Karampiperis, P., & Sampson, D. (2005). Adaptive Learning Resources Sequencing in Educational Hypermedia Systems. *Educational Technology & Society*, 8 (4), 128-147.

Adaptive Learning Resources Sequencing in Educational Hypermedia Systems

Pythagoras Karampiperis and Demetrios Sampson
Advanced e-Services for the Knowledge Society Research Unit
Informatics and Telematics Institute, Centre for Research and Technology Hellas
42, Arkadias Street, Athens, GR-15234 Greece

Department of Technology Education and Digital Systems
University of Piraeus, 150, Androutsou Street, Piraeus
GR-18534, Greece
pythk@iti.gr
sampson@iti.gr

ABSTRACT

Adaptive learning resources selection and sequencing is recognized as among the most interesting research questions in adaptive educational hypermedia systems (AEHS). In order to adaptively select and sequence learning resources in AEHS, the definition of adaptation rules contained in the Adaptation Model, is required. Although, some efforts have been reported in literature aiming to support the Adaptation Model design by providing AEHS designers direct guidance or semi-automatic mechanisms for making the design process less demanding, still it requires significant effort to overcome the problems of inconsistency, confluence and insufficiency, introduced by the use of rules. Due to the problems of inconsistency and insufficiency of the defined rule sets in the Adaptation Model, conceptual "holes" can be generated in the produced learning resource sequences (or learning paths). In this paper, we address the design problem of the Adaptation Model in AEHS proposing an alternative sequencing method that, instead of generating the learning path by populating a concept sequence with available learning resources based on pre-defined adaptation rules, it first generates all possible learning paths that match the learning goal in hand, and then, adaptively selects the desired one, based on the use of a decision model that estimates the suitability of learning resources for a targeted learner. In our simulations we compare the learning paths generated by the proposed methodology with ideal ones produced by a simulated perfect rule-based AEHS. The simulation results provide evidence that the proposed methodology can generate almost accurate learning paths avoiding the need for defining complex rule sets in the Adaptation Model of AEHS.

Keywords

Adaptive Educational Hypermedia, LO Sequencing, Personalization, Learning objects

1. Introduction and Problem Definition

"eLearning can be viewed as an innovative approach for delivering well designed, learner-centered, interactive, and facilitated learning environment to anyone, anyplace, anytime by utilizing the attributes and resources of various digital technologies along with other forms of learning materials suited for open, flexible, and distributed learning environment", (Khan, 2001). However, eLearning courses have witnessed high drop out rates as learners become increasingly dissatisfied with courses that do not engage them (Meister, 2002; Frankola, 2001). Such high drop out rates and lack of learner satisfaction are due to the "one size fits all" approach that most current eLearning course developments follow (Stewart et. al., 2005), delivering the same static learning experience to all learners, irrespective of their prior knowledge, experience, preferences and/or learning goals.

Adaptive Educational Hypermedia (AEH) (Brusilovsky, 2001; De Bra et. al., 2004) solutions have been used as possible approaches to address this dissatisfaction by attempting to personalize the learning experience for the learner. This learner empowerment can help to improve learner satisfaction with the learning experience. Towards a general definition of an adaptive educational hypermedia system (AEHS) reflecting the current state-of-the-art, Henze and Nejdl (Henze and Nejdl, 2004) introduced a quadruple (KS, UM, OBS, AM) with the following notation:

- the Knowledge Space (KS), that contains two sub-spaces. The first one, referred to as, the *Media Space* contains educational resources and associated descriptive information (e.g. metadata attributes, usage attributes etc.) and the second, referred to as, the *Domain Model* contains graphs that describe the structure of the domain knowledge in-hand and the associated learning goals.
- the User Model (UM), that describes information and data about an individual learner, such as knowledge status, learning style preferences, etc. The User Model contains two distinct sub-models, one for representing the learner's state of knowledge, and another one for representing learner's cognitive

characteristics and learning preferences (such as learning style, working memory capacity etc.). This distinction is made due to the fact that the first model (*Learner Knowledge Space*) can be frequently updated based on the interactions of the learner with the AEHS. On the other hand, learner's cognitive characteristics and learning preferences are more static, having the same property values during a significant time period.

- the Observations (OBS) which are the result of monitoring learner's interactions with the AEHS at runtime. Typical examples of such observations are: whether a user has visited a resource, the amount of time spent interacting with a given resource, etc. Observations related with learner's behavior are used for updating the User Model.
- the Adaptation Model (AM), that contains the rules for describing the runtime behavior of the AEHS. These rules contain *Concept Selection Rules* which are used for selecting appropriate concepts from the Domain Model to be covered, as well as, *Content Selection Rules* which are used for selecting appropriate resources from the Media Space. These rule sets represent the implied didactic approach of an AEHS.

From the above definition, it is clear that in order to define the runtime behavior of the AEHS, the definition of how learner's characteristics influence the selection of concepts to be presented from the domain model (*Concept Selection Rules*), as well as the selection of appropriate resources (*Content Selection Rules*), is required.

In the literature, there exist several approaches aiming to support the design of these rules by providing either direct guidance to AEHS designers, such as the Authoring Task Ontology (ATO) (Aroyo and Mizoguchi, 2004) and the Adaptive Hypermedia Architecture (AHA) (De Bra and Calvi, 1998; De Bra et. al., 2002), or semi-automatic mechanisms for making the rule design process less demanding, such as the Layered AHS Authoring-Model and Operators (LAOS) (Cristea and Mooij, 2003) and the Adaptive Course Construction Toolkit (ACCT) (Dagger et. al., 2005).

However, still the design of adaptive educational hypermedia systems requires significant effort (De Bra, Aroyo and Cristea, 2004), since dependencies between educational characteristics of learning resources and learners characteristics are too complex to exhaust all possible combinations. This complexity introduces several problems on the definition of the rules required (Wu and De Bra, 2001), namely:

Inconsistency, when two or more rules are conflicting.
Confluence, when two or more rules are equivalent.
Insufficiency, when one or more rules required have not been defined.

The problems of inconsistency and insufficiency of the defined rule sets are responsible for generating conceptual "holes" to the produced learning resource sequence (learning path). This is due to the fact that, even if appropriate resources exist in the Media Space, the conflict between two or more rules (inconsistency problem) or the absence of a required rule (insufficiency problem), prevents the AEHS to select them and use them in the learning resource sequence. As a result, either less appropriate resources are used from the Media Space, or required concepts are not covered at all by the resulting path.

In this paper, we address the design problem of the Adaptation Model in adaptive educational hypermedia systems proposing an alternative to the rule-based design approach. The proposed alternative sequencing method, instead of generating the learning path by populating a concept sequence with available learning resources based on pre-defined adaptation rules, it first generates all possible learning paths that match the learning goal in hand, and then, adaptively selects the desired one, based on the use of a decision model that estimates the suitability of learning resources for a targeted learner. This decision model mimics an instructional designer's decision model on the selection of learning resources (Karampiperis and Sampson, 2004). In order to evaluate the proposed sequencing methodology, we compare the produced learning paths with those produced by a simulated perfect rule-based AEHS, using a specific Domain Model and Media Space. The simulation results provide evidence that the proposed methodology can generate almost accurate sequences avoiding the need for defining complex rule sets in the Adaptation Model of AEHS.

The paper is structured as follows: First, we discuss the generalized architecture of AEHS and present the abstract layers for adaptive educational hypermedia sequencing as they have been proposed in the literature. Then we present the current trends in design tools for adaptive educational hypermedia focusing on the methods used for the definition of the Adaptation Model. In Section 3, we present our proposed methodology for adaptive educational hypermedia sequencing. Finally, we present simulation results of the proposed approach and discuss our findings and the conclusions that can be offered.

2. Adaptive Educational Hypermedia Systems: A Literature Review and Discussion

Current state-of-the-art adaptive educational hypermedia systems such as AHA! (De Bra et. al., 2002), OntoAIMS (Aroyo et. al., 2003), The Personal Reader (Dolog et. al., 2004), WINDS (Kravcik and Specht, 2004), ACCT (Dagger et. al., 2005) are based on the Adaptive Hypermedia Application Model (AHAM) (De Bra, Houben and Wu, 1999).

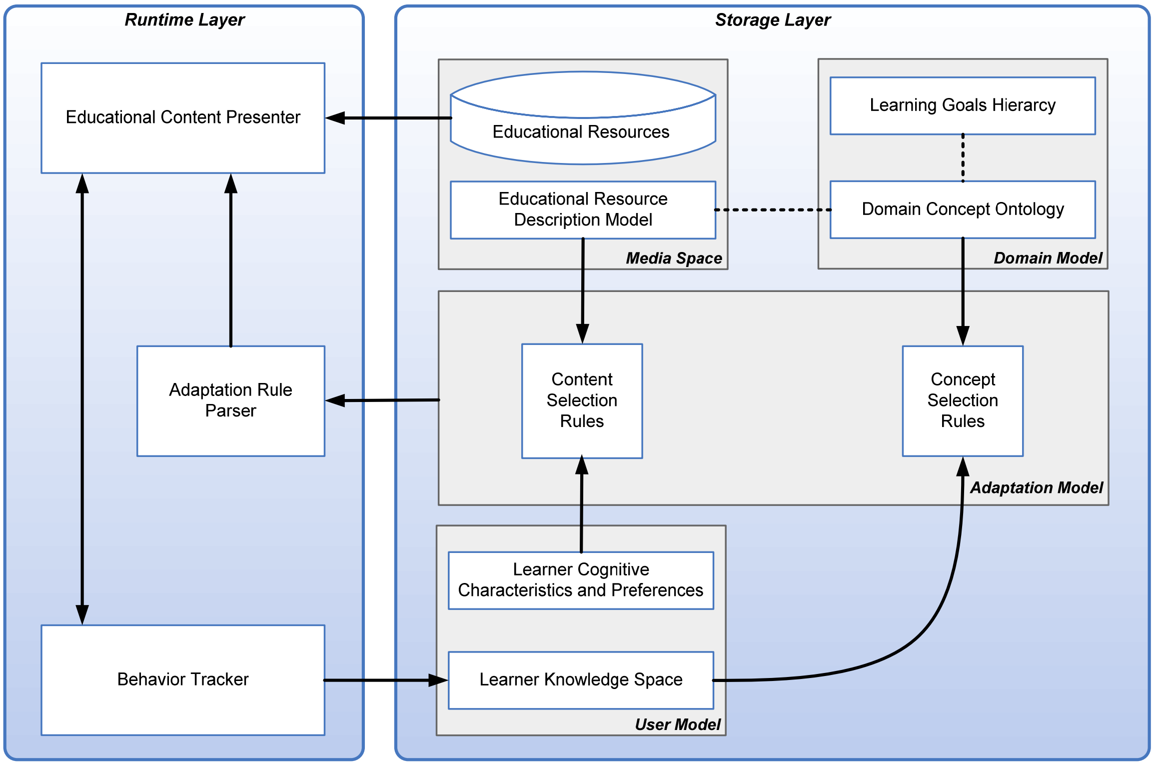


Figure 1: Generalized Architecture of Adaptive Educational Hypermedia Systems

The AHAM builds upon the Dexter model (Halasz and Schwartz, 1994), that is, a common model for hypertext-based systems that was designed for general purpose adaptive web applications. The AHAM model refines the Dexter model so as to be used for educational purposes and extends the hypertext resources to include the full variety of hypermedia objects. The AHAM model consists of two main layers, namely, the *run-time layer* which contains the adaptation engine that performs the actual adaptation and the *storage layer,* which stores information about the Media Space, the Domain Model, the User Model and the Adaptation Model. Figure 1 presents a generalized architecture of an AEHS, presenting the main components of the AHAM model and their structural interconnection. The dashed lines in this figure represent a logical connection between the linked models. According to the above architecture the design process of an AEHS involves four key steps (Brusilovsky, 2003):

- *Designing the Domain Model*, that is, the process of designing a hierarchy of learning goals, as well as, a concept hierarchy (Domain Concept Ontology) for describing the subject domain concepts. For each learning goal specified in the Learning Goals Hierarchy, a set of associated concepts in the Domain Concept Ontology need to be specified. This information is used by the AEHS to determine which concepts need to be covered for reaching a specific learning goal.
- *Designing the User Model*, that is, the process of designing the Learner Knowledge Space, as well as, designing the model for learner's cognitive characteristics and preferences. For the design of the Learner Knowledge Space, there exist two main approaches, the *overlay modeling* (Paiva and Self, 1995) where the learner's state of knowledge is described as a subset of the Domain Concept Ontology and the *stereotype modeling* (Beaumont, 1994) where learners are classified into stereotypes inheriting the same characteristics to all members of a certain class.
- *Designing the Media Space*, that is, the process of designing the educational resource description model. This model describes the educational characteristics of the learning resources e.g. the learning resource type, or its difficulty, as well as structural relationships between learning resources e.g. if a resource requires another resource. For each learning resource contained in the Media Space a set of related concepts from the

Domain Concept Ontology need to be specified. This information is used by the AEHS to determine if a specific learning resource covers a certain concept of the subject domain.

- *Designing the Adaptation Model* that is the process of defining the concept selection rules that are used for selecting from the Domain Model appropriate concepts to be covered, as well as, the content selection rules that are used for selecting appropriate resources from the Media Space. The concept selection rules are defined over the Learner Knowledge Space which represents the learner's state of knowledge by comparing it with the Domain Concept Ontology. The content selection rules are defined over the learner's cognitive characteristics and preferences, relating the educational characteristics of learning resources defined in the educational resource description model with the learner's attributes in the User Model.

After designing the AEHS by following the above mentioned steps, the adaptation engine (Adaptation Rule Parser in Figure 1), is responsible for interpreting the adaptation rules specified in the Adaptation Model in order to generate personalized learning paths. This process is called in the literature *adaptive educational hypermedia sequencing*. Following the previous discussion on the systematic design of AEHS, one could identify three distinct design roles, namely:

- *The Domain Expert*, that is, the person who is responsible for defining the structure of the subject domain (Domain Concept Ontology), the structure of the Learner Knowledge Space, as well as, the concept selection rules of the Adaptation Model.
- *The Instructional Designer*, that is, the person who is responsible for defining the learner cognitive characteristics and preferences of the User Model, the structure of the educational resource description model, as well as, the content selection rules of the Adaptation Model.
- *The Content Expert,* that is, the person who develops the learning resources and structures the Media Space by describing the produced learning resources using the educational resource description model.

In practice, these distinct roles do not operate independently, but, they cooperate for designing some of the system's models. As presented in Table 1, the Domain Expert and the Instructional Designer need to work together for the definition of the Learning Goals Hierarchy, since learning goals are strongly related to the concept and content selection rules. Additionally, the Instructional Designer and the Content Expert need to work together for the definition of the educational resource description model, since, on one hand, this model is used for describing each learning resource developed by the Content Expert and, on the other hand, it is strongly related to the content selection rules defined by the Instructional Designer.

Table 1. Role Participation in the design of AEHS models

	AEHS Models						
	Domain Model		User Model		Educational Resource Description Model	Adaptation Model	
Design Roles	Learning Goals Hierarch	Domain Concept Ontology	Learner Characteristics & Preferences	Learner Knowledge Space		Concept Selection Rules	Content Selection Rules
Domain Expert	X	X		X		X	
Instructional Designer	X		X		X		X
Content Expert					X		

Next section presents the current state-of-the-art tools for designing AEHS that implement the above mentioned abstract sequencing model, focusing on the methods used for the definition of the content selection rules in the Adaptation Model.

2.1 Designing methods of the Adaptation Model in AEHS

Adaptive educational hypermedia sequencing is based on two main processes, namely, the concept selection process and the content selection process. In the *concept selection* process, a set of learning goals from the Learning Goals Hierarchy is selected by the learner e.g. the AIMS (Aroyo and Mizoguchi, 2004), or in some cases by the designer of the AEHS e.g. INSPIRE (Papanikolaou et. al., 2003). For each learning goal, related concepts from the Domain Concept Ontology are selected. These concepts are filtered by the pre-existing knowledge of the learner (Learner Knowledge Space) creating a sequence of missing concepts that need to be covered in order to reach the selected learning goals e.g. the APeLS (Conlan et. al., 2002).

In the *content selection* process, learning resources for each concept of the concept sequence are selected from the Media Space based on the content selection rules that relate the educational characteristics of learning resources with the cognitive characteristics and learning preferences of learners. The result of this process is a personalized learning path that matches the selected learning goals. Typical AEHS examples that utilize this process are the ApeLS (Conlan et. al., 2002) and the MOT (Cristea, 2004b; Cristea and Stewart, in press).

Figure 2 presents the abstract layers of adaptive educational hypermedia sequencing, demonstrating the connection of the above mentioned processes.

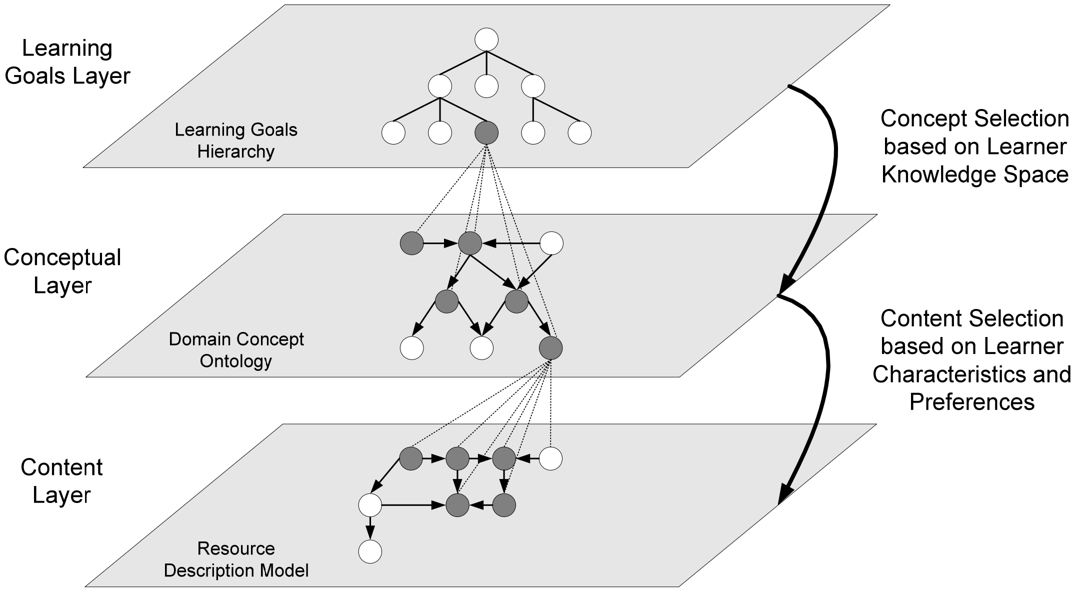


Figure 2: Abstraction Layers of Adaptive Educational Hypermedia Sequencing

In literature, two main approaches appear to be used for the definition of the content selection rules by the AEHS Designers Team, namely, the direct definition and the indirect definition using predefined adaptation patterns. In the direct definition approach, the content selection rules are defined by the Instructional Designer during the design process and they are based on the elements of the Resource Description Model, which is specified through the collaboration with the Content Expert. On the other hand, in the indirect definition approach, predefined adaptation patterns (or templates), which contain both the structure of the educational resource description model and the content selection rules of the Adaptation Model, are selected by the Instructional Designer. Consequently, we can classify the design tools for AEHS recorded in the literature, with regard to their approach for defining the content selection rules, in the following two classes:

- Design Tools supporting the direct definition of the Content Selection Rules. These systems support the Instructional Designer in the process of directly defining content selection rules. They require the Instructional Designer to have good knowledge of the parameters of the system that can be adapted, as well as the details of the User Model. Typical examples of these systems are the AHA! (De Bra and Calvi, 1998; De Bra et. al., 2002), the OntoAIMS (Aroyo et. al., 2003), the AIMS (Aroyo and Mizoguchi, 2004), The Personal Reader (Dolog et. al., 2004), and others.

Although these systems provide graphical environments for the definition of the content selection rules and/or visual representation of the resulting learning/teaching scenario, still it is difficult for Instructional Designers to overcome the problems of inconsistency, confluence and/or insufficiency of the selection rules (De Bra, Aroyo and Cristea, 2004). This is due to the fact that, on one hand, dependencies between educational characteristics of learning resources and cognitive characteristics of learners are rather complex (Cherniavsky and Soloway, 2002; Karampiperis and Sampson, 2004), and on the other hand, it is difficult for an Instructional Designer to know the details of each User Model in use and the corresponding meaningful pedagogical adaptations required (Cherniavsky and Soloway, 2002), since there exist several different models for each learner cognitive characteristic. For example, only in the case that learning styles are used as the main adaptation parameter, there exist more than seventy different models in use (Brown et. al., 2005).

- Design Tools supporting the indirect definition of the Content Selection Rules. These systems use pre-existing adaptation patterns (or templates) that have been a-priori defined by an Instructional Designer during the development phase of the design tool. Typical examples of these systems are the MOT (Cristea, 2004b; Cristea and Stewart, in press), the ACCT (Dagger et. al., 2005), and others.

The main advantage of these systems is that they simplify the design process of adaptive hypermedia, since the educational resource description model and partly the Adaptation Model is predefined. However, when more than one patterns are to be used the AEHS Designers Team is required to know the details of each selected pattern in order to avoid the problems of inconsistency and/or confluence. Additionally, it is nearly impossible for the AEHS Designers Team to extend an existing adaptation pattern, since the definition of new adaptation rules in a pattern would require the AEHS Designers Team to be familiar with the implementation details of the pattern notation language used.

3. The proposed Adaptive Sequencing Methodology

As described in section 2, existing adaptive educational hypermedia systems implement a rule-based sequencing approach based on a two steps procedure. They first generate a sequence of concepts that matches the learning goal in hand, and then select learning recourses for each concept of the concept sequence. Due to the problems of inconsistency and insufficiency of the defined rule sets in the Adaptation Model, conceptual "holes" can be generated in the produced learning resource sequence.

To overcome this problem, we propose an alternative sequencing method that instead of generating the learning path by populating the concept sequence with available learning resources, it first generates all possible sequences that match the learning goal in hand and then adaptively selects the desired personalized learning path from the set of available paths. More precisely, the following two steps procedure is used:

Step1: *Learning Paths Generation*
At this step a graph containing all possible learning paths based on the relation between the Learning Goals Hierarchy, the concepts of the Domain Concept Ontology and the learning resources contained in the Media Space, is generated. This graph is constructed as follows:

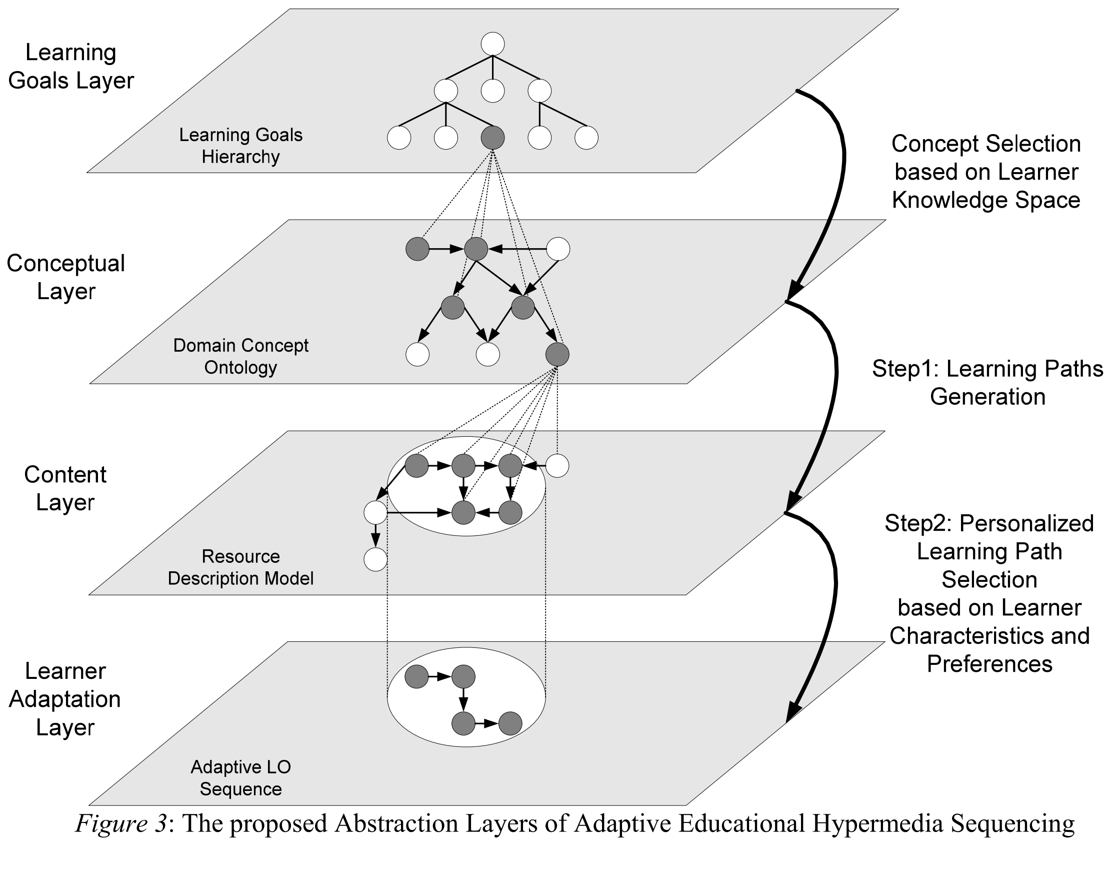


Figure 3: The proposed Abstraction Layers of Adaptive Educational Hypermedia Sequencing

Step1a: *Construction of the Concepts Path Graph.*
The Concepts Path Graph (CPF) is a directed graph which represents the structure of the concepts of the Domain Concept Ontology that matches the learning goal in hand. The concepts contained in the CPF are selected based on the connection between the Learning Goals Hierarchy and the Domain Concept Ontology. The structure of the CPF is directly inherited by the structure of the Domain Concept Ontology. CPF is a simple directed graph, that is, a directed graph having no multiple nodes. This means that each concept is contained only once in the CPF. Additionally, CPF is an acyclic directed graph, that is, a directed graph containing no directed cycles. This means that in every possible concept sequence represented by the CPF, each concept has a unique existence.

Step1b: *Construction of the Learning Paths Graph.*
The Learning Paths Graph (LPG) is a directed graph which represents all possible learning paths (sequence of learning resources) that matches the learning goal in hand. To construct the LPG, for each concept of the CPF related learning resources are selected from the Media Space based on the connection between the Domain Concept Ontology and the Resource Description Model. Each node in the CPF is then replaced by the related set of learning resources retrieved from the Media Space. The structure of the learning resources set is directly inherited by the structure of the Media Space. The final graph is the Learning Paths Graph. Assuming that the Media Space does not contain circular references between learning resources, the LPG is again a simple acyclic directed graph. Although this assumption does not directly affect either the design of an AEHS, nor our sequencing methodology, it is necessary for avoiding infinite learning paths.

Step2: *Personalized Learning Path Selection.* At this step a personalized learning path is selected from the graph that contains all the available learning paths based on learner's attributes in the User Model. As a result, we introduce an additional layer (Figure 3) in the abstract sequencing layers of adaptive educational hypermedia systems, namely the *Learner Adaptation Layer*, which is used for selecting the personalized learning path.

In the proposed sequencing method, we replace the content selection rules defined in the Adaptation Model with a decision-making function that estimates the suitability of a learning resource for a specific learner by relating the educational characteristics of learning resources defined in the educational resource description model with the learner's cognitive characteristics and preferences stored in the User Model. This suitability function is used for weighting each connection of the Learning Paths Graph. From the weighted graph, we then select the most appropriate learning path for a specific learner (personalized learning path) by using a shortest path algorithm. Next sections present the methodology used for creating the suitability function, as well as, for selecting the personalized learning path for a learner.

3.1. Creating the Suitability Function

Next, we present the algorithm for creating a suitability function that estimates the suitability of a learning object for a specific learner. In our previous work, we have proposed a decision model that constructs a *suitability function* which maps learning object characteristics over learner characteristics and vice versa. We have already used that model for the direct selection of learning objects, proving that this suitability function can safely extract dependencies between a User Model and a Resource Description Model (Karampiperis and Sampson, 2004). In that work the User Model elements were not directly defined by the Instructional Designer, but they were dynamically selected from a set of elements during the suitability calculation phase. In this paper, we construct a similar suitability function with the assumption that the elements of the User Model are directly defined by the Instructional Designer and remain the same during the whole life cycle of the AEHS. To this end, before proceeding with the calculation of the suitability function, we assume that the learners' cognitive characteristics and preferences stored in the User Model, as well as, the structure of the Educational Resource Description Model have already been defined by the Instructional Designer. The process of creating the suitability function consists of the following steps, as shown in Figure 4:

Step1: Reference Sets Generation
The first step of the suitability calculation process includes the generation of the reference sets of learning objects and learners that will be used for calculating the suitability function. More precisely, we generate two sets of learning objects, namely, the Learning Objects Training Set (LOTS) and the Learning Objects Generalization Set (LOGS), as well as, two sets of learners, namely, the Learners Training Set (LTS) and the Learners Generalization Set (LGS). The two training sets (LOTS and LTS) are used for calculating the suitability function, and the two generalization sets (LOGS and LGS) are used for evaluating the consistency of the produced suitability function.

Each one of the generated reference learning objects has a unique identifier of the form LO_i and is characterized by a set of n independent properties $g^{LO_i} = (g_1^{LO_i}, g_2^{LO_i}, \ldots, g_n^{LO_i})$ of the Educational Resource Description Model. Similarly, each one of the generated reference learners has a unique identifier of the form L_j and is characterized by a set of m independent properties $u^{L_j} = (u_1^{L_j}, u_2^{L_j}, \ldots, u_m^{L_j})$ of the User Model. The reference learning objects are randomly generated with normal distribution over the value space of each metadata element of the Resource Description Model. Similarly, the reference learners are randomly generated with normal distribution over the value space of each learner characteristic of the User Model.

Step2: Reference LO rating by the Instructional Designer

For each reference learner L_j contained in the LTS, we ask the Instructional Designer to define his/her preference rating of the reference learning objects contained in LOTS, as well as, to define his/her preference rating of the reference learning objects contained in LOGS. These preference ratings are expressed using two preference relations, namely, the strict preference relation and the indifference relation. A strict preference relation means that a learning object is preferred from another one and an indifference relation means that two learning objects are equally preferred. Additionally, for each reference learner L_j contained in the LGS, we ask the Instructional Designer to define his/her preference rating of the reference learning objects contained in LOGS.

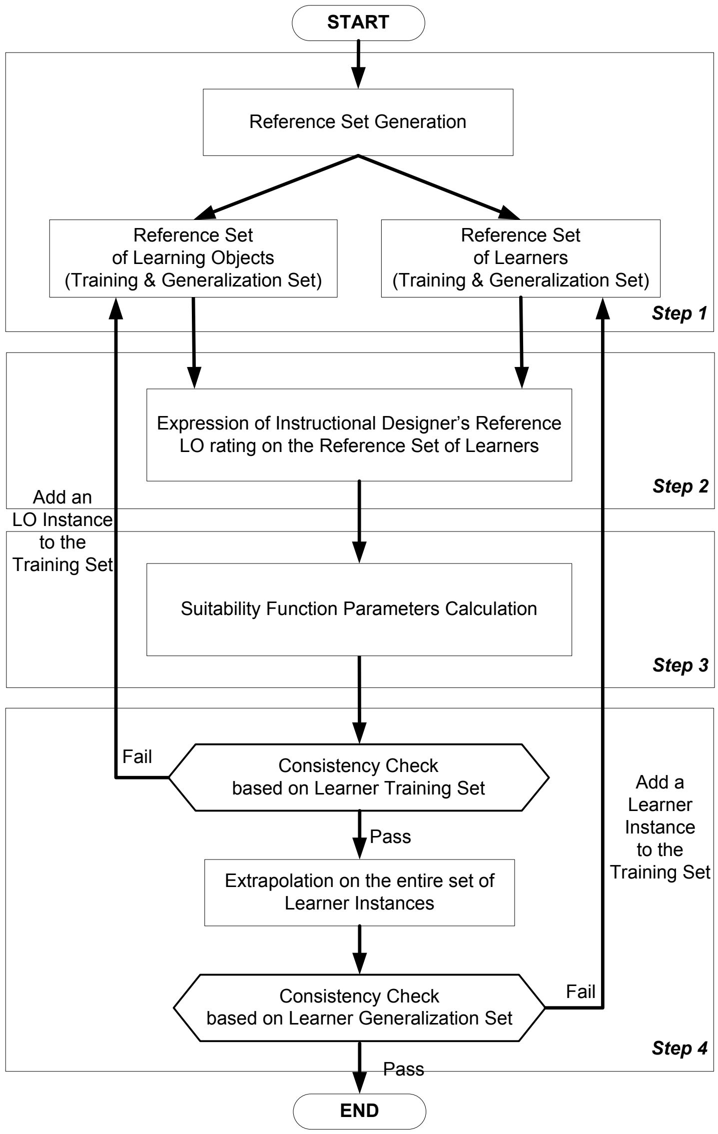


Figure 4: Suitability Function Creation Workflow

Step3: Suitability Function Parameters Calculation

For a specific learner L_j we define as marginal suitability function of the Resource Description Model property g_k a function that indicates how important is a specific value of the property g_k when calculating the suitability of a learning resource LO_i for the learner L_j. This function has the following form (Karampiperis and Sampson, 2004):

$s_{g_k}^{L_j}(g_k^{LO_i}) = a_{g_k}^{L_j} + b_{g_k}^{L_j} g_k^{LO_i} \exp(-c_{g_k}^{L_j} {g_k^{LO_i}}^2)$, where $g_k^{LO_i}$ is the property value of learning object LO_i in the g_k element of the Resource Description Model and $a_{g_k}^{L_j} \in R,\ b_{g_k}^{L_j} \in R,\ c_{g_k}^{L_j} \in R$ are parameters that define the form of the marginal suitability function. The calculation of these parameters for all g_k properties of the Resource Description Model lead to the calculation of the suitability function for the learner L_j.

More precisely, for a specific learner L_j we define the *suitability function* as the aggregation of the marginal suitability functions for the learner Lj, as follows:

$S^{L_j}\left(g^{LO_i}\right) = \frac{1}{n}\sum_{k=1}^{n} s_{g_k}^{L_j}(g_k^{LO_i})$ with the following additional notation:

$s_{g_k}^{L_j}(g_k^{LO_i})$: Marginal suitability of the g_k element of the Resource Description Model, valued $g_k^{LO_i}$ for the learning object LO_i ,

$S^{L_j}\left(g^{LO_i}\right)$: The global suitability of the learning object LO_i for the learner L_j.

If $S_{LO_1}^{Lj}$ is the global suitability of a learning object LO_1 and $S_{LO_2}^{Lj}$ is the global suitability of a learning object LO_2 for the learner L_j, then the following properties generally hold for the suitability function *S*:

$$S_{LO_1}^{Lj} > S_{LO_2}^{Lj} \Leftrightarrow (LO_1)P(LO_2)$$
$$S_{LO_1}^{Lj} = S_{LO_2}^{Lj} \Leftrightarrow (LO_1)I(LO_2),$$

where *P* is the strict preference relation and *I* the indifference relation in Instructional Designer's preference rating. These properties express that for a specific learner L_j, when a learning object LO_1 is preferred from another learning object LO_2, then the suitability function for LO_1 is greater than the suitability function for LO_2 and vise versa. Similarly, when two learning objects LO_1 and LO_2 have the same preference rating for a specific learner L_j, then they also have the same suitability function value.

Using the provided by the Instructional Designer preference rating of the reference learning objects contained in LOTS, for each reference learner L_j contained in the LTS, we define the *suitability differences* $\Delta^{L_j} = (\Delta_1^{L_j}, \Delta_2^{L_j}, \ldots, \Delta_{q-1}^{L_j})$ for the reference learner L_j, where q is the number of learning objects in the LOTS and $\Delta_l^{L_j} = S_{LO_l}^{L_j} - S_{LO_{l+1}}^{L_j} \geq 0$ the suitability difference between two subsequent learning objects in the rated LOTS. We then define an *error function e* for each suitability difference: $\Delta_l^{L_j} = S_{LO_l}^{L_j} - S_{LO_{l+1}}^{L_j} + e_l^{L_j} \geq 0$. Using Lagrange multipliers and Conjugate Gradient, we can then solve for each one of the learner instances L_j in the LTS the following constrained optimization problem:

Minimize $\sum_{l=1}^{q-1}(e_l^{L_j})^2$ subject to the constraints: $\left.\begin{array}{ll}\Delta_l > 0 & \text{if } (\mathrm{LO}_l)P(LO_{l+1}) \\ \Delta_l = 0 & \text{if } (\mathrm{LO}_l)I(LO_{l+1})\end{array}\right\}$ and

$$0 \leq s_{g_k}^{L_j}(g_k^{LO_i}) \leq 1,\ \forall g_k$$

This optimization problem leads to the calculation of the values of the parameters a, b and c for each g_k property of the Resource Description Model over the instances of the LTS, that is, for each separate learner profile included in the LTS.

Step4: Consistency Check and Extrapolation

We then evaluate the consistency of the resulting suitability function, that is, the evaluation of how well the suitability function works for learning objects and/or learners that have not been used in the suitability function parameters calculation (step 3). To this end, we first use the provided by the Instructional Designer preference rating of the reference learning objects contained in LOGS, for each reference learner L_j contained in the LTS.

For a reference learner L_j, we estimate using the suitability function calculated in the previous step (step 3) the Instructional Designer's preference rating of each learning object contained in LOGS. We then compare the provided by the Instructional Designer preference rating with the estimated one. If the preference rating estimation of a learning object LO_i in LOGS is different than that provided by the Instructional Designer, we add the learning LO_i in the Learning Object Training Set (LOTS) and recalculate the suitability function parameters (step 3).

If the estimated and the provided preference ratings are the same, then we generalize the resulted suitability function from the LTS to all learners, by calculating the corresponding suitability values for every learner property $u_z^{L_j}$, using the following linear interpolation formula:

$$s_{g_k}^{L_j}\left(g_k^{LO_i}\right)=\begin{cases} s_{g_k}^{L_1}\left(g_k^{LO_i}\right), \; if \; s_{g_k}^{L_1}\left(g_k^{LO_i}\right)=s_{g_k}^{L_2}\left(g_k^{LO_i}\right) \\ s_{g_k}^{L_1}\left(g_k^{LO_i}\right)+\dfrac{u_z^{L_j}-u_z^{L_1}}{u_z^{L_2}-u_z^{L_1}}\left[s_{g_k}^{L_2}\left(g_k^{LO_i}\right)-s_{g_k}^{L_1}\left(g_k^{LO_i}\right)\right], \; if \; s_{g_k}^{L_2}\left(g_k^{LO_i}\right)>s_{g_k}^{L_2}\left(g_k^{LO_i}\right) \end{cases}$$, where

L_1 and L_2 are the learners of the LTS closest (measured by Euclidean distance) to the learner L_j, $u_z^{L_1}$ and $u_z^{L_2}$ are the values of learner property u_z for learners L_1 and L_2 respectively, and $s_{g_k}^{L_1}$ and $s_{g_k}^{L_2}$ are the marginal suitability functions of the Resource Description Model property g_k for learners L_1 and L_2 respectively.

After the extrapolation on the entire set of learner instances, we evaluate again the consistency of the resulting suitability function, using the provided by the Instructional Designer preference rating of the reference learning objects contained in LOGS, for each reference learner L_j contained in the LGS. For a reference learner L_j, we estimate using the suitability function calculated in the previous step (step 3) the Instructional Designer's preference rating of each learning object contained in LOGS. We then compare the provided by the Instructional Designer preference rating with the estimated one. If the preference rating estimation for a learner L_j in LGS is different than that provided by the Instructional Designer, we add the learner L_j in the Learners Training Set (LTS) and recalculate the suitability function parameters (step 3).

3.2. Selecting a Personalized Learning Path

Following our proposed 2-step methodology for adaptive sequencing, in order to be able to select from the Learning Paths Graph the learning path that matches the characteristics and preferences of a specific learner, we need to add learner-related information to the LPG. This information has the form of weights on each connection of the LPG and represents the inverse of the suitability of a learning resource for the specific learner. This means that the higher value a weight in the LPG has, the less suitable the corresponding learning object in the sequence is for a specific learner. For a specific learner Lj we define the weighting function for each directed connection (edge) of the Learning Paths Graph as $W^{L_j}\left(g^{LO_i}\right)=1-S^{L_j}\left(g^{LO_i}\right)\in[0,1]$, where $S^{L_j}\left(g^{LO_i}\right)$ is the global suitability for the learner L_j of the targeted learning object LO_i in the edge.

After weighting the LPG using the weighting function, we need to find the most appropriate learning path for a learner. Since the weights in the LPG are calculated in such a way that the lower value they have the more suitable a learning object is, the calculation of the most appropriate learning path is equivalent to the calculation of the shortest path in the LPG. By relaxing the edges of the LPG according to a topological sort of its vertices (nodes of the graph), we can compute the shortest path.

The algorithm starts by topologically sorting the LPG to impose a linear ordering on the vertices. If there is a path from vertex u to vertex υ, then u precedes υ in the topological sort (Figure 5a).

Let us call V the set of vertices contained in the LPG. For each vertex $\upsilon\in V$, we maintain an attribute d[υ] called shortest-path estimation, which is an upper bound on the weight of a shortest path from source s to υ. Additionally, for each vertex $\upsilon\in V$, we maintain an attribute π[υ] called shortest-path predecessor. We initialize the shortest-path estimates and predecessors using the following values: π[υ]=NIL for all $\upsilon\in V$, d[s]=0, and $d[\upsilon]=\infty$ for $\upsilon\in V-\{s\}$ (Figure 5a). We make just one pass over the vertices in the topologically sorted order. As we process each vertex, we relax each edge that leaves the vertex. The process of relaxing an edge (u,υ) consists of testing whether we can improve the shortest path to υ found so far by going through u and, if so, updating d[υ] and π[υ]. A relaxation step may decrease the value of the shortest-path estimate d[υ] and update υ's predecessor field π[υ] (Figure 5b-g).

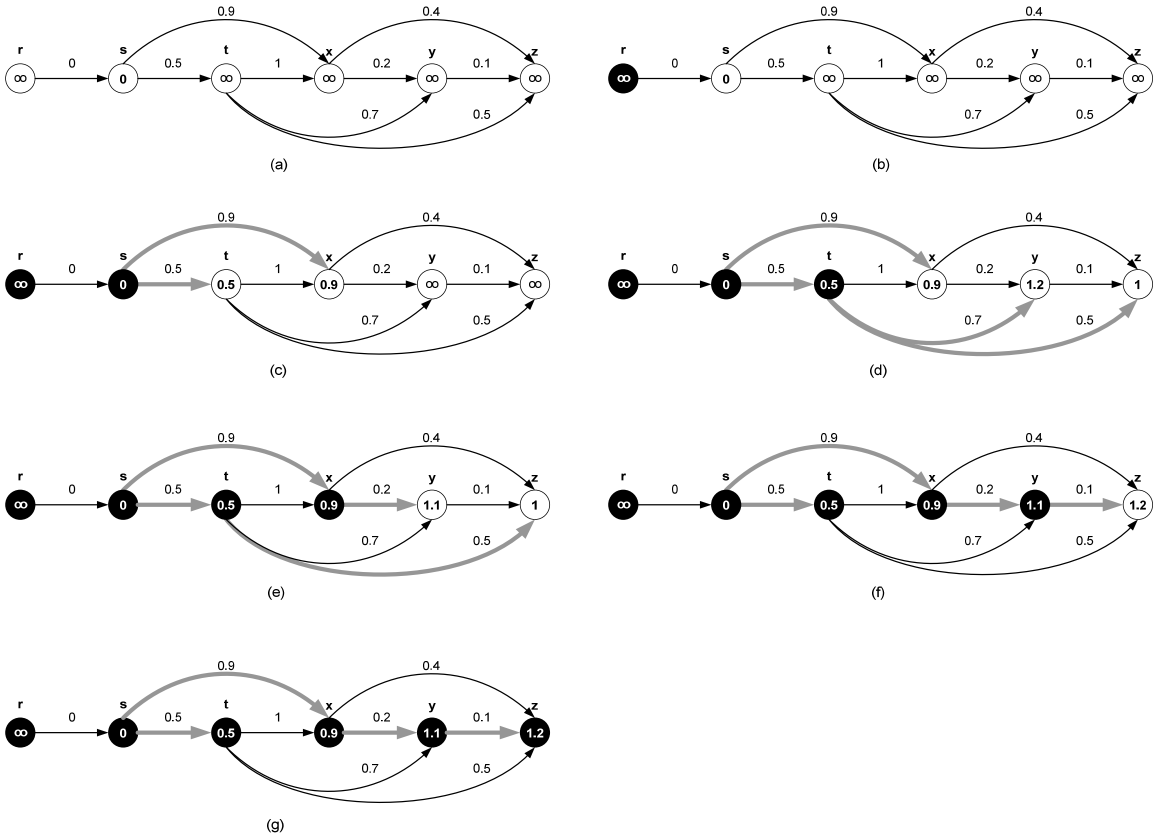


Figure 5: The execution of the algorithm for personalized learning path selection from the LPG. The d values are shown within the vertices, and shaded edges indicate the π values.

The result of this process is the calculation of the shortest path in the LPG that corresponds to the sequence of learning objects that are most suitable for a specific learner L_j.

4. Setting up the Simulation

In this paper we present a sequencing methodology for AEHS that aims to overcome the problem of generating sequences with conceptual holes. In order to evaluate the proposed sequencing methodology, we compare the produced learning paths with those produced by a simulated perfect rule-based AEHS, using a specific Domain Model and Media Space. A perfect rule-based AEHS is assumed to contain consistent and sufficient adaptation rule sets. As a result, it is anticipated that such a system would generate for a desired learning goal, solid learning paths with no conceptual holes. For the simulation of the learning paths produced by a perfect rule-based AEHS, we use a specific Domain Model and Media Space (as described later in this section) and generate for each learning goal specified in the Learning Goals Hierarchy all consistent learning paths that can be defined over the specific Media Space. In our simulations we measure how close the learning paths produced by our proposed methodology are to these ideal paths. By this way, we intent to demonstrate the capacity of the proposed methodology and investigate parameters that influence this performance.

In order to setup our simulations, we use the common design steps of an AEHS, as described in section 2. More specifically:

Designing the Domain Model. The selected domain for our simulations was the Computer Science Domain. For the description of the subject domain concepts, that is, the Domain Concept Ontology, we extracted the ontology from the ACM Computing Curricula 2001 for Computer Science (ACM, 2001). As discussed in section 2, the use of ontologies for structuring the Domain Concept Ontology is commonly used in AEHS, since it provides a standard-based way for knowledge representation (Henze, Dolog and Nejdl, 2004; Aroyo and Dicheva, 2004). The extracted ontology is complete consisting of 950 topics organized in 132 units and 4 areas (see Table 2). A partial view of the concept hierarchy in the domain ontology in use is shown in Figure 6.

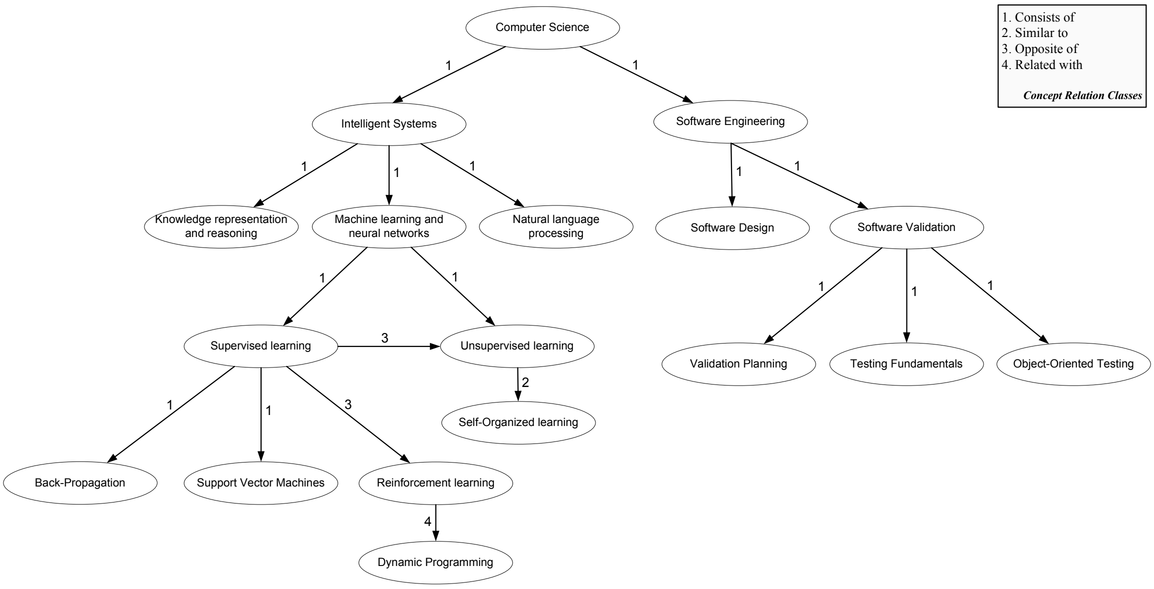


Figure 6: Partial View of Concept Hierarchy in the Domain Concept Ontology in use (ACM Computing Curricula 2001 for Computer Science)

For the description of the relations between the subject domain concepts we used four classes of concept relationships, as shown in Figure 6, namely:

- "Consists of", this class relates a concept with its sub-concepts
- "Similar to", this class relates two concepts with the same semantic meaning
- "Opposite of", this class relates a concept with another concept semantically opposite from the original one
- "Related with", this class relates concepts that have a relation different from the above mentioned
-

Table 2: Subject Domain Concepts covered in the Ontology

Area	Units	Topics
Discrete Structures	6	45
Programming Fundamentals	5	32
Algorithms and Complexity	11	71
Architecture and Organization	9	55
Operating Systems	12	71
Net-Centric Computing	9	79
programming languages	11	75
Human-Computer Interaction	8	47
Graphics and Visual Computing	11	84
Intelligent Systems	10	106
Information Management	14	93
Social and Professional Issues	10	46
Software Engineering	12	85
Computational Science	4	61

Furthermore, for the definition of the Learning Goals Hierarchy in our simulations, we have used again the ACM Computing Curricula 2001 for Computer Science, which defines for each subject domain concept associated learning objectives (ACM, 2001). From this list of learning objectives we have created a Learning Goals Hierarchy which is presented in Figure 7. We then associated each topic of the 950 topics included in the Domain Concept Ontology in use with at least one node of the generated Learning Goals Hierarchy, so as to provide a connection between learning goals and concepts of the particular Domain Concept Ontology in hand.

Designing the User Model. For the design of the User Model in our simulations, we have used an overlay model for representing the Learners Knowledge Space and a stereotype model for representing learners' preferences. More precisely, for the learners' knowledge level we track the existence of a related certification for each node of the Learners Knowledge Space, the evaluation score in testing records and the number of attempts made on the evaluation. For modeling of learners' preferences we use learning styles according to Honey and Mumford model (Honey and Mumford, 1992), as well as modality preference information consisting of three modality types, namely, the visual modality, the textual modality, the auditory modality and the mixed modality preferences. Each element of the User Model was mapped to the IMS Learner Information Package (IMS LIP) specification, as shown in Table 3.

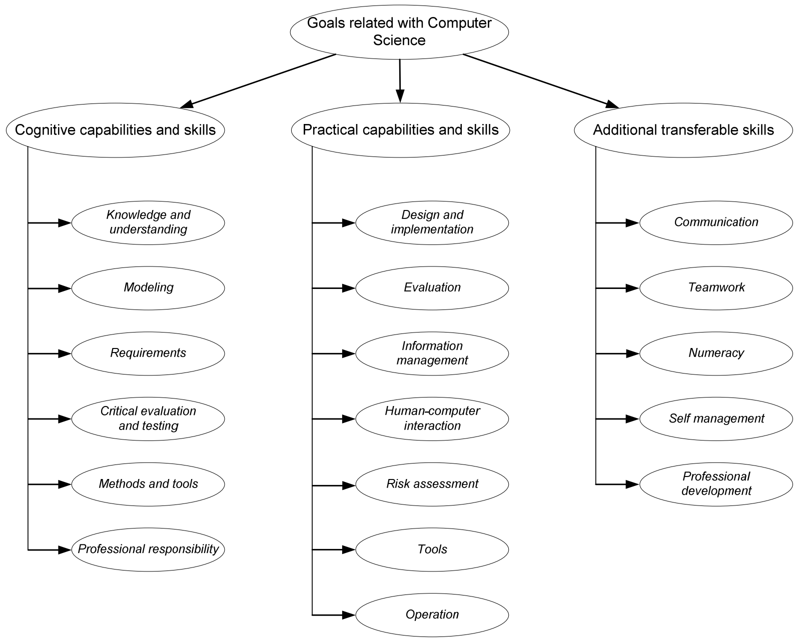


Figure 7: Learning Goals Hierarchy (ACM Computing Curricula 2001 for Computer Science)

Table 3: Using the IMS LIP specification for representing User Model elements

User Model Element	IMS LIP Element	Explanation
Learning Style	Accessibility/Preference/typename	The type of cognitive preference
	Accessibility/Preference/prefcode	The coding assigned to the preference
Modality Preference	AccessForAll/Context/Content	The type of modality preference
Knowledge Level	QCL/Level	The level/grade of the QCL
	Activity/Evaluation/noofattempts	The number of attempts made on the evaluation.
	Activity/Evaluation/result/interpretscope	Information that describes the scoring data
	Activity/Evaluation/result/score	The scoring data itself.

Designing the Media Space. For the design of the Media Space in our simulations we have used as Educational Resource Description Model a subset of the IEEE Learning Object Metadata standard elements, illustrated in Table 4. The Aggregation Level and the Relation/Kind elements are used for structuring the Media Space and the Classification element is used for connecting learning resources with the concepts of the Domain Concept Ontology.

The Aggregation Level was used for classifying the available learning resources in two classes, namely, the raw media and the structured learning objects (Table 5). Each learning resource was tagged with a unique identifier depending on the aggregation level class that it belongs. For example, the identifier of learning resources with aggregation level 1 has the form of AG1:LOi, whereas, the identifier of learning resources with aggregation level 2 has the form of AG21:LOj, where i and j are the unique identifiers of the learning resources inside a specific aggregation class.

In order to define the structure of learning resources at aggregation level 2 (that is, a collection of several learning resources at aggregation level 1) we have used the 'Relation' Category of the IEEE LOM standard.

More specifically, in our simulations we have used eight types of relationships out the 12 predefined values at the Dublin Core Element Set (DCMI, 2004), namely:

- "is part of" / "has part"
- "references" / "is referenced by"
- "is based on" / "is basis for"
- "requires" / "is required by"

Table 4: Educational Resource Description Model used

IEEE LOM Category	IEEE LOM Element	Explanation
General	Structure	Underlying organizational structure of a Learning Object
	Aggregation Level	The functional granularity of a Learning Object
Educational	Interactivity Type	Predominant mode of learning supported by a Learning Object
	Interactivity Level	The degree to which a learner can influence the aspect or behavior of a Learning Object.
	Semantic Density	The degree of conciseness of a Learning Object
	Typical Age Range	Developmental age of the typical intended user.
	Difficulty	How hard it is to work with or through a Learning Object for the typical intended target audience.
	Intended End User Role	Principal user(s) for which a Learning Object was designed, most dominant first.
	Context	The principal environment within which the learning and use of a LO is intended to take place.
	Typical Learning Time	Typical time it takes to work with or through a LO for the typical intended target audience.
	Learning Resource Type	Specific kind of Learning Object. The most dominant kind shall be first.
Relation	Kind	Nature of the relationship between two Learning Objects
Classification	Taxon Path	A taxonomic path in a specific classification system.

Table 5: Learning Objects' Aggregation Level according to IEEE LOM standard

IEEE LOM Element	Value Space	Description
General/Aggregation_Level	1	The smallest level of aggregation, e.g. raw media data or fragments
	2	A collection of level 1 learning objects, e.g. a lesson chapter or a full lesson

A partial view of the Media Space based on the use of the IEEE LOM Aggregation Level element and the Relation/Kind element is presented in Figure 8.

Furthermore, for each learning resource included in the Media Space, a set of related concepts from the Domain Concept Ontology is specified using the Classification element of the IEEE LOM standard. This element describes the position of a specific learning object within a particular classification system and it is typically used in AEHS to determine if a specific learning resource covers a certain concept of the subject domain. Typical systems that used this approach are the Personal Reader (Dolog et. al., 2004), the WINDS (Kravcik and Specht, 2004) and others.

In the literature, several approaches exist that integrate the IEEE LOM metadata elements within domain concept ontologies (Kay and Holden, 2002; Sicilia et. al., 2004; Hayashi, Ikeda and Mizoguchi, 2004; Simon et. al., 2004). The use of the classification element of the IEEE LOM standard, on one hand, models the connection between concepts of the Domain Concept Ontology and the learning resources, and on the other hand, enables the separation of the Educational Resource Description Model from the Domain Concept Ontology. This separation enables the use of separate metadata records for learning resources, thus, enabling the use of resources and associated metadata contained in external from the AEHS repositories.

Designing the Adaptation Model. For the design of the Adaptation Model in our simulations, we have used the methodology presented in section 3 based on a set of 50 learning object metadata records (30 for the Learning Objects Training Set and 20 for the Learning Objects Generalization Set) and a set of 10 simulated learner instances (5 for the Learners Training Set and 5 for the Learners Generalization Set). These sets were used to calculate the suitability function presented in section 3.1 of this paper.

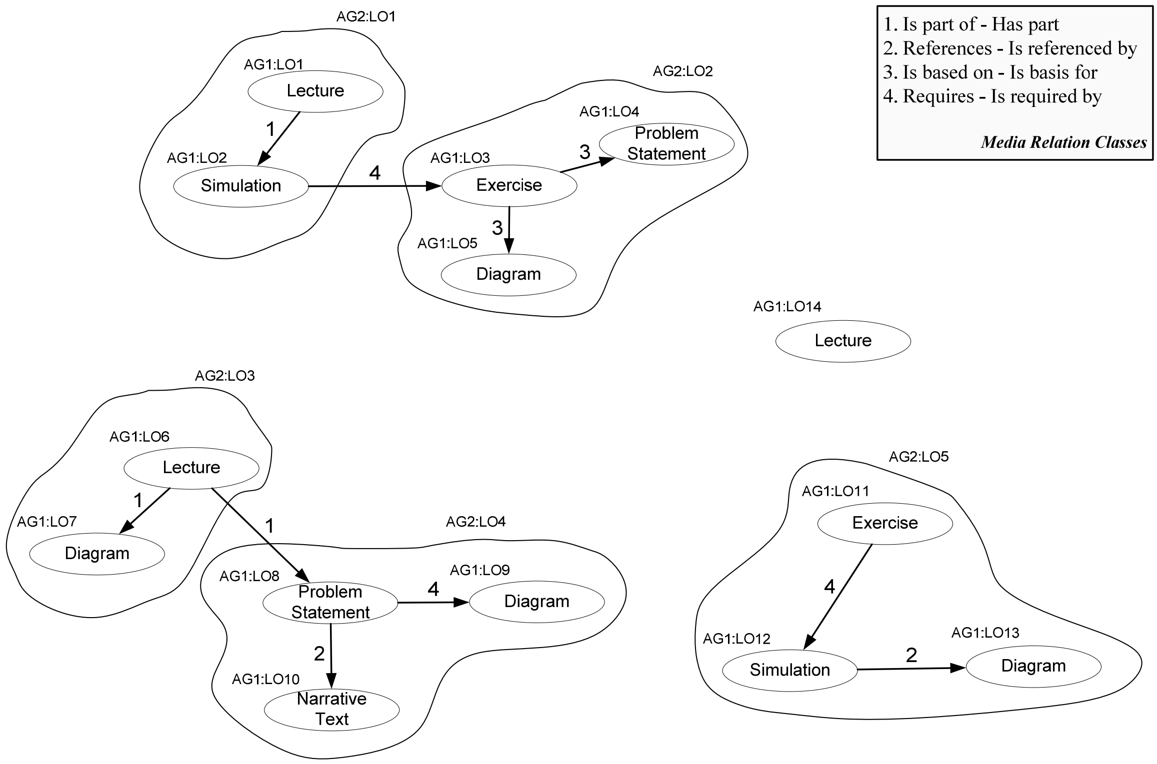


Figure 8: Partial View of Media Space Representation

For our simulations we created an additional set of learning object metadata records that we call Learning Objects Estimation Set - LOES, consisting of 142.500 records (that is, 150 simulated learning objects for each one of the 950 topics), with normal distribution over the value space of each metadata element. Additionally, we created a set of 20 simulated learner instances that we call Learner Estimation Set - LES, with normal distribution over the value space of each learner characteristic. These estimation sets were used for evaluating the efficiency of the proposed approach in generating learning paths with no conceptual holes, as it is discussed in the next section.

5. Simulation Results and Discussion

In our simulations we evaluate the proposed sequencing methodology by comparing the produced learning paths with those produced by a simulated perfect rule-based AEHS, as described in the previous section. To this end, we have defined an evaluation criterion based on Kendall's Tau (Wilkie, 1980), which measures the match between two learning object sequences, as follows:

$$\text{Success}\,(\%) = 100 * \left(\frac{1}{2} + \frac{N_{concordant} - N_{discordant}}{k(k-1)} \right),\ k = \sum_{\forall\, topic\, level} n$$

where $N_{concordant}$ stands for the concordant pairs of learning objects and $N_{discordant}$ stands for the discordant pairs when comparing the resulting learning objects sequence with the ideal reference one and n is the maximum requested number of learning objects per concept level.

The efficiency of the proposed method was evaluated by comparing the resulting learning object sequences with ideal reference sequences for 50 different cases (10 randomly selected learner instances from the Learner Estimation Set per level of sequence root) over the concept hierarchy. Average evaluation results are shown in Figure 9 presenting the success of the proposed sequencing method for different cases of sequence roots (that is, the concept level in Domain Ontology) and different cases of maximum requested number of learning objects per concept level (n). In this figure the different concept levels express the depth in the Domain Ontology of the root concept in the desired learning path. For example, topic levels (1-5) correspond to concepts in the Domain Ontology with depth between one and five. These concepts are included in a Unit (see also Table 2) and they possibly include topics with depth greater than five, depending on the structure of the Domain Ontology.

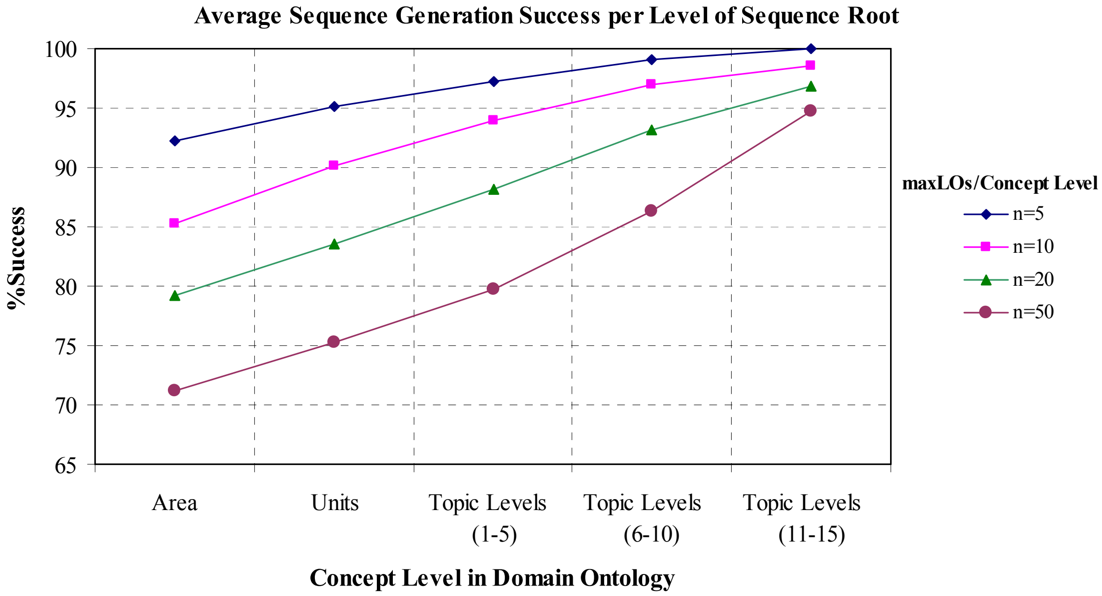


Figure 9: Average Simulation Results for Learning Path Selection

From these results we conclude that the success rate of the resulting learning object sequences is depending on the concept levels that the end sequence covers, as well as the maximum requested resources for each level. The less number of resources per level are requested, the smallest would be the resulted LO sequence, producing less probability of possible mismatches. Accordingly, for the same number of requested objects per level, the higher level the sequence root is, the longer would be the resulted sequence introducing more mismatches from the Learning Paths Graph weighting process.

These observations introduce two main design principles that should be followed in order to successfully generate personalized learning paths, namely:

The Content Expert of an AEHS should design the Media Space by creating structured learning resources (with Aggregation Level equal to 2) rather than raw media. This internal structuring, on one hand, enables the AEHS to select less (but more aggregated) learning resources, and on the other hand, increases the probability of generating meaningful learning paths since less decisions about the structuring of the learning resources are taken by the AEHS.

The end-user of an AEHS should request an adaptive web-based course covering the minimum needed parts of the Domain Concept Ontology, in order for avoiding the generation of huge sequences that introduce mismatches.

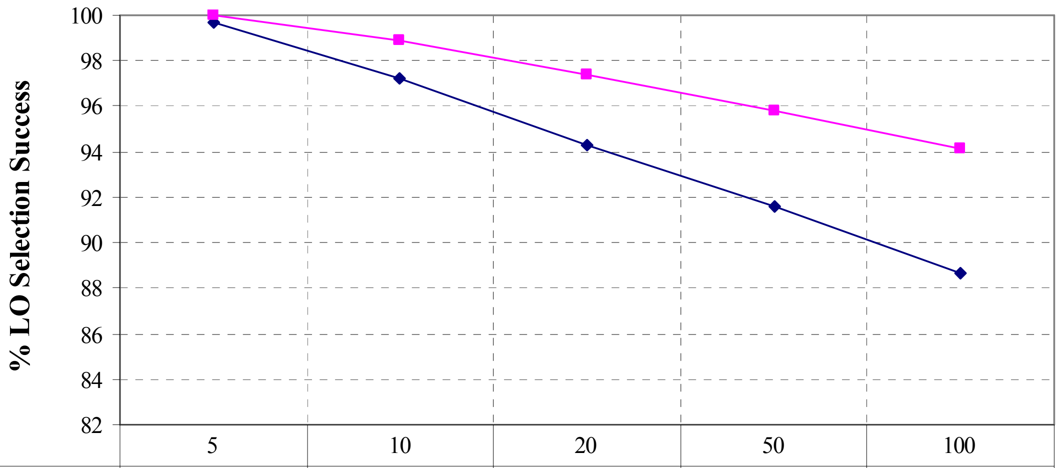
In order to investigate in more detail these mismatches, we have designed another evaluation criterion, which measures the success in selecting appropriate resources per concept node, defined by:

$$\text{Selection Success (\%)} = 100 * \left(\frac{\textit{Correct Learning Objects Selected}}{m} \right)$$

where m is the number of requested learning objects from the Media Space per concept node.

We have evaluated the selection success on two different sub sets of the Learning Objects Estimation Set. The first data sets contains learning object metadata records with aggregation level 1 (raw media) and the second data set contains learning object metadata records with aggregation level 2 (structured learning objects), as defined in section 4. Figure 10 presents average simulation results for learning objects selection.

From these results we can once again confirm the observation that using structured learning objects rather than raw media, increases the probability of generating flawless learning paths. More analysis on the results, presented in Figure 10, shows that when the desired number of learning objects (m) is relatively small (less or equal to 10), the efficiency of selection is almost the same for raw media and structured learning objects. However, when the desired number of learning objects is relatively large (more than 10) the success in selecting learning objects is strongly affected by the aggregation level of the learning objects.



	5	10	20	50	100
Agg. Level 1	99.7	97.2	94.3	91.6	88.7
Agg. Level 2	100.0	98.9	97.4	95.8	94.1

Num of LOs/Concept Node (m)

Figure 10: Average Simulation Results for Learning Objects Selection

If we consider that for one learner instance, the different combinations of learning objects, calculated as the multiplication of the value instances of characteristics presented in Table 4, leads to more than one million learning objects, it is evident that it is almost unrealistic to assume that an instructional designer can manually define the full set of selection rules which correspond to the dependencies extracted by the proposed method and at the same time to avoid the inconsistencies, confluence and insufficiency of the produced selection rules. The simulation results demonstrate that the proposed approach is capable of extracting dependencies between learning object and learner characteristics producing almost accurate sequences of learning objects (that is, almost similar to the ideal ones). Furthermore, it was exhibited that the granularity of learning object sequences, as well as, the aggregation level of the learning objects are the main parameters affecting the sequencing success. A learning path that covers a whole concept area is more likely to produce mismatches when comparing with a sequence that covers only a specific unit or even a specific topic, and a sequence that uses raw media is more likely to produce mismatches when comparing with a sequence that uses structured learning objects.

This is due to the fact that structured learning objects partly contain information about the underlying pedagogical scenario. When only raw media are used for sequencing, then the pedagogical scenario is totally implied in the decisions made by the AEHS. Our future work, focuses on separating the learning scenario from the adaptation decision model. By this way, we anticipate, on one hand, to support better the sequencing of unstructured raw media, and on the other hand, to facilitate the support of different pedagogical strategies without redesigning the adaptation decision model.

6. Conclusions

In this paper, we address the design problem of the Adaptation Model in AEHS proposing an alternative sequencing method that instead of generating the learning path by populating a concept sequence with available learning resources based on adaptation rules, it first generates all possible sequences that match the learning goal in hand and then adaptively selects the desired sequence, based on the use of a decision model that estimates the suitability of learning resources for a targeted learner. In our simulations we compare the produced sequences by the proposed methodology with ideal sequences produced by a perfect rule-based AEHS. The simulation results provide evidence that the proposed methodology can generate almost accurate sequences avoiding the need for defining complex rule sets in the Adaptation Model of AEHS. Additionally, the simulation results showed that the success in learning object sequencing is strongly affected by the aggregation level of the learning objects and the number of the concepts covered by the desired learning object sequence.

7. Acknowledgements

The work presented in this paper is partially supported by European Community under the FP6 Information Society Technologies (IST) programme ICLASS contract IST-507922, and the Leonardo da Vinci programme eAccess contract EL/2003/B/F/148233.

8. References

ACM. (2001). *ACM Computing Curricula 2001 for Computer Science*, Retrieved October 25, 2005, from http://www.acm.org/education/curricula.html.

Aroyo, L., Mizoguchi, R., & Tzolov, C. (2003). OntoAIMS: Ontological Approach to Courseware Authoring. *Paper presented at the International Conference on Computers in Education (ICCE2003)*, December 2-5, 2003, Hong Kong.

Aroyo, L., & Dicheva, D. (2004). The New Challenges for E-learning: The Educational Semantic Web. *Educational Technology & Society, 7* (4), 59-69.

Aroyo, L., & Mizoguchi, R. (2004). Towards Evolutional Authoring Support Systems. *Journal of Interactive Learning Research, 15* (4), 365-387.

Beaumont, I. (1994). User modeling in the interactive anatomy tutoring system ANATOMTUTOR. *International Journal of User Modeling and User-Adapted Interaction, 4* (1), 21-45.

Brown, E., Cristea, A., Stewart, C., & Brailsford, T. (2005). Patterns in Authoring of Adaptive Educational Hypermedia: A Taxonomy of Learning Styles. *Education Technology & Society, 8* (3), 77-90.

Brusilovsky, P. (2001). Adaptive hypermedia. Methods and techniques of adaptive hypermedia. *International Journal of User Modeling and User-Adapted Interaction, 11* (1/2), 87-110.

Brusilovsky, P. (2003). Developing adaptive educational hypermedia systems: From design models to authoring tools. In Murray, T., Blessing, S., Ainsworth, S. (Eds.), *Authoring Tools for Advanced Technology Learning Environment*, Dordrecht: Kluwer Academic Publishers, 377-409.

Cherniavsky, J. C., & Soloway, E. (2002) A survey of research questions for intelligent information systems in education. *Journal of Intelligent Information Systems, 18* (1), 5-14.

Conlan, O., Hockemeyer, C., Wade, V., & Albert, D. (2002). Metadata Driven Approaches to Facilitate Adaptivity in Personalized eLearning Systems. *Journal of the Japanese Society for Information and Systems in Education, 1* (1), 38-44.

Cristea, A., & Mooij de A. (2003). LAOS: Layered WWW AHS Authoring Model and their corresponding Algebraic Operators *Paper presented at the 12th International World Wide Web Conference*, May 20-24, 2003, Budapest, Hungary.

Cristea, A. (2004a). What can the Semantic Web do for Adaptive Educational Hypermedia? *Educational Technology & Society, 7* (4), 40-58.

Cristea, A. (2004b). Is Semi-Automatic Authoring of Adaptive Educational Hypermedia possible? *Advanced Technology for Learning Journal, 1* (4), 227-236.

Cristea, A., & Stewart, C. (In Press). Automatic Authoring of Adaptive Educational Hypermedia. In Ma, Z. (Eds.), *Web-Based Intelligent e-Learning Systems: Technologies and Applications*, IDEA Publishing group.

Dagger, D., Wade, V., & Conlan O. (2005). Personalisation for All: Making Adaptive Course Composition Easy. *Education Technology & Society, 8* (3), 9-25.

DCMI (2004) *Dublin Core Metadata Element Set, Version 1.1*, Retrieved October 25, 2005, from, http://dublincore.org/documents/dces/.

De Bra, P., Houben, G. J., & Wu, H. (1999). AHAM: A Dexter based Reference Model for Adaptive Hypermedia. *Paper presented at the ACM Conference on Hypertext and Hypermedia*, February 21-25, 1999, Darmstadt, Germany.

De Bra, P., & Calvi, L. (1998). AHA! An open Adaptive Hypermedia Architecture. *The New Review of Hypermedia and Multimedia, 4*, 115-139.

De Bra, P., Aerts, A., Smits, D., & Stash, N. (2002). AHA! Version 2.0, More Adaptation Flexibility for Authors. *Paper presented at the World Conference on e-Learning in Corporate, Government, Healthcare & Higher Education (ELearn'2002)*, October 15-19, 2002, Montreal, Canada.

De Bra, P., Aroyo, L., & Cristea, A. (2004). Adaptive Web-based Educational Hypermedia. In Levene, M. & Poulovassilis, A. (Eds.), *Web Dynamics, Adaptive to Change in Content, Size, Topology and Use*, Heidelberg, Germany: Springer, 387-410.

De Bra, P., Aroyo, L., & Chepegin, V. (2004). The Next Big Thing: Adaptive Web-Based Systems, *Journal of Digital Information, 5* (1), Retrieved October 25, 2005, from, http://jodi.tamu.edu/Articles/v05/i01/DeBra/.

Dolog, P., Henze, N., Nejdl, W., & Sintek, M. (2004). The Personal Reader: Personalizing and Enriching Learning Resources Using Semantic Web Technologies. *Paper presented at the Third International Adaptive Hypermedia and Adaptive Web-based Systems Conference (AH2004)*, August 23-26, 2004, Eindhoven, The Netherlands.

Frankola, K. (2001). Why online learners dropout. *Workforce, 10*, 53-63.

Halasz, F., & Schwartz, M. (1994). The Dexter Hypertext Reference Model. *Communications of the ACM, 37* (2), 30-39.

Hayashi, Y., Ikeda, M., & Mizoguchi, R. (2004). A Design Environment to Articulate Design Intention of Learning Contents. *International Journal of Continuing Engineering Education and Life Long Learning, 14* (3), 276-296.

Henze, N., & Nejdl, W. (2004). A Logical Characterization of Adaptive Educational Hypermedia. *New Review of Hypermedia and Multimedia (NRHM), 10* (1), 77-113.

Henze, N., Dolog, P., & Nejdl, W. (2004). Reasoning and Ontologies for Personalized E-Learning. *Educational Technology & Society, 7* (4), 70-81.

Honey, P., & Mumford, A. (1992). *The manual of Learning Styles'*, Maidenhead: Peter Honey.

IEEE (2002). *IEEE Draft Standard for Learning Object Metadata*, IEEE P1484.12.1/d6.4, Retrieved October 25, 2005, from, http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf.

IMS. (2001). *IMS Global Learning Consortium Inc., Learner Information Package (LIP) Final Specification v1.0*, Retrieved October 25, 2005, from, http://www.imsglobal.org/profiles/.

Karampiperis, P., & Sampson, D. (2004). Adaptive Learning Object Selection in Intelligent Learning Systems. *Journal of Interactive Learning Research, 15* (4), 389-409.

Kay, J., & Holden, S. (2002). Automatic extraction of ontologies from teaching document metadata. *In Proc. of the ICCE 2002 Workshop on Concepts and Ontologies in Web-based Educational Systems*, Auckland, New Zealand, 25-28, Retrieved October 25, 2005, from, http://wwwis.win.tue.nl/~laroyo/ICCE2002_Workshop/proc-Workshop-ICCE2002.pdf.

Khan, B. (2001). *Managing E-Learning Strategies: Design, Delivery, Implementation and Evaluation*, Hershey, PA, USA: Idea Group Inc.

Kravcik, M., & Specht, M. (2004). Flexible Navigation Support in the WINDS Learning Environment for Architecture and Design. *Paper presented at the Third International Adaptive Hypermedia and Adaptive Web-based Systems Conference (AH2004)*, August 23-26, 2004, Eindhoven, The Netherlands.

Meister, J. (2002). *Pillars of e-learning success*, New York, USA: Corporate University Exchange.

Paiva A., & Self J. (1995). TAGUS - A User and Learner Modeling Workbench. *International Journal of User Modeling and User-Adapted Interaction, 4* (3), 197-226.

Papanikolaou, K., Grigoriadou, M., Kornilakis, H., & Magoulas G. (2003) Personalising the Interaction in a Web-based Educational Hypermedia System: the case of INSPIRE. *International Journal of User Modeling and User-Adapted Interaction, 13* (3), 213-267.

Sicilia, M. A., Garcia, E., Diaz, P., & Aedo, I. (2004). Using links to describe imprecise relationships in educational contents. *International Journal of Continuing Engineering Education and Life-Long Learning (IJCEELL), 14* (3), 260–275.

Simon, B., Dolog, P., Miklos, Z., Sintek, M., & Olmedilla, D. (2004). Conceptualising Smart Spaces for Learning. *Journal of Interactive Media in Education, 9* (8), Retrieved October 25, 2005, from, http://www.l3s.de/~olmedilla/pub/jime04.pdf.

Stewart C., Cristea A., Brailsford T., & Ashman H. (2005), Authoring once, Delivering many: Creating reusable Adaptive Courseware. *Paper presented at the Web-Based Education Conference WBE'05*, February 21-23, 2005, Grindelwald, Switzerland.

Wilkie, D. (1980). Pictorial Representation of Kendall's Rank Correlation Coefficient. *Teaching Statistics, 2*, 76-78.

Wu, H., & De Bra, P. (2001). Sufficient Conditions for Well-behaved Adaptive Hypermedia Systems. *Lecture Notes In Computer Science, 2198***,** 148-152.