects*, 3(4). IMS Learner Information Package (LIP). www.imsglobal.org/profiles/index.cfm.
- IEEE Learning Technology Standards Committee (LTSC). Ieee 1484.12.1-2002 draft standard for learning object metadata, 2002.
- Melis, E. ; Andrès, E. ; Büdenbender, J. ; Frishauf, A.; Goguadse, G. ;Libbrecht, P. ; Pollet, M. & Ullrich, C. (2001). Activemath : A web-based learning environment. *International Journal of Artificial Intelligence in Education*, 12(4) , pp. 385-407.
- Miller, B.N. ; Albert, I. ; . Shyong, K. L; Konstan, J.A. & Riedl, J.(2003). Movielens unplugged: Experiences with an occasionally connected recommender system. *In Proceedings of ACM 2003 Conference on Intelligent User Interfaces (IUI'03)* (Accepted Poster), Chapel Hill, North Carolina.
- Madhour, H. ; Fernandes, E. & Wentland Forte, M. (2006). Learning object network : Towards a semantic navigation support. *In Ed-Media world conference on educational multimedia, hypermedia and telecommunications*, Orlando, FL, USA.
- Mitrovic, A.(2003). An intelligent sql tutor on the web. *International Journal of Artificial Intelligence in Education*, 13(2-4) , pp. 171-195.
- Mladenic, D.(1996). Personal webwatcher : Design and implementation.
- IEEE Public and Private Information (PAPI). jtc1sc36.org/doc/36n0186.pdf, 2002.
- Peylo, C.; Teiken, W.; Rollinger, C. & H. Gus.(1999). Der vc-prologtutor : Eine internet-basierte lernumgebung. *Künstliche Intelligenz*, 13(4), pp. 32-35.
- Resnick, P.; Iacovou, N.; Suchak, M. ; Bergstorm, P. & Riedl, J. (1994). GroupLens : An Open Architecture for Collaborative Filtering of Netnews. *In Proceedings of ACM 1994 Conference on Computer Supported Cooperative Work*, pp. 175-186, Chapel Hill, North Carolina.
- Semet, Y. (2003). Application de l'optimisation par colonies de fourmis à la structuration automatique de parcours pédagogiques. Mémoire de fin d'études d'ingénieur. Université de Technologie de Compiègne.
- Shimazu, H. (2002). Expertclerk : A conversational case-based reasoning tool for developing salesclerk agents in e-commerce webshops. *Artif. Intell. Rev.*, 18(3-4) , pp. 223-244.
- Shardanand, U. & Maes, P. (1995). Social information filtering : algorithms for automating word of mouth. *In CHI '95 : Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 210-217, New York, NY, USA. Publishing Co.
- Serengul, A. & Smith-Atakan, G. (1998). Personal notes on intelligent tutoring systems. <http://www.cs.mdx.ac.uk/startpages/serengul/>.
- Shang, Y. ; Shi, H. & Chen, S.. An intelligent distributed environment for active learning. *In WWW '01 : Proceedings of the 10th international conference on World Wide Web*, pp. 308-315, New York, NY, USA.
- Tochtermann, K. & Dittrich, G. (1996). The dortmund family of hypermedia systems. *Journal of Universal Computer Science*, 2(1), pp. 34-56.
- Timm, U. & Rosewitz, M. (1998). Benutzermodellierung in der elektronischen produktberatung- konzept und prototypische realisierung in einer on-line umgebung. *In Proceedings of ABIS-98 : workshop on adaptivity and user modeling in interactive software systems*.

- Vassileva, J. (1996). A task-centered approach for user modelling in a hypermedia office documentation system. *User modelling and user adapted interaction journal*, 6 , pp. 185-223.
- Wagner, E. D.(2002). Steps to creating a content strategy for your organization. *The e-Learning Developers' Journal*.
- Weber, G. & Brusilovsky, P.(2001). Elm-art : An adaptive versatile system for web-based instruction. *International Journal of Artificial Intelligence in Education*, 12(4) , pp. 351-384.

Advances in Learning Processes

Edited by Mary Beth Rosson

ISBN 978-953-7619-56-5

Hard cover, 284 pages

Publisher InTech

Published online 01, January, 2010

Published in print edition January, 2010

Readers will find several papers that address high-level issues in the use of technology in education, for example architecture and design frameworks for building online education materials or tools. Several other chapters report novel approaches to intelligent tutors or adaptive systems in educational settings. A number of chapters consider many roles for social computing in education, from simple computer-mediated communication support to more extensive community-building frameworks and tools. Finally, several chapters report state-of-the-art results in tools that can be used to assist educators in critical tasks such as content presentation and grading.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Hend Madhour and Maia Wentland Forte (2010). Personalized Learning Path Delivery, *Advances in Learning Processes*, Mary Beth Rosson (Ed.), ISBN: 978-953-7619-56-5, InTech, Available from:
<http://www.intechopen.com/books/advances-in-learning-processes/personalized-learning-path-delivery>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2010 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](https://creativecommons.org/licenses/by-nc-sa/3.0/), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen