Authoring of Adaptive Hypermedia;

Adaptive Hypermedia and Learning Environments

Abstract: This chapter focuses on the aspect of *Authoring* in *Adaptive Hypermedia*, from some of its different perspectives, including authoring for learning environments. It starts by showing the necessity of research in this area, then describes a new framework model for authoring of Adaptive Hypermedia, LAOS. Within LAOS, the adaptation model, which is the main aspect of adaptive hypermedia, is detailed into a separate model, LAG. The flexibility offered by the LAOS framework is analyzed and estimated. To illustrate the theory, the chapter describes an implementation of this framework, MOT, and test results. The chapter ends with conclusions and some discussion on future trends.

KEYWORDS: Adaptive Hypermedia, Authoring of AH, User Modelling

INTRODUCTION

Adaptive Hypermedia (AH, Brusilovsky 2002) is here, and researchers in the field (Bajraktarevic et al 2003, Brailsford et al 2002) hope that it is here to stay. Although a relatively new field (dating back only to the early 1990s), it has taken on board the advantages, whilst avoiding the pitfalls of its parent disciplines, Intelligent Tutoring Systems and User Modelling. An advantage it shares is offering a *personalized* *environment* (adaptive or adaptable[1]). Moreover, AH moves this environment to the web. The main pitfall that it managed to avoid is complexity: traditional AH systems are simple, built on sketchy user models, mostly featuring a knowledge attribute overlaid on a simple domain model. This *simplicity* gives it the power of fast response and wide usage range.

From an authoring perspective, however, it turns out that efficient AH is not at all simple to design. Even with basic domain and user models, creating a powerful adaptive environment requires many alternatives of contents, linking, etc. Furthermore, granularity of information chunks, alternative display modes, etc., have to be taken into consideration.

Therefore, our main aim is to create a framework for powerful, flexible authoring tools for authors of adaptive hypermedia. This main aim is translated in this chapter into requirements of this framework: *data storage* with sufficient metadata labelling for reuse (both for collaborative authoring and adaptive presentation), *data clustering* depending on the intended level of reuse, and '*automatic authoring*', i.e., automatic generation of some default content structure, labelling and behaviour. We shall see how the products of this research also lead to patterns that could be used to extend existent standards (e.g., LOM, simple sequencing standard, SCORM) or even to generate new standards for AH.

The remainder of this chapter is structured as follows. First, we will give more background information on the driving forces behind the research on authoring of adaptive hypermedia systems, as well as a very short glimpse into the state of the art. Next, we present LAOS, a theoretical framework for authoring of AH, that we claim allows enough flexibility to embrace not only the existing adaptive hypermedia

[1] *Adaptivity* implies the system making inferences about possible choices, and then executing them. In *adaptability*, the inferences about possible choices, as well as the selections are made by the user.

systems, but also to establish a solid basis for structured, pattern-based authoring of adaptive hypermedia. The latter is enabled by LAG, the three-layers model of adaptation granularity. We will also show some automatic transformations allowed by LAOS that give it its flexibility. Following that, we describe MOT (My Online Teacher, Cristea & De Mooij, 2003a), a system that is gradually implementing the LAOS framework, and sketch the first tests done with MOT. Finally, we try to extract future trends for this line of research, and conclude.

BACKGROUND

Adaptive hypermedia systems were traditionally custom-designed applications for single use implementing hypermedia-based user models (Brusilovsky, 2002). Only recently, their authoring aspect started being taken into consideration, partly because initial AHs were of small size (Brusilovsky et al 1996). In such systems, reuse wasn't an issue. The interest in authoring shows the field's first steps towards maturity, as authoring first requires widely accepted common characteristics.

There are many other reasons of why the time is now ripe to concentrate on authoring in adaptive hypermedia, instead of on new adaptive hypermedia techniques; such as: the fact that the field is advanced enough; and that we cannot expect any major break-through theoretical advances[2]. Another reason is that there are a number of common features we see repeated in almost all adaptive hypermedia, such as *user model* (Brusilovsky 2002), *knowledge level* (De Bra & Calvi 1998), *goals* (Grigoriadou et al 2001), etc. A framework covering these features could, in principle, cover any type of AH system.

---

[2] benefits can come from cross-field developments, e.g., connections to ontological research, open hypermedia, web standards, etc.

However, the main impetus for authoring research and development in AH comes from outside the field: from *distance learning* and *web-based educational systems*, but also from *e-commerce*, all driven by the pressure from the fastest growing hypermedia system, the *web*. The web is a huge information resource, not just for research laboratories, but for everybody. The 'lost-in-hyperspace' syndrome, which adaptive hypermedia set out to fight, is becoming more of an everyday reality. Personalization is urgently required, in the sense of *adaptability* and *adaptivity* to the end-user. The many successful (educational) hypermedia authoring tools (WebCT, Blackboard, etc.) don't offer enough personalization. Adaptive hypermedia has the answers, but not yet the tools. This fact is gradually being perceived by the AH community, which is now investing more effort now into the authoring issue (Brusilovsky, 2003).

When this research started, authoring research was almost non-existent within adaptive hypermedia. The AH taxonomies (Brusilovsky 2002) and frameworks (AHAM, Wu 2001; The Munich model, Koch & Wirsing 2001) that were developed were primarily aimed at describing and classifying extant AH systems. The authoring benefits of a common framework were merely a side-effect.

Recently, AH authoring has started developing along "two main axes" (Brusilovsky, 2003): *mark-up* (Interbook; AHA!; WHURLE, Brailsford et al, 2002) and *form (or GUI)-based AH authoring* (the newer AHA! 3.0, emerged from discussions on benefits of concept-based visualization, De Bra et al 2002; MetaLinks, Murray 2002, linking form-based concepts in a hyperspace; SIGUE, Carmona et al, 2002, an open corpus AH authoring approach, with external documents only; complex interface approaches, e.g., NetCoach, Weber et al, 2001 – "the only commercial AH authoring system", according to Brusilovsky 2003 - and ALE, Specht et al, 2002). The form-based approach is considered more beneficial for inexperienced authors (Brusilovsky, 2003).

Our implementation approach is, according to the above classification, form based. The theoretical framework, however, enables both types of approaches. Next we describe this theoretical framework.

THEORETICAL FRAMEWORK

This research has two major parts:

- *Theory:* gradual creation of a new framework for AH authoring, LAOS (and LAG).
- *Implementation:* integration of the framework's concepts and ideas into an AH authoring environment, MOT, the platform for analysis and testing.

Testing in this context has two directions:

- *Testing with designers and authors:* authoring environment testing, with all necessary criteria (expressivity, adaptive flexibility, collaboration issues, author satisfaction, etc.) and methodologies (questionnaires analysis, tracing author's work, etc.),
- *Testing with adaptive hypermedia users:* testing of the created AH environment with AH users.

LAOS

LAOS (Layered WWW AHS Authoring Model and their corresponding Algebraic Operators, Cristea & De Mooij, 2003a) is a general framework of data storage and manipulation model for authoring of adaptive hypermedia, composed of five components (Figure 1):

- *domain model* (DM),
- *goal and constraints model* (GM),
- *user model* (UM),
- *adaptation model* (AM) and
- *presentation model* (PM).

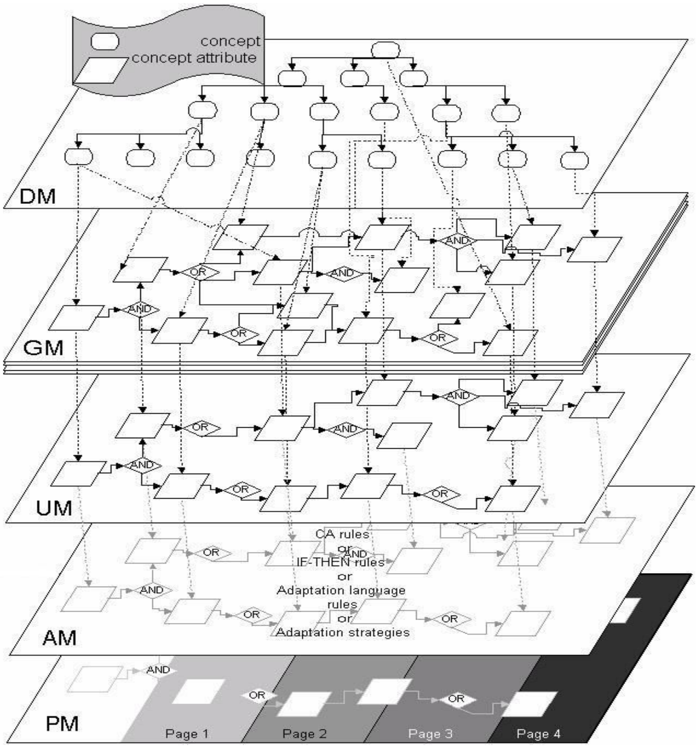

Figure 1. The five levels AHS authoring model.

LAOS builds on AHAM (Wu, 2001)[32], one of the first, well-known adaptive hypermedia architecture models. The major differences are:

- The clear separation of *information* (or knowledge) - and *presentation-goal related connectivity* (e.g., pedagogical methodology in educational hypermedia). This is done to facilitate information reuse, by separating information chunks from specific context.

- The above separation generated two different models, instead of one: a domain model (DM) and a goal and constraints model (GM). This separation can be understood easily if we use the encyclopaedia metaphor: the DM represents the encyclopaedia(s) on which the presentation (e.g., with PowerPoint^{TM} and represented by the GM) is built. From one encyclopaedia (or DM) we can construct several presentations (here, GMs), depending on our *goal*. These presentations don't contain everything in the encyclopaedia, just some (*constrained*) part of it, which we consider relevant. Moreover, a presentation can contain information from several encyclopaedias. This separation therefore gives a high degree of flexibility, as shown later.
- Another important difference is given by the notion of 'concept' that we use in the domain model. Our concepts have different representations given by their attributes, which can also represent resources (as in RDF[28]). The only restriction is that concepts should have a *semantic unity* (unlike in AHAM).
- The adaptation engine has to actually implement not only *selectors*, but also *constructors* (Wu, 2001), as presentations can contain any type of combination of (ordered and weighted) concept attributes (which is different to AHAM).

Next we look at the LAOS composing models in more detail.

Domain model (DM)

The domain model is composed of concept maps, containing linked concepts. These concepts are further comprised of attributes. This model represents the learning resources and their characteristics.

Goal and constraints model (GM)

This model filters, regroups and restructures the previous (DM) model, with respect to an instructional goal. It allows ordering and AND-OR relations between these attributes, as well as weights for the OR relations. The actual interpretation of this structure is done by the adaptation model.

User model (UM)

UM and AM have been described relatively well by AHAM. Another way of representing the UM (Cristea & Kinshuk, 2003) is to view it also as a concept map. In this way, relations between the variables within the user model can be explicitly expressed as relations in the UM, and do not have to be "hidden" within adaptive rules.

Adaptive model (AM): Layered Adaptation Granulation (LAG)

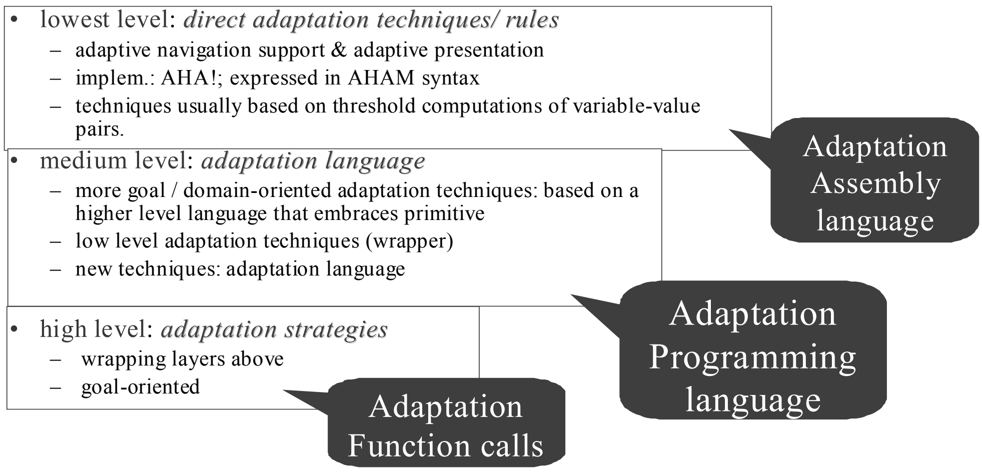

Figure 2. LAG: The three layers of adaptation.

The AH adaptation model traditionally consists of a set of IF-THEN rules that are triggered when some event occurs (e.g., accessing of a page or a concept). However, this type of structure has proven to be quite cumbersome for authoring. To overcome the limitations of the inexperienced author, but also to allow enough flexibility for the advanced author, we have introduced (Cristea & Calvi, 2002; 2003) a new three-layer adaptation model (by adding, over the typical low level *assembly*-like *adaptive language*, a medium level *programming adaptive language* and *adaptive strategies* language) called LAG (Figure 2).

This model allows different difficulty levels for different authors, being a "frame-based model" (Brusilovsky, 2003) with added semantics. Moreover, as the higher levels of authoring imply grouping of low level adaptation constructs, reuse can occur. In this way it is possible to reuse not only the AH content, but also adaptive techniques, moving towards discovery of adaptive patterns.

In the following, these layers are described in more detail, by the type of rules they allow.

Direct Adaptive techniques – Adaptive Assembly Language

Low-level adaptive techniques are all techniques traditionally used in adaptive hypermedia applications (*content adaptation* – adaptive presentation: inserting/removing of fragments, altering fragments, stretchtext, sorting fragments, dimming fragments - and *link adaptive* techniques: adaptive guidance – adaptive navigation support: direct guidance, link sorting, link hiding/ removal/ disabling, link annotation, link generation, map adaptation), summarized by Brusilovsky, 2002. They are usually determined by a mixture of fine-grained elements of the domain, user, adaptation, goal (GM) (Calvi & Cristea, 2002) and presentation model (PM).

Adaptive Language

This level is determined by grouping the elements of the previous layer into typical adaptation mechanisms and constructs (low-level rules into *higher-level adaptive rules*; operators or language constructs and variables into *adaptive language interface variables*). The result is a 'programming language' for adaptive strategies (Cristea & Calvi, 2003), called *adaptive language*. The instantiation of this language is the basis for elaborating adaptation patterns in the Minerva ADAPT project[1]. Constructs now in use are: *while-do*, *for-do*, *generalize* and *specialize*, but we are looking at extending this basic set. These constructs have the purpose of allowing (semantic) grouping and labelling of typical adaptive behaviour.

EXAMPLE: To illustrate the compression power with a simple example, and to show how they translate into if-then rules (so into regular adaptation rules), let's consider we want to show all 7 sub-concepts of the 'NN Introduction' concept in the goal and constraints map derived from the 'NN – intro' course in Figure 3.

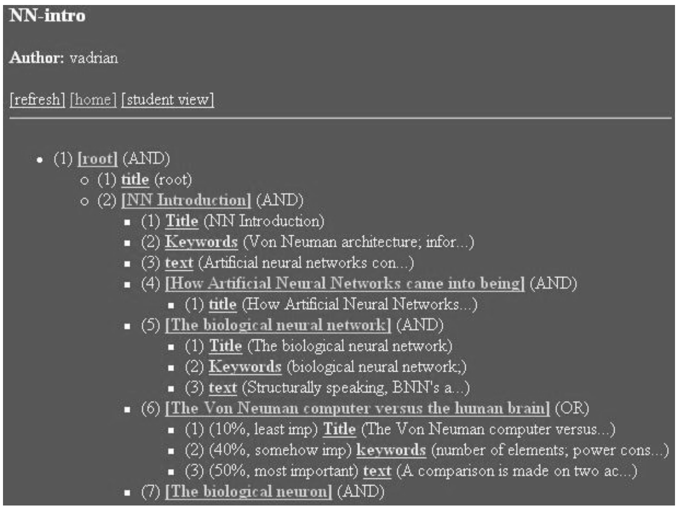

Figure 3. GM concepts.

In regular adaptation rule syntax, we would write seven lines:

IF NN Introduction.access == 'yes' THEN NN Introduction.Title.available = 'yes';

IF NN Introduction.access == 'yes' THEN NN Introduction.Keywords.available = 'yes';

IF NN Introduction.access == 'yes' THEN NN Introduction.text.available = 'yes';

IF NN Introduction.access == 'yes' THEN NN Introduction.How Neural ….available = 'yes';

IF NN Introduction.access == 'yes' THEN NN Introduction.The biological neural….available = 'yes';

IF NN Introduction.access == 'yes' THEN NN Introduction.The von Neuman….available = 'yes';

IF NN Introduction.access == 'yes' THEN NN Introduction.The biological neuron.available = 'yes';

In adaptation language constructs this would transform into:

> *IF NN Introduction.access == 'yes' THEN*
>
> *(NN Introduction.i = 1;*
>
> *FOR 7 DO (NN Introduction[i].available = 'yes'; UM.NN Introduction.i +=1))*

The latter form, as can be seen, is much shorter. In MOT, this looks like in Figure 4.

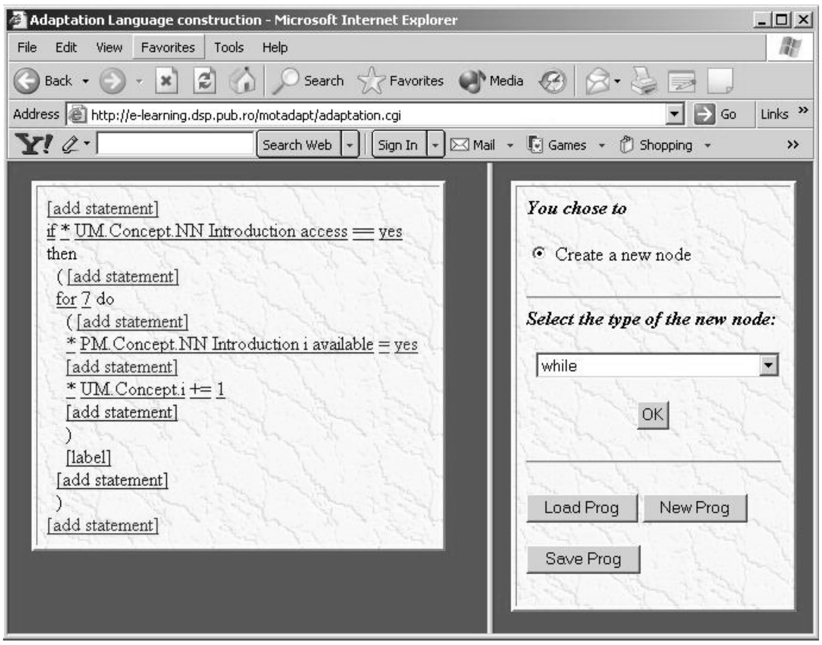

Figure 4. AM in MOT.

These language constructs represent the first level of *adaptive pattern reuse*.

Adaptive strategies

What we have done is to use the above building blocks (*adaptive language* or direct *adaptive techniques*), to build higher level programs. The four-line program above can itself be saved as a higher level compound, called *adaptive procedure* or *adaptive strategy*. Actually, more complex adaptive strategies than the one created in Figure 4 can be created, e.g., instructional strategies based on learning styles (Bajraktarevic et al, 2003; Cristea & De Bra, 2002), or tours, knowledge-related strategies, taste-oriented strategies (especially for commercial purposes), etc.

Presentation model (PM)

The PM takes into consideration the physical properties and environment of the presentation and provides the bridge to the actual code generation for different platforms (e.g., HTML, SMIL[30]). The presentation model in LAOS is similar to the user model, as it is a combination of an overlay model of both the domain model and the goal and constraints model, as well as of independent variables and their respective values. This structure allows attaching to each specific concept a certain representation type on screen, whilst at the same time retaining independent representation types, which depend on the current values of the user model.

Authoring steps in LAOS

The authoring steps[3] in LAOS [10]are shown in Table 1.

Table 1. Adaptive Authoring Steps in LAOS.

STEP 1	write concepts + hierarchy (in *DM*)
STEP 2	define concept attributes (main and extra attributes) (in *DM*)
STEP 3	fill concept attributes (write contents) (in *DM*)
STEP 4	perform selection and redesign hierarchy given the GM goal (design alternatives – AND, OR, weights, add extra narrative smoothening attributes, etc.) (in *GM*)
STEP 5	add UM related features (simplest way, tables, with attribute-value pairs for user-related entities; UM can be represented as a concept map) (in *UM*)
STEP 6	decide among adaptive strategies, write in adaptive language medium-

[3] Some steps can be done in parallel, or in different order, and even by different, collaborating authors.

	level adaptive rules (such as defined by LAG) or give the complete set of low level rules (such as condition-action, or IF-THEN rules) to decide interpretation of GM (or DM). (in *AM*)
STEP 7	define format (presentation means-related; define chapters) (in *PM*)
STEP 8	add (if necessary) adaptive features regarding presentation means (define variable page lengths, variables for figure display, formats, synchronizations points, etc.). (in *AM*)

With the LAOS structure, flexible (adaptive) presentation generation becomes possible, as follows. The actual presentation seen by the user can contain both elements of the goal and constraints model as well as domain model (e.g., clarification of a text-attribute from the GM, the system can leave the prescribed GM path to show other attributes of the respective DM parent concept, or other concepts related to this parent concept. This is similar to a pointer back to the encyclopaedia from which the presentation as generated). This increases flexibility and expressivity of the adaptive presentation creation process. This flexibility can be estimated, as we shall see, looking at automatic transformations in LAOS.

Automatic transformations in LAOS

The flexibility of an authoring system determines the capacity of performing 'automatic authoring'. Here we present some examples of automatic transformations between the models in LAOS [10]which can be performed directly by the authoring system.

In regular AH design and authoring, the equivalent of these processes are done by hand, which is time-consuming. We have identified some *patterns* based on the structure of LAOS that allow generalizing and automatically performing these transformations, leading to possible reuse; the reuse range can be estimated as a flexibility degree.

Definition 1. The *flexibility index, flex,* is defined as the combinatorial index enumerating all the possible results that can be generated by a specific (set of) automatic transformation(s).

From Domain Model to Domain Model (DM→DM)

Implicit DM information can become explicit, via some information retrieval technique.

- **DM→DM According to Concept Attribute Type**

The easiest way to enrich the domain model is by automatically finding new links between existing concepts[4]. In Cristea & De Mooij (2003b) we have developed formulas for the *relatedness calculations* between the different concepts, to find potential links between concepts sharing a common topic. These were computed at concept attribute level, using the name of the attribute as a type. Concepts C1, C2 can be linked in MOT if:

```
link(C1,C2,label,weight)=
=link(C1,C2,attribute-name,weight-formula)
```

[4] New links can be between concepts of *current content* (concept map: e.g., course), between current content and some *other content created by the same author*, or *created by a different author*.

Here, the link detection flexibility index, *flex(*,*)*, enumerates all possible links of unequivocal type[5]. If types can be mixed, we obtain the *mixflex(*,*)* index. Next, we show how we estimate these indexes.

Theorem 1. The *flexibility link index* for link generation based on concept attribute type for concept map ***C*** is can be estimated as:

$$flex(*,*) \geq \sum_{i=1}^{C} \sum_{j=i+1}^{C} A_{\min} = \frac{C(C-1)}{2} A_{\min} .$$

where: A_{min} - minimum number of attributes per concept; $C = card(\boldsymbol{C})$.

The *mixed link flexibility index* of concept map ***C*** is:

$$mixflex(*,*) \geq \frac{C(C-1)}{2} A_{\min}{}^{2} .$$

Proof:

The *mixed link flexibility index* of the links that can be generated between concepts C_1 (current concept) and C_2 is:

$$mixflex(1,2) = A_1 A_2 \geq A_{\min}{}^{2} ;$$

where: A_i the actual number of attributes of concept C_i.

For links that having unequivocal type, we obtain:

$$flex(1,2) = card(\mathcal{A}_{c1} \cap \mathcal{A}_{c2}) \geq card(\mathcal{A}_{\min}) = A_{\min} .$$

where: $\mathcal{A}_{ci}$ is the set of attributes of concept C_i

The *flexibility index* of linking concept C_1 with the rest of the concepts in ***C*** is:

$$flex(1,*) = \sum_{j=2}^{C} card(\mathcal{A}_{c1} \cap \mathcal{A}_{cj}) \geq (C-1) A_{\min} .$$

The *mixed flexibility index* for concept C_1 is:

$$mixflex(1,*) = A_1 \sum_{j=2}^{C} A_j \geq (C-1) A_{\min}{}^{2} .$$

---

[5] meaning that attributes determining the link are of same type in both concepts.

Therefore, the (concept attribute type based) link generation *flexibility index* for concept map ***C*** is:

$$flex(*,*) = \sum_{i=1}^{C} \sum_{j=i+1}^{C} card(\mathcal{A}_{ci} \cap \mathcal{A}_{cj}) \geq$$
$$\geq \sum_{i=1}^{C} \sum_{j=i+1}^{C} A_{\min} = \frac{C(C-1)}{2} A_{\min}$$
.

Similarly, the *mixed link flexibility index* of concept map ***C*** is:

$$mixflex(*,*) = \sum_{i=1}^{C} A_i \sum_{j=i+1}^{C} A_j \geq \frac{C(C-1)}{2} A_{\min}{}^2$$ .q.e.d.

EXAMPLE: concretely, in MOT, $\mathcal{A}_{\min}$ *={title, keywords, introduction, text, explanation, pattern, conclusion}, so* A_{min} *=7. In the concept map called 'Neural Networks I' (Figure 5) C=card(**C**)=145, so: flex(*,*)≥10440*7= 73080 and mixflex(*,*)≥10440*49= 511560.*

Please note that these are connections implied by only one concept map. MOT already allows inter-linking of concept maps, increasing this number.

Therefore, it results that many (annotated, semantic) links can be generated automatically, making the adaptive hypermedia process easier.

- **DM→DM According to Link Type**

It is also possible to create new links via a link-type check algorithm. The most common links (hierarchical) are already exploited in the adaptive language, via constructs such as *specialize* and *generalize* (Cristea & Calvi, 2003).

However, the most important contribution of link analysis would be comparing similar concepts[6] and finding missing attributes (or even sub-concepts) (*verification*).

EXAMPLE: The concept (Figure 5: concepts in left frame, attributes in right) called 'Discrete Neuron Perceptrons' from a Neural Networks course has an 'Example' attribute, whereas the concept 'Continuous Neuron Perceptrons' doesn't, although they are linked via their 'Title' attribute (weight: 67%). Here, the system could look for possible examples via other links to this concept, or just signal the missing content item.

For a concept map ***C***, the extra attribute *flexibility index* can be shown to be:

$$flex(*,*) = \sum_{i=1}^{C} \sum_{j=i+1}^{C} card(\mathcal{A}_{cj} - \mathcal{A}_{ci}) \geq 0 \,.$$

Please note that an extended version of the content search could look *outside* the space defined by the LAOS model.

- **DM→DM Combination of Concept Attribute - and Link Type**

We have seen above some computations for concept attribute type and link type only. Combining the above automatic transformations would require a tree-type check of the whole space. Table 3 gives the pseudo-code algorithm for this combination.

Table 3. Combination of concept attribute – and link type automatic transformations

```
FOR ALL Ci DO # Ci in set of concept maps
{ # Compute possible automatic links from current
concept Ci
```

[6] concepts sharing the father-concept, or at the same level of the hierarchy, or related with each other via some special link (of a given type), etc.

```
  FOR ALL Cj, j≠I DO
  { IF accepted(Ci,Cj,link,label,weight) THEN
      {
       link(Ci,Cj,label,weight); # Or just notify
user about it
      }
    # Compare attributes of Cj;
    IF EXISTS(link(Ci,Cj,*,*))AND
NOT(EXISTS(Ci.attr[k]))
       AND EXISTS(Cj.attr[k])
      {# Notify user about missing attr[k]
       Alert('missing attribute',Ci,attr[k]);
      }
    # Compare links of Cj with
Cj.link.type=label=l;
    IF EXISTS(link(Cj,Ck,label, weight))
       AND NOT(EXISTS(link(Ci,Ck,label,*))
      {# Notify user about missing link[k];
       Alert('missing link',Ci,link[k],label,
weight);
      }
   }
}
```

EXAMPLE: E.g., the concept called 'Discrete Neuron Perceptrons' from a Neural Networks course (Figure 5) has a link via the label='Title' attribute to the concept *called 'Continuous Neuron Perceptrons' (with some weight). The latter has a different link via the 'Title' label to the concept 'The artificial neuron'. The system can notice this link and prompt the user to consider connecting the first concept to the last (with, e.g., the same label, 'Title', and 33% weight).*

The computation of the flexibility index for the combined version is gained by combining the flexibility degree for the separate versions.

There are many more possible transformations from the domain model to the others, but in the following we limit ourselves to essential ones from the point of view of adaptivity and LAOS design originality.

From domain model to adaptation model (DM→AM)

Adaptive rules, by definition, are what make a hypermedia system adaptive; and should therefore be independent from the domain representation. However, a good domain representation can be the basis of smart adaptive behaviour. Moreover, domain model features can be interpreted to automatically generate adaptive rules. This can happen at the *direct adaptive technique* level, or at a higher level of *adaptive language* or *adaptive strategies* (LAG[8]). Therefore, instead of assigning a specific transformation for a given link type (or concept type), the same link (or concept) could be transformed differently, according to a different (e.g., pedagogically or financially rooted) adaptive strategy.

- **DM→AM According to Concept Attribute Type: attribute type related rules**

Attribute types can be used to create rules that determine which specific attributes are shown in some specific conditions. These conditions can be automatically deduced by the system (as in adaptivity) or triggered by the AHS user (adaptability).

EXAMPLE: A specific automatic adaptive low-level rule can determine showing the 'text' attribute of concept C1 only after the 'title' and 'introduction' were read:

IF (C1.title.access='yes' AND C1. introduction.access='yes')

THEN C1. text.available='yes';

Note that we have written the condition in this form for the purposes of simplification, but that attribute variables such as 'access' and 'available' are part of the user model[7]. For this to be a *generic automatic transformation rule*, for any concept C in the domain model, all concepts in the overlay user model reflecting the DM should have attributes 'access' and 'available'.

EXAMPLE: Instead of the rule above, the following generic rule can be used:

IF (C.title.access='yes' AND C. introduction.access='yes')

THEN C. text.available='yes';

The number of possible rules to generate is potentially infinite, because it is dependent on newly added UM variables (besides of *access* and *available*, referring to the state of some concepts with respect to the user, we can add general user variables such as *motivation*, *interest*, etc.). Even in the case with *s=2* variables, as above, and with the restriction that the 'access' variable can only be found on the left side of the rule, we obtain for the rule generation *flexibility degree*:

$$flex(1) = \left(\sum_{i=1}^{A_{min}} C(A_{min}, i)\right)^3 = \left(\sum_{i=1}^{A_{min}} \frac{A_{min}!}{(A_{min} - i)!i!}\right)^3.$$

[7] more precisely, part of the overlay part of the UM, as the UM can contain also other attributes such as user's prior knowledge, user's interest, etc., that are not an overlay model of the DM (or GM).

- ○ **DM→AM According to Link Type**

The links between concepts can be also interpreted in an adaptation model, so that, e.g, only specific links are 'fired' by the adaptation engine. In Cristea & Calvi (2003)[8] we have already used the inherent structure of the DM by defining the 'generalize' and 'specialize' adaptive language commands.

From goal and constraints model to adaptation model (GM→AM)

This type of transformation is more natural to the design of the LAOS structure, as the GM model contains a pre-selection of the material to present to the hypermedia user, according to some (pedagogical) *goal* and delimited by some (spatial, time, pedagogical, etc.) *constraints*.

The GM also pre-orders the DM information. This structure can already be interpreted in terms of the adaptation to be performed on it. For instance, the GM allows 'AND' relations between concepts, as well as 'OR' relations with some weights.

EXAMPLE: Expressing that all concepts in an 'AND' relation should be read:

> *IF ((C.name.access='yes' OR C.contents.access='yes') AND link(C,C2,'AND',*)) THEN { C2.name.accessible='yes'; C2.contents.accessible='yes';}*

Similarly, an 'OR' relationship can be interpreted as inhibiting reading of other related concepts [8]:

[8] In such a case, an 'OR' relationship acts actually as a 'XOR'.

IF ((C.name.access='yes' OR C.contents.access ='yes') AND

link(C,C2,'OR',))*

THEN { C2.name.accessible='no'; C2.contents.accessible='no';}

In such a way, various constructs can be automatically added to the generic adaptive rules, directly by interpreting the *goal and constraints model*.

PRACTICAL IMPLEMENTATION

This section describes the practical implementation and its testing.

MOT

MOT (My Online Teacher) instantiates the LAOS theoretical framework in the context of adaptive educational hypermedia. MOT is an AHS web-authoring environment, based on MyET (Cristea et al, 2000). MOT implements at present a first version of three of the LAOS models: domain model (DM), lesson model (GM) and adaptation model (AM).

MOT has been tested for authoring purposes in a classroom setting, using a first version which implemented the DM and GM; and then as a version with three models (DM, GM, AM). Further work is underway which will allow MOT to interface with AHA! (version 3.0) system (De Bra et al, 2002)[17].

MOT testing

At the time of the first tests, MOT could demonstrate the ideas of separating domain model and goal and constraints model, as well as some of the ideas on automatic authoring and transformations (specifically, automatic generation of links within the concept domain model and automatic generation of an instance of the goal and constraints model from an instance of the domain model). The adaptation model interface based on LAG (figure 4) was not yet present. Since then, more tests have been performed using the extended version of the system, and the results will soon be processed and reported.

MOT was designed and implemented to illustrate the ideas of the theoretical framework, and makes no great claims to a good user interface, or information display facility. Therefore, as described in the experimental settings section, our main testing goal was mainly to *validate the theoretical ideas* and their implementation. However, we also allowed as a secondary goal *feedback on user-system interaction*, as this was the main expertise of the students involved. This testing experiment belongs to the first category of tests mentioned in section 3: testing of design and authoring.

MOT is currently being further developed according to the flexible LAOS five-layer adaptation model for AH and adaptive web-material, towards reflecting all the separate models (domain, goal, user, adaptation and presentation models).

Testing goal

With the first tests of MOT within a classroom environment we wanted to get feedback on:

1. the extent to which our goals were realized with this system from an outsiders' perspective (LAOS representation, separation of domain and goal and constraints model, automatic authoring and automatic linking)

2. the usability of the system.

In particular, we wanted to find answers to questions as listed in Table 4.

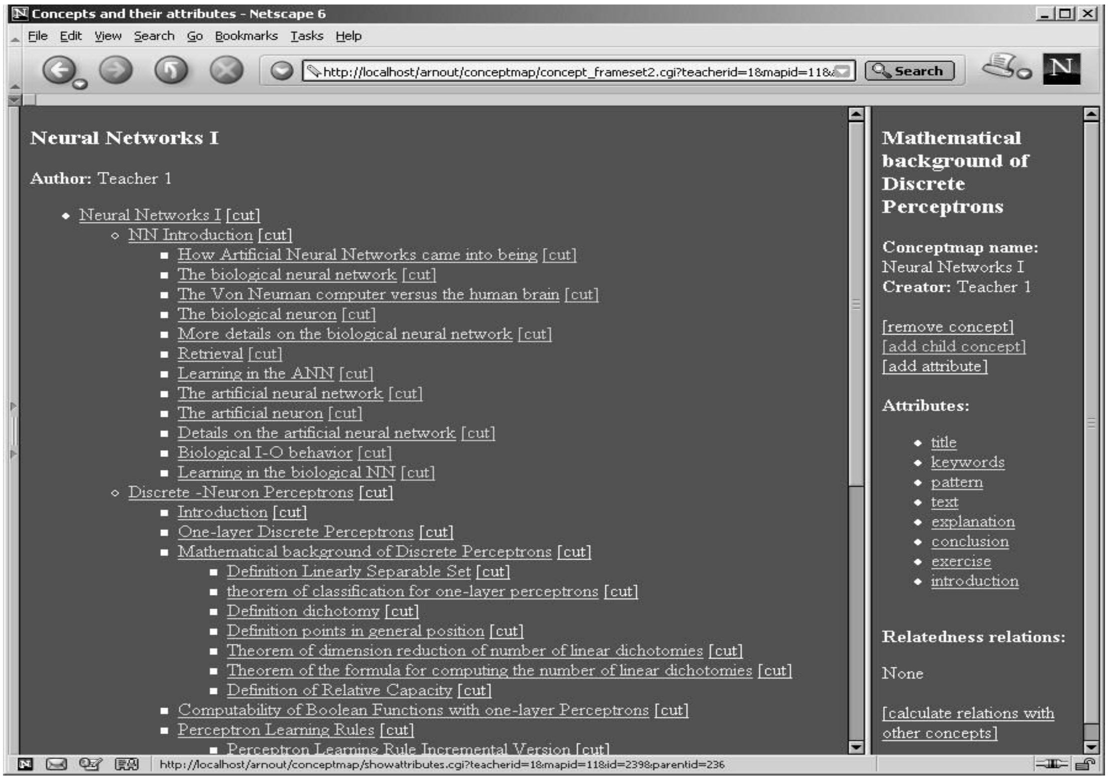

Figure 5. The authoring interface for the domain model.

Table 4. Goal point of view evaluation

Collaboration more authors collaborating at a	- what are the problems? Suggestions for solving them? How did you try and solve them? What are the good points? - Comparison - with collaboration (two or more working at one course together: experimental group) and without collaboration (one person only, with a smaller task: control group); the satisfaction degree should be measured, as well as the result

course;	evaluated.
Completeness looking at given goal (LAOS)	- what is the perceived percentage of completeness? What is the expressivity? What is the (perceived) connectivity degree? Should there be more connections, or less? What would these extra connections be? What should be deleted (e.g., is superfluous)?
Adaptivity	- How much flexibility is perceived?
Design range	- How much more can be achieved in this way as compared to the linear model?

The Class

The group was of about twenty students following a post-graduate two-year study of user interfaces and user-system interaction (USI). Students came from different backgrounds (from art to mathematics), and had different nationalities, genders and previous degrees. They were exposed to an intensive two-week course with the following teaching and testing procedure.

1. Background knowledge on adaptive systems, user modelling, with focus on AH
2. Exercises with building concept maps on paper
3. Theoretical framework of the MOT system (LAOS, LAG)
4. Exercises with writing LAOS-based rules on paper
5. Course evaluation: questionnaire (anonymous, individual)

6. Installing, experimenting and finally creating a presentation using MOT (domain and goal and constraints model) and on paper (adaptation, user and presentation model) in groups of 2-4.
7. Project evaluation: questionnaire (anonymous, individual)
8. MOT system evaluation: questionnaire (anonymous, individual) and free evaluation around the points listed in Table 4 (not anonymous, in groups).
9. Evaluation of assignments and staff grading

The questions of all the questionnaires were mapped on a Likert scale (0:min to 5:max). Issues that students wanted to point at were included in the free evaluations. The students were told from the beginning that their negative evaluation of the system would not affect their grades, but that the thoroughness and constructiveness of their answers would.

Evaluation

The students' numerical evaluation results for the MOT system were analysed for:

· the mean,

· standard deviation,

· correlation.

The analysis was made with online statistical software[25].

The student course creation results themselves were analysed, in order to reflect on the:

· time necessary to familiarize oneself with MOT,

· perceived flexibility of MOT,

· perceived freedom of expression in MOT,

· time necessary to create some courseware with MOT, etc.

Students were asked to make a MOT presentation on any theme they wanted within Adaptive Systems. Actually selected themes were ‘Concept maps for adaptive systems’, ‘Complex adaptive systems’, ‘Intelligent tutoring systems’, and ‘Introduction to artificial intelligence’. Their concept maps and respective lessons are on the online Unix version of MOT.

From the performed evaluations, we present here only those of interest to the research and development of the MOT (Figure 6, third questionnaire). However, these results are not completely independent from others; e.g., students missing theoretical background or unhappy with the course structure might have had difficulty in understanding the MOT system functionality. To ameliorate this problem, the first question concerns their understanding of how MOT works. The response is slightly positive (over 2,5), but must be taken as a prior probability influencing the precision of the further responses. Another question refers to their understanding of LAOS, the theoretical model behind MOT. The students’ response is similar (over 2,5). This understanding is again another prior probability influencing the certainty of the given responses.

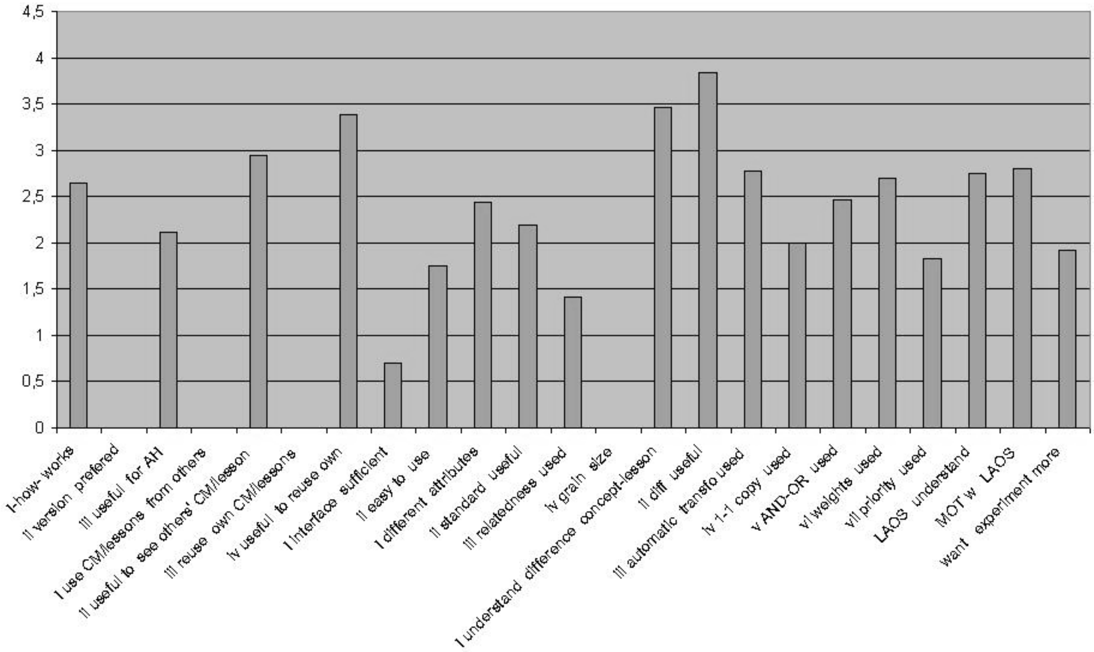

Figure 6. MOT evaluation.

With this premises we can look at the results.

Results

In figure 6, alpha-numeric results are not represented; e.g., the preferred MOT version was the online version. Students declared they used concept maps of others, as well as their lessons and link structures. On the issue of reuse of their own material, responses were various. The granularity of stored concepts also varied.

Among numeric results, it is interesting to note that students claimed they understood (Mean = 3.46; Standard Deviation = 1.20) the difference between domain model and lesson (i.e., goal and constraints) model, and that they find this difference very useful (Mean = 3.83; Standard Deviation = 0.389). This result is extremely important because this difference has been introduced as one of the originalities of the LAOS framework, intending to separate pure content related issues (e.g., concepts, their hierarchies and their relations) from presentation relation issues (such as how different aspects of concepts should be presented together, in what order, etc). Until recently the necessity of this separation has been a controversial issue in AH academic and research circles, but the students accepted it without any hesitation as being intuitive, and were able to work well with it.

Reuse of personal material (domain model, lessons) was considered useful, as well as the reuse of other teachers' makeshift material. One response involves pure reuse whereas the other response involves collaboration. Nevertheless, a comparison 'useful to see other's CM/lessons' with 'useful to reuse ones own' shows a 60,6% correlation between the two variables, pointing to the fact that these responses are related.

Another interesting and important result was that the students used automatic translations from the domain model to the lesson (goal and constraints) model. This is important because MOT was trying to illustrate that, although adaptive educational hypermedia authoring requires a complex process and the population of many models in order to achieve the best performance, simplifications and automatic transformations (as in section 3.2) are possible and can help the beginner author. Therefore, as the students were indeed beginner authors, their option for automatic translations confirms this hypothesis.

The special features of the goal and constraints model, such as weights (importance or difficulty of items to be presented), priority ordering of the items to be presented and grouping into AND or OR relations were used, but moderately. Therefore, the appropriateness of the components of this model has to be tested further.

A very low usage score was given to the relatedness relations between concepts, which can be computed automatically by the system. It seems that some students haven't even discovered this feature of MOT. The problem seems to be that the present implementation doesn't allow any direct application of these relations (except for informing the author about the existence of these relations).

The current domain model provides a set of predefined attributes for each concept. Students have used those moderately, but also have defined their own (which is allowed in MOT). The predefined attributes can be used to semantically label the concept and allow automatic machine processing (such as automatic concept linking). Different attributes were used with a mean of 2.19 and a standard deviation of 1.33 (actual responses varying from 0 – no other attributes used – to 4 – almost only other attributes used). The added attributes are more difficult to interpret, but nevertheless seem to be necessary for allowing full expression of the intentions of the author.

CONCLUSIONS AND FUTURE TRENDS

In this chapter we described a framework for authoring of adaptive hypermedia, LAOS, with its detailed adaptation model, LAG. This structure is in conformity with the requirements of W3C towards the third generation Web, called the Semantic Web[29].

Validation of this proposal was first based on existing background research and on an existing well-known framework, AHAM, which is not refined enough to allow flexible automatic authoring. The flexibility of LAOS was estimated theoretically, and some examples were given. Moreover, tests with real students were performed in MOT.

The research and development of AH authoring are new and will be developed in the future. We have pointed in this chapter to many possible future trends and developments within this area, which are briefly reviewed here. The main two axes of developments that we shall see in the future in this field are *theory* and *praxis*.
Theory developments will reflect improved organization of AH contents, as well as improved representations of the manipulation techniques. Especially, we shall see more collaborations between the semantic web community and the AH community towards better meta-data representations. For manipulations in AH, we shall see new, emerging patterns of adaptive behaviour, leading to new standards for the Web. This means, patterns that can be used to extend existent standards (such as learning object metadata LOM[23], simple sequencing standard or SCORM) or even generate new standards for adaptive hypermedia. These patterns will allow various automatic transformations and data generation for adaptive hypermedia, towards the AH that writes itself. Especially in the educational domain, high granularity patterns can emerge from embedding learning styles and cognitive styles research into AH, as in Bajrakrarevic, N et al (2002).
Moreover, it is interesting to note that products of this research match with findings from research on open (adaptive) hypermedia systems (see Henze & Nejdl, 2003). There, the necessity of creating patterns emerges not from authoring needs, but for needs of better interfaces between the objects of the open hyperspace. However, the solutions are similar and will eventually merge.
This is actually a time where benefits can come from cross-field developments, such as connections to ontological research, open hypermedia, web standards, etc.
From a praxis point of view, we will need research related to implementing these frameworks, but also, a lot of consideration will have to be given to visualization

techniques for the great bulk of information contained in AH systems. The author will have to benefit of fish-eye, bird-eye, and other views, slices through the information, etc.

Acknowledgement

This research is supported by the Minerva project ADAPT.

MOT can be downloaded at: http://wwwis.win.tue.nl/~acristea/HTML/USI/MOT/

MOT can be tried out at:

- http://e-learning.dsp.pub.ro/mot/

  (or http://wwwis.win.tue.nl/MOT03/TeachersSite-html/enter.html) and

- http://e-learning.dsp.pub.ro/motadapt/

LAOS discussion and MOT testing has been performed within two courses:

- April-May 2003, TU/e, Netherlands:

  http://wwwis.win.tue.nl/acristea/HTML/USI/index.html/

- January 2004, UPB, Romania:

  http://wwwis.win.tue.nl:/~acristea/HTML/PUB04/

Thanks to organizations involved in the process of *refining of adaptation patterns*: IFETS discussion group, (LTSC)(Kinshuk), CEDEFOP, University of Pittsburgh (Peter Brusilovsky).

References

[1] ADAPT project. "Adaptivity and adaptability in ODL based on ICT" (European Community Socrates Minerva 101144-CP-1-2002-NL-MINERVA-MPP). http://wwwis.win.tue.nl/~alex/HTML/Minerva/.

[2] Bajraktarevic, N., Hall W. and Fullick P. (2003) Incorporating learning styles in hypermedia environment: Empirical evaluation. *AH 2003 workshop session at HT'03 conference*, Nottingham, UK, available online at http://wwwis.win.tue.nl/ah2003/schedule-ht.html.

[3] Brailsford, T. J., Stewart, C. D., Zakaria, M. R., & Moore, A. (2002). Autonavigation, links, and narrative in an adaptive Web-based integrated learning environment. *Proceedings of World Wide Web 2002 Conference*, May 7-11, 2002. Honolulu, HI.

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

[5] Brusilovsky, P. (2002) Adaptive hypermedia, *User Modeling and User Adapted Interaction, Ten Year Anniversary Issue* 11 (1/2), 2002. 87-110.

[6] Brusilovsky, P., Schwarz, E., & Weber, G. (1996). A tool for developing adaptive electronic textbooks on WWW. In H. Maurer (Eds.), *Proceedings of WebNet'96*, October 15-19, 1996. San Francisco, CA, AACE. 64-69.

[7] Calvi, L. & A. Cristea, A (2002). Towards Generic Adaptive Systems: Analysis of a Case Study. *AH 2002, Adaptive Hypermedia and Adaptive Web-Based Systems*. LNCS 2347. Springer. 79-89.

[8] Carmona, C., Bueno, D., EduardoGuzman, & Conejo, R. (2002). SIGUE: Making Web Courses Adaptive. In P. De Bra, P. Brusilovsky, & R. Conejo (Eds.), *Proceedings of Second International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH'2002)*, May 29-31, 2002. Málaga, Spain, 376-379.

[9] Cristea, A.I., and Calvi, L. (2003) The three Layers of Adaptation Granularity. *Proceedings of UM'03*. LNAI 2702. Springer. 4-14.

[10] Cristea, A., De Mooij, A. (2003) LAOS: Layered WWW AHS Authoring Model and its corresponding Algebraic Operators. *Proceedings of WWW'03, Alternate Education track*, Budapest, Hungary 20-24 May 2003. ACM Press.

[11] Cristea, A.I., and Kinshuk. (2003) Considerations on LAOS, LAG and their Integration in MOT. *Proceedings of ED-MEDIA'03*, Honolulu, Hawaii, AACE.

[12] Cristea, A., De Mooij, A. (2003a) Adaptive Course Authoring: MOT, My Online Teacher. *Proceedings of ICT-2003*, "Telecommunications + Education" Workshop, Feb 23 - March 1, Tahiti Island in Papetee - French Polynesia, IEEE LTTF IASTED.

[13] Cristea, A., De Mooij, A. (2003b) Designer Adaptation in Adaptive Hypermedia. *Proceedings of ITCC'03*. Las Vegas, US 28-30 April. IEEE Computer Society.

[14] Cristea, A.I., and De Bra, P. (2002) Towards Adaptable and Adaptive ODL Environments. *Proceedings of AACE E-Learn'02*. Montreal, Canada, October. 232-239.

[15] Cristea, A.I. & L. Aroyo, L. (2002) Adaptive Authoring of Adaptive Educational Hypermedia. *Proceedings of AH 2002*, Adaptive Hypermedia and Adaptive Web-Based Systems, LNCS 2347. Springer, 122-132.

[16] Cristea, A., Okamoto, T. & P. Cristea, P. (2000) MyEnglishTeacher - An Evolutionary, Web-based, multi-Agent Environment for Academic English Teaching, *Proceedings of CEC 2000*, July 16-19, San Diego, USA, published on CD.

[17] De Bra, P., Aerts, A., Smits, D., Stash, N. (2002). AHA! Version 2.0, More Adaptation Flexibility for Authors. *Proceedings of ELearn'2002*, October 2002. AACE. 240-246.

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

[19] Grigoriadou, M., Papanikolaou, K., Kornilakis, H., & Magoulas, G. (2001). INSPIRE: An INtelligent System for Personalized Instruction in a Remote Environment. In P. D. Bra, P. Brusilovsky, & A. Kobsa (Eds.), *Proceedings of Third workshop on Adaptive Hypertext and Hypermedia*, July 14, 2001. Sonthofen, Germany, Technical University Eindhoven. 13-24.

[20] Henze, N. & Wolfgang Nejdl, W. (2003) Logically Characterizing Adaptive Educational Hypermedia Systems. *Proceedings of AH 2003 workshop session at WWW'03 conference*, Budapest, Hungary, http://wwwis.win.tue.nl/ah2003/proceedings/www-2/.

[21] Koch, N., Wirsing, M. (2001) Developing Adaptive Educational Hypermedia Systems: From Design Models to Authoring Models. In P. D. Bra, P. Brusilovsky, & A. Kobsa (Eds.), *Proceedings of Third workshop on Adaptive Hypertext and Hypermedia*, July 14, 2001.

[22] IMS simple sequencing protocol (2003) http://www.imsglobal.org/simplesequencing/index.cfm

[23] LOM: Draft Standard for Learning Object Metadata. http://ltsc.ieee.org/wg12/doc.html.

[24] Murray, T. (2002). MetaLinks: Authoring and affordances for conceptual and narrative flow in adaptive hyperbooks. *International Journal of Artificial Intelligence in Education*, *13* (1).

[25] Statistical tools online: http://home.clara.net/sisa/

[26] Specht, M., Kravcik, M., Klemke, R., Pesin, L., & Hüttenhain, R. (2002). Adaptive Learning Environment (ALE) for Teaching and Learning in WINDS. *Proceedings of Second International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH'2002)*. LNCS 2347. Springer. 572-581.

[27] SCORM: The sharable content object reference model .(2001) http://www.adlnet.org/Scorm/scorm.cfm.

[28] W3C, RDF (Resource Description Framework). http://www.w3.org/RDF/

[29] W3C, Semantic Web. http://www.w3.org/2001/sw/

[30] W3C, SMIL, Synchronized Multimedia Language. http://www.w3.org/AudioVideo/

[31] Weber, G., Kuhl, H.-C., & Weibelzahl, S. (2001). Developing adaptive internet based courses with the authoring system NetCoach. In P. D. Bra, P. Brusilovsky, & A. Kobsa (Eds.), *Proceedings of Third workshop on Adaptive Hypertext and Hypermedia*, July 14, 2001 (Sonthofen, Germany) 35-48. http://wwwis.win.tue.nl/ah2001/papers/GWeber-UM01.pdf.

[32] Wu, H. (2001) A Reference Architecture for Adaptive Hypermedia Applications (Doctoral dissertation, Eindhoven University of Technology, The Netherlands,) ISBN 90-386-0572-2.