Journal of Universal Computer Science, vol. 14, no. 22 (2008), 3627-3647
submitted: 7/1/07, accepted: 31/7/08, appeared: 28/12/08 © J.UCS

Learning Design in Adaptive Educational Hypermedia Systems

Adriana J. Berlanga
(Open University of the Netherlands, Heerlen, The Netherlands
adriana.berlanga@ou.nl)

Francisco J. García-Peñalvo
(University of Salamanca, Spain
fgarcia@usal.es)

Abstract: After more than ten years of research aiming at defining methods and techniques to deliver personalized instruction, Adaptive Educational Hypermedia Systems have not made the jump into real practice systems. Reasons for this include the complexity of their development, their use of exclusive methods for defining adaptivity and educational elements, and their lack of interoperation amongst courses and applications. A possible alternative to cope with these issues is using as a common notational method the IMS Learning Design specification. This paper attempts to bring AEHS and IMS LD closer to each other in order to define adaptivity behaviour. To this end, it outlines how IMS LD could be used to define personalization properties and adaptive techniques and, based on that, it proposes a component called Adaptive Learning Designs, and an authoring tool to create these components. Furthermore, the paper discusses the benefits and limitations of IMS LD to define adaptivity behaviour, and ends suggesting additional research lines.

Keywords: Learning Design, IMS LD, Adaptive Hypermedia Systems, Adaptive Rules, Authoring Tools
Categories: H.5.2, H.5.4, J.7

1 Introduction

Adaptive hypermedia systems consider users' characteristics to provide them with hypertext contents and links that they might find useful or adequate to their preference, needs, objectives or knowledge.

Certainly, this approach has many benefits for learning environments: providing learners with personalized support based on their characteristics might help them to comprehend the learning material. Educational applications are, indeed, the most popular area of adaptive hypermedia systems. Adaptive Educational Hypermedia Systems (AEHS) try to enhance learners understanding of the learning material providing them with paths and contents tailored to their characteristics and preferences. Nevertheless, the use of these systems in real learning environments is exceptional. Reasons for this include the lack of credible evidence of their benefits [Murray, 04], the pitfalls in their evaluation [Weibelzahl, 05], the absence of general propose tools and the proliferation of stand-alone systems [Stash, 07], the high cost of production and maintenance of these systems, their complex and time consuming authoring processes, and the lack of mechanisms for sharing and reusing components, such as adaptive strategies or learning resources, amongst courses and systems.

An additional remark is the tendency to design AEHS from the computer science perspective. This tendency, which is still apparent today, does not integrate instructional designers in the description of requirements and design of AEHS. In most of the cases the result has been content-orientated approaches that do not conceptualize learning as a process where learners interact not only with resources but also with learning activities, teachers and classmates.

An alternative to lessen the AEHS problems is using a common notational method such as the educational modelling language described in the IMS Learning Design specification [IMS-LD, 03]. This paper attempts to bring AEHS and IMS LD closer to each other in order to define adaptivity behaviour. To this end, we propose a component called Adaptive Learning Design (ALD), which objective is twofold. On one hand, the objective is to permit the definition of the characteristics of the learning strategy, like its learning objectives, prerequisites, learning activities, method of instruction and adaptive behaviour and, on the other, to support the reusability and exchangeability of the defined components amongst learning designs and tools compliant with IMS LD.

The definition of an ALD component is founded on the characterization, techniques, and elements that a number of AEHS take into consideration for performing adaptivity, as well as on the IMS LD attributes for modelling learning strategies with adaptive capabilities.

The rest of this paper is structured as follows: first, it mentions the concept of learning design and explains the most relevant characteristics of IMS LD for defining adaptivity behaviour. Subsequently, the paper explains how IMS LD can be used to model features for performing adaptivity as well as adaptive hypermedia techniques. Then, it describes how the ALD component integrates IMS LD and, finally, discusses the limitations of IMS LD to model adaptive behaviour, and exposes conclusions.

2 Learning design

Learning design is the application of learning design knowledge when developing courses or lessons. Learning design knowledge can be identified by means of instructional design theories, examples of best practices, and patterns and experiences [Koper, 05].

Instructional design theories attempt to define prescriptive design principles that depict the best way for supporting people to learn and develop cognitive, emotional and physical aspects [Reigeluth, 99]. These theories have been conceptualized from different points of view. Traditional models (e.g., [Gagné, 79], [Merrill, 94]), which are suitable for well structured domains, are considered as objectivist and prescriptive approaches, focused on content and learning outcomes. Conversely, constructivism approaches (e.g., [Spiro, 90], [Van Merriënboer, 97]), which are considered appropriate for ill-structured domains, seek to facilitate learners the construction of knowledge through learning activities.

Alternatively, novel approaches move away from prescribing instructional strategies, and propose models to solve problems in the description of learning designs by considering best practices and experiences from experts (http://www.pedagogicalpatterns.org).

From the computing perspective, the challenge is twofold. On one hand, it is necessary to design authoring tools that support the modelling of learning designs without prescribing any particular approach and, on the other, to reduce time and development costs, it is desirable that the resources and elements used in those learning designs could be reused and exchanged between lessons and platforms.

2.1 IMS Learning Design (IMS LD)

The IMS Consortium identifies and defines an e-learning set of specifications for interoperable learning technology. This framework includes specifications that aim at describing, for instance, learning resources [IMS-LOM, 04], content packages [IMS-CP, 04], learner information [IMS-LIP, 03], questions and tests [IMS-QTI, 05], and learning designs [IMS-LD, 03].

The objective of IMS LD is to provide a common notation that can be used to describe any learning process in a formal way. This notation, at the same time, aims at meeting requirements for completeness, pedagogical flexibility, personalization, formalization, reproducibility, interoperability, compatibility and reusability.

In short, IMS LD is a language for modelling units of learning (i.e., the smallest unit that satisfies one or more learning objectives). “A learning design, modelled using the language described in the IMS LD specification, captures who does what, when and using which materials and services in order to achieve particular learning objectives” [Tattersall, 04, pp. 3].

A learning design is mainly based on the learning and support activities learners and staff need to perform to reach a learning objective. Activities, which can be grouped into activity structures, must be included in a method that defines which activities should be performed by which roles. The definition of these methods follows a theatre-play-like structure: within a play, people (learners, staff) perform several roles (called role-parts) in certain order and sequence (called acts).

Figure 1 shows the hierarchical order of the IMS LD main elements (the asterisk represents that an element may occur more than once). They include learning objectives, prerequisites, components (properties, roles, activities, and environments), and a method of instruction that consists of a play (i.e., the way the method will be executed) and conditions.

Figure 2 shows a basic example of an IMS LD unit of learning notation. The example contains two learning activities (LA-Intro and LA-History) that are grouped in a sequence (AS-Intro), which specifies that both activities have to be completed sequentially (number-to-select="2" structure-type="sequence"). In the method section, this sequence is included in a play (Ply-EducTec) that has to be performed by the role learner (R-learner) in the act (Act-Intro).

```
Learning-design
  Learning-objectives
  Prerequisites*
  Components
    Properties*
    Role
      learner*
      staff*
    Activities
      learning-activity*
        environment-ref*
        activity-description
      support-activity*
      activity-structure {sequence |selection}
        environment-ref*
        activity-ref*
        activity-structure-ref*
    Environments
      environment*
        learning object*
        services*{mail-send | conference}
 Method
    Play*
      Act*
        Role-parts*
          role-ref
          activity-ref
    Conditions*
  Metadata
```

Figure 1: Order of the IMS LD main elements

```
<learning-design identifier="EducTec">
  <title>Educational Technology</title>
  <components>
    <roles>
      <learner identifier="R-Learner"/>
    </roles>
    <activities>
      <learning-activity identifier="LA-Intro"/>
      <learning-activity          identifier="LA-
History"/>
      <activity-structure identifier="AS-Intro"
number-to-select="2" structure-
type="sequence">
        <learning-activity-ref ref="LA-Intro"/>
        <learning-activity-ref          ref="LA-
History"/>
      </activity-structure>
    </activities>
  </components>
<method>
    <play identifier="Ply-EducTec">
      <title>First Lesson </title>
        <act identifier="Act-Intro">
          <title>Initial information</title>
          <role-part identifier="RP-First">
          <role-ref ref="R-Learner"/>
          <activity-structure-ref       ref="AS-
Intro"/>
…
```

Figure 2: Example of IMS LD notation

IMS LD has three levels of implementation and compliance that gather together. Level A contains vocabulary for supporting pedagogical diversity, Level B contains attributes and conditions that permit the definition of personalized sequences, and Level C adds notifications. Figure 3 shows the information model of IMS LD Level A and B. Grey marked classes belong to Level B.

In Level B the definition of personalization characteristics and more elaborated sequences and learning interactions is possible. This level includes elements as *properties* to store information about users, *global elements* to set and view the information of the properties, and *conditions* to manage and change the value of the properties.

Properties can be local or global. In the former case they can have the same value for all the users, contain information about each role in a unit of learning, or contain information about each person; whereas global properties can contain information about the user (e.g., portfolio information) or a single value for all users in all units of learning.

For personalizing the learning flow, properties are used in conjunction with conditions. Conditions have the following format:

IF [expression] THEN [action]

Thereby, an action will be performed if an expression is true or false. Actions can be defined, for instance, to show or hide an element (e.g., a learning activity, a play, etc.), change the value of a property, or send a notification. Expressions can include logical operations (e.g., and, or, greater than, less than, etc.), calculations (e.g., sum, multiply, etc.) or references to a particular element (e.g., role, act, learning activity, etc.).

Finally, there are attributes in elements of Level B that are specific for adapting learning activities, activity sequences, acts and plays, such as <when-property-value-is-set> and <when-condition-true>. Likewise the element <role-parts>, which describes the activities to be performed by a role in an act, could be used to group learners in stereotypes. In this way, every role-part covers a set of learning activities directed to a specific group of learners.

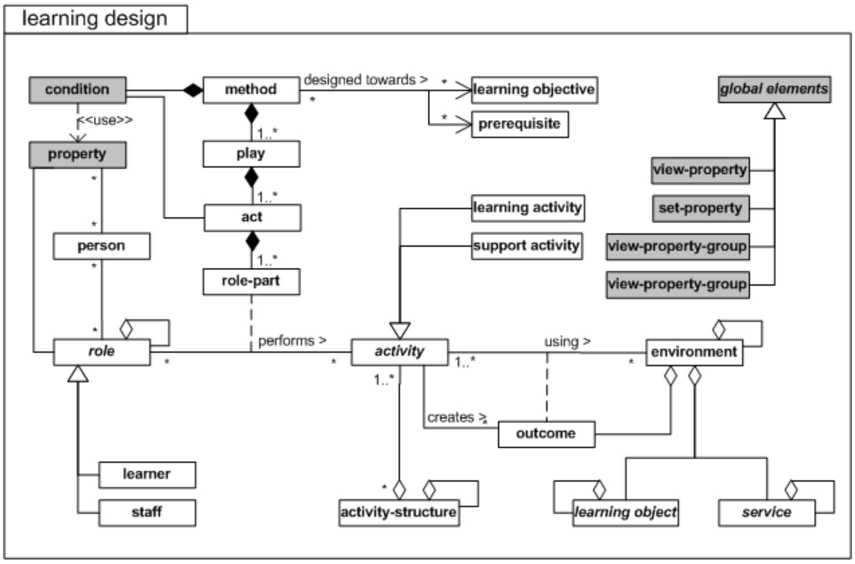

Figure 3: IMS LD information model Level B [IMS-LD, 03]

3 AEHS and IMS LD

Wu *et al.*, [Wu, 00] define two levels to control the adaptation in adaptive hypermedia systems: the author level and the system level. In the former, a person (expert, teacher, designer, etc.) defines and specifies the adaptation rules that will govern the system. In the latter, all the rules defined on the author level are executed by an adaptation engine. Given the elements and characteristics of IMS LD, we hold that this specification can be used in the author level to model adaptivity behaviour. This level includes the description of the features that are taken into account for performing adaptivity, such as learner preferences and knowledge, as well as their inclusion into rules for implementing adaptivity techniques [Brusilovsky, 96a].

To support this argument, this section first presents the features for performing adaptivity used in AEHS and explains how they can be modelled in IMS LD. Then, it describes how these features can be integrated in the definition of adaptive hypermedia techniques using IMS LD.

AEHS	Domain	Features for performing adaptivity	Adaptive techniques
InterBook [Brusilovsky,96b]	Domain independent Authoring adaptive contents	Prerequisites Learner knowledge Learning state	**Link annotation** **Direct guidance**
AHA! [De Bra, 01]	Domain independent Authoring AEHS	Attributes associated with concepts	**Inserting/Removing fragments** **Link annotation**
KBS-Hyperbook [Henze, 99]	Domain independent Authoring Educational Hypermedia Books	Prerequisites Learner knowledge Learner preferences	**Link annotation** **Direct guidance** **Routes/Project adaptation**
TANGOW [Carro, 99]	Domain independent Authoring of adaptive courses	Learner stereotype Learner preferences on learning strategy Learning styles (Felder & Silverman approach [Felder, 88])	**Inserting/Removing fragments** **Direct guidance**
INSPIRE [Papanikolaou, 03]	Computer architecture	Learner knowledge Learning styles (Honey & Mumford approach [Honey, 92])	**Direct guidance** **Inserting/Removing fragments** **Link annotation**
ALE [Specht, 02]	Learning Management System	Learner knowledge Learner preferences Learning styles (Felder & Silverman approach [Felder, 88])	**Link annotation**

Table 1: AEHS: features for performing adaptivity and adaptive techniques

ID	Category	Features	Examples
[LOB]	Learning Objectives		**Knowledge, skills, attitude, competence**
[LD]	Learner Demographics	Age	
		Language	**Spanish, English, Dutch**
[LC]	Learner Characteristics	Learner knowledge	
		Learning style	**Sensitive/ intuitive, visual/verbal, sequential/global [Felder, 88]**
[LP]	Learner Preferences	Level of detail	**Basic, medium, high**
		Learning style	**Activist, Pragmatist, Reflector, Theorist [Honey, 92]**
		Level of interactivity type	**Linear/interactive**
		Learning strategy	**Theory/practice; learning by example/learning by doing**
[MC]	Media Characteristics	Technical characteristics	**OS, bandwidth, hardware**
		Communication	**Synchronous/asynchronous**
		Media type	**Video, text, graphic**
		Interactivity type	**Linear/interactive**
[SC]	Setting Characteristics	Work type	**Individual/group**
		Work place	**Home/school**

Table 2: Classification of the features for performing adaptivity using IMS LD

3.1 Features for performing adaptivity using IMS LD

Features that adaptive hypermedia systems take into account for adaptation (in general, not only for AEHS) include [Brusilovsky, 96a] [Kobsa, 01]: user knowledge, user objectives, user experience in other fields of study (profession, experience, etc.), user preferences, demographic characteristics, information about the interaction of the user with the system, as well as technical information that affects the functionality of the system (e.g., software, hardware, bandwidth, etc.).

In AEHS, learner knowledge is, obviously, the most used feature for performing adaptivity. Table 1 summarizes some well-known examples of AEHS. It includes the domain of the AEHS, the features they consider for performing adaptivity and the adaptive techniques they use (see next subsection).

As explained before, information about users can be modelled using the `<properties>` element of IMS LD. Thus, this element is useful to include the elements AEHS normally use for performing adaptivity. The definition of this property is very flexible; it can represent any type of data. Consequently, the features for performing adaptivity presented in Table 1 can be modelled using the `<properties>` element of IMS LD.

The features these AEHS use for performing adaptivity can be seen as important features. They are, thus, learning conditions that can be placed into one of the following categories [Koper, 05]: learning objectives, learner characteristics, setting characteristics, and media characteristics. Table 2 shows an ID for each one of them, the options they might include, and an example. For instance, the category "learner characteristics" ([LC]) includes features such as learner knowledge and learning style of the learner –features that are used, for example, in AEHS such as ALE, INSPIRE or TANGOW–. Moreover, the table includes categories such as "learning demographics", "media characteristics", and "setting characteristics" that, as mentioned earlier, are reported by [Brusilovsky, 96a] [Kobsa, 01] as features that are considered for performing adaptivity.

3.2 Adaptive hypermedia techniques and IMS LD

Brusilovsky [Brusilovsky, 96a] [Brusilovsky, 01] distinguishes between methods and techniques for adaptive hypermedia. A method describes, from the conceptual point of view, a notion of adaptivity, while a technique is an implementation of that notion. It is possible, then, to implement the same method using different techniques, and vice versa, use different techniques to implement the same method.

Drawing from these concepts, Brusilovsky classifies adaptive hypermedia technologies in two areas: adaptive presentation and adaptive navigation support. Adaptive presentation techniques manipulate the content, whereas adaptive navigation support techniques manipulate the links. Figure 4 shows the taxonomy of these techniques.

Methods in adaptive presentation aim at providing users with additional, comparative or different explanations of the content; or at presenting the most significant text fragments by sorting them. Most of the research in this area is focused on adaptive text presentation and, particularly, on canned text adaptation [De Bra, 05]: text fragments are inserted, removed, altered, embedded (called stretchtext), sorted or dimmed in order to adapt the content that is shown to the user. Additionally, adaptive multimedia presentation and adaptation of modality techniques deal, respectively, with the selection of (canned or automatic) multimedia fragments and with the selection of different types of media (text, video, audio, etc.) to present the content.

Adaptive Presentation	**Adaptive Navigation Support**
▪ Adaptive multimedia presentation ▪ Adaptive text presentation o Natural language adaptation o Canned text adaptation - Inserting/removing fragments - Altering fragments - Stretchtext - Sorting fragments - Dimming fragments ▪ Adaptation of modality	▪ Direct guidance ▪ Adaptive link sorting ▪ Adaptive link hiding o Hiding o Disabling o Removal ▪ Adaptive link annotation ▪ Adaptive link generation ▪ Map adaptation

Figure 4: Adaptive Hypermedia Technologies Taxonomy [Brusilovsky, 01]

Methods in adaptive navigation support aim at helping users to find their way in the hyperspace by providing guidance and orientation support. Guidance is related to suggest the next step to take, whereas orientation support is related to present an overview of the link structure of the hyperspace. Therefore, techniques in this area manipulate the structure of the hypertext links to present relevant and appropriate information. Links, thus, can be sorted, hiding, annotated (by using visual clues), or automatically generated. Moreover, they can be manipulated to indicate the best next link to follow (direct guidance technique), or graphically organized to present an overall picture of the link structure (map adaptation).

As mentioned before, Table 1 shows which adaptive techniques use some well-known examples of AEHS. In adaptive presentation the most popular technique is inserting/removing fragments, whereas in adaptive navigation support the most popular ones are adaptive link annotation [De Bra, 05] and direct guidance techniques.

Adaptive technique	Description	IMS LD element	Features for performing adaptivity
Direct guidance	Next best IMS LD element to follow is presented	Play | Act | Role-Part | Activity Sequence	[LC] | [LP] | [LD] | [MC]
Show/Hiding links	Show/Hiding IMS LD element	Play | Act | Role-Part | Activity Sequence | Learning Activities	[LC] | [LP] | [LD] | [MC]
Link annotation	Annotate IMS LD element	Play | Act | Role-Part | Activity Sequence | Learning Activities	[LC] |[LP]
Inserting fragments/pages	Show IMS LD element	Play | Act | Role-Part	[LC] | [LP] | [LD]

Table 3: Adaptive techniques and IMS LD

Defining these techniques in IMS LD could be done in different levels of the learning design. Table 3 shows which IMS LD elements can be used to model adaptive techniques and which features these techniques can take into account for performing adaptivity. Note that the elements in the last column of Table 3 are equivalent to those defined in Table 2. As adaptive techniques manipulate pages and content, whereas IMS LD manipulates elements such as plays, acts, role-parts, activity sequences and learning activities, Table 3 also includes a description of the adaptive technique in terms of IMS LD.

For instance, an adaptive technique that shows or hides links can be defined at the level of a play, act, role-part, activity sequence and learning activities; and it can consider features such as prerequisites, learner characteristics, learner preferences, learning demographics and media characteristics.

The next section explains a component that helped us to test how the features for performing adaptivity and the adaptive techniques can be defined in IMS LD. To this end, we characterized and developed an authoring tool for creating the so-called Adaptive Learning Designs.

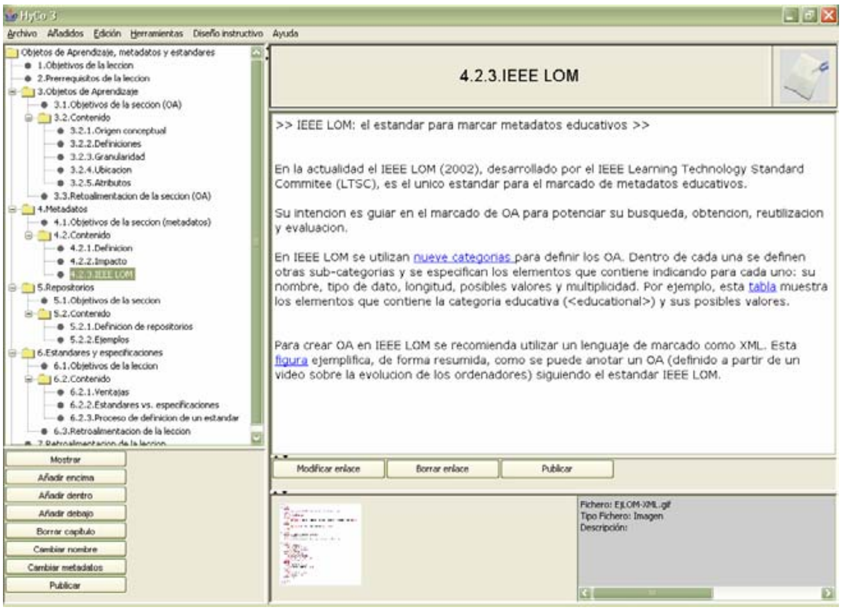

Figure 5: HyCo - main screen

4 Adaptive Learning Designs (ALD)

An ALD is characterized as a learning design that considers learner characteristics (i.e., knowledge, learning styles, etc.) to deliver a personalized learning flow [Berlanga, 05a]. In order to permit their reutilization, ALD are semantically structured according to IMS LD. Therefore, they aim at combining the personalization and reusability characteristics of IMS LD.

Consequently, an ALD is characterized by elements such as learning objectives, prerequisites, components (that include roles, learning activities, activity sequences and personalization properties), and a method of instruction (that includes adaptive rules, plays and acts). The definition of these elements, and their sub-elements, follows a *Lego* metaphor [Berlanga, 05b] where each one of them is defined and stored as a separate object and, as a result, elements, sub-elements and ALD (as a whole component) can be reused in different learning contexts, lessons and courses, or amongst different AEHS, applications and tools. That is to say, a learning activity could be incorporated in different activity sequences, and defining a new method of instruction does not imply to define again learning activities or objectives that have been created for other methods already included in existing ALD.

Based on this notion of a Lego metaphor, an ALD authoring tool was developed. This tool, called HyCo-LD, was built up enhancing the functionality of the Hypermedia Composer (HyCo) [García, 06]. HyCo is an authoring tool for hypermedia books structured as chapters and subchapters. These chapters and subchapters, as well as the resources they include can be linked to ALD elements such as learning activities, learning objectives, prerequisites, or feedbacks. Therefore, using the same tool, users can create learning resources and integrate them in ALD. HyCo includes, also, a metadata editor compliant with IMS LOM. This allows users to export resources and chapters as learning objects.

Figure 5 shows the main screen of HyCo. In the left section users can visualise and manage the chapters and subchapters of the book. In the editing section, which is on the central area, users can type the content of the book, and include multimedia resources and bibliographical references. From the main screen users can select the menu instructional design ("*diseño instructivo*" in the interface), which shows a submenu (see Figure 6) from where users can define or edit:

- Learning objectives ("*objetivos de aprendizaje*")
- Prerequisites ("*prerrequisitos*")
- Components ("*componentes*"), which include the definition of roles, learning activities, activity sequences, and personalization properties
- Methods ("*flujo de aprendizaje*"), which include the definition of adaptive rules, acts and plays
- Adaptive Learning Designs ("*diseño de aprendizaje*")
- Content packages ("*empaquetado*") compliant with IMS CP

Moreover, using the option "*player*" of this sub-menu, users can execute an ALD using the Coppercore player [Martens, 04] that is integrated into HyCo-LD as a third-party IMS LD player.

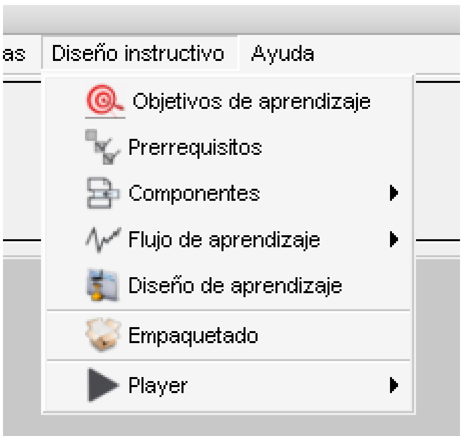

Figure 6: HyCo-LD - instructional design menu

Consistent with the Lego metaphor, in HyCo-LD every element is defined independently from each other and stored in a repository. This facilitates their handling and incorporation into other elements. The definition of each element is presented using a tab structure; it groups the sets of attributes needed to describe the element. As an example of the HyCo-LD interface, Figure 7 shows the interface for defining learning activities.

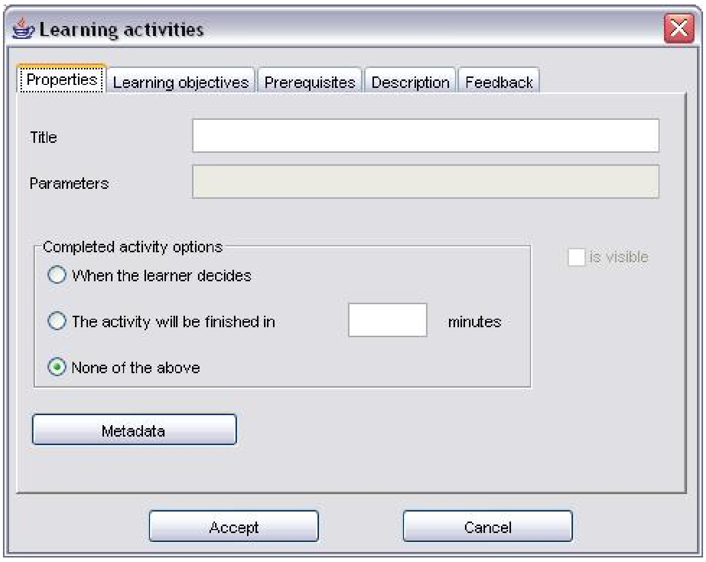

Figure 7: HyCo-LD - definition of learning activities

For defining adaptive behaviour authors should create personalization properties that, afterwards, can be included in adaptive rules. Personalization properties, which definition is based on the set of features for performing adaptivity explained before (see Table 3), contain users information, whereas adaptive rules are prescriptions defined by authors that will be taken into account to adjust the learning method. There are two ways of defining these rules. The first one, which is based on the description of adaptive hypermedia techniques explained before, is intended for non-expert users of the specification. The second one is intended for users with deep knowledge of IMS LD.

4.1 Creating personalization properties

As in IMS LD, in HyCo-LD users can create different types of properties: local, personal or role. Local properties have the same value for all users, personal properties have different values for every user, and role properties have the same value for every role.

As it has been said before, in IMS LD the definition of properties is very flexible. To define them authors have to indicate the title, data type (integer, character, boolean, etc.) and the initial value of the property. It is also possible to include restrictions, such as the minimum and maximal permitted values. Figure 8 shows the interface for defining role properties.

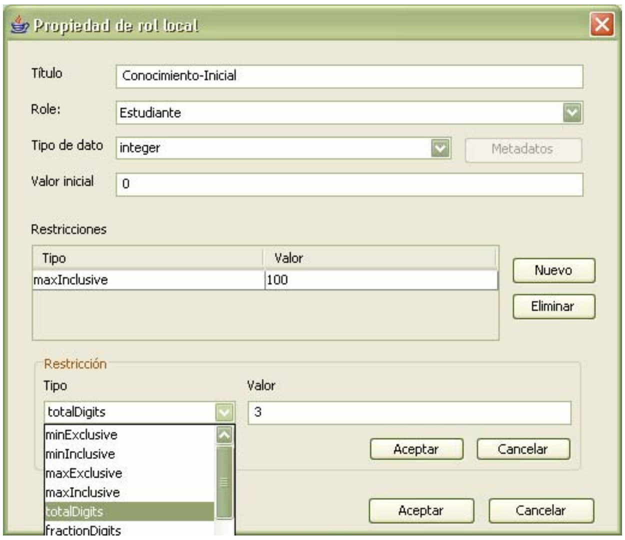

Figure 8: HyCo-LD - definition of roles

4.2 Authoring adaptive hypermedia techniques

HyCo-LD has a wizard for supporting users in the definition of basic adaptive techniques. These techniques are managed as separate objects and can be included into different learning methods. The wizard follows the definition of adaptive techniques presented in Table 3, and the features for performing adaptivity presented in Table 2.

Figure 9 shows the interface for defining adaptive hypermedia techniques. In the first tab "Attributes, type and level" ("*Atributos*, *tipo y nivel*" in the interface) the type of the technique (e.g., direct guidance) and its name and level (e.g., play, activity sequence or learning activity) should be indicated. Then, in the second tab, authors should select the feature that has to be taken into account for performing the adaptivity behaviour; the available options are those defined in Table 2. Therefore, when the author selects a category (e.g., learner preferences; "*Basado en*" in the interface) then, the list box displays the features that the selected category contains (e.g., level of detail, learning style, etc.; "*opciones*" in the interface). Hereafter, the author should indicate the operation ("*operación*"), data type ("*datos*") and value ("*valor*") of the selected element, as well as the property ("*propiedad*") from which the value for performing the adaptive technique should be taken.

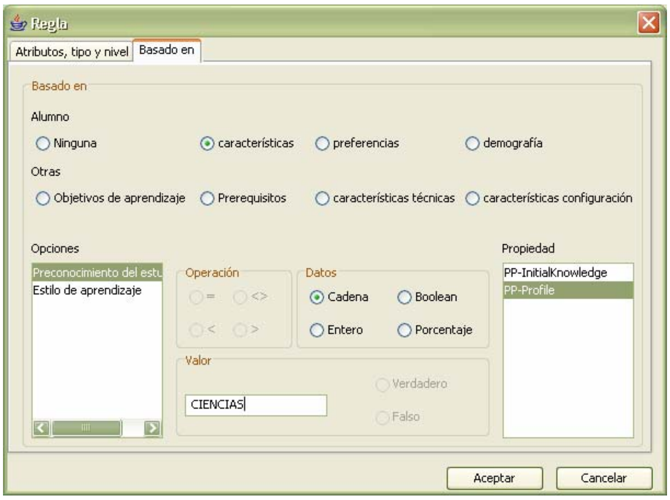

Figure 9: HyCo-LD - wizard to create adaptive hypermedia techniques

4.3 Authoring advanced adaptive rules

The definition of adaptive rules is guided by an expression-builder tool that gives authors more flexibility for deciding the characteristics and variables that should be defined. This tool uses a formalism [Berlanga, 06] based on the <conditions> element of IMS LD that allows authors to identify the elements of the learning design structure, as well as both the characteristics of the learner and of the learning activities authors want to include in the adaptive rule.

Figure 10 shows the interface of the expression-builder tool. It contains different boxes that include the operations, properties and elements that can be selected to create an adaptive rule: operators ("*operador*" in the interface), properties ("*property*"), roles ("*roles*"), learning activities ("*actividades de aprendizaje*"), activity structures ("*estructuras de actividades*") and plays ("*ejecuciones*"). From these boxes authors select the element they want to include in the adaptive rule. The tool guides them showing only the operator, property or element that can be chosen in each part of the construction of the rule.

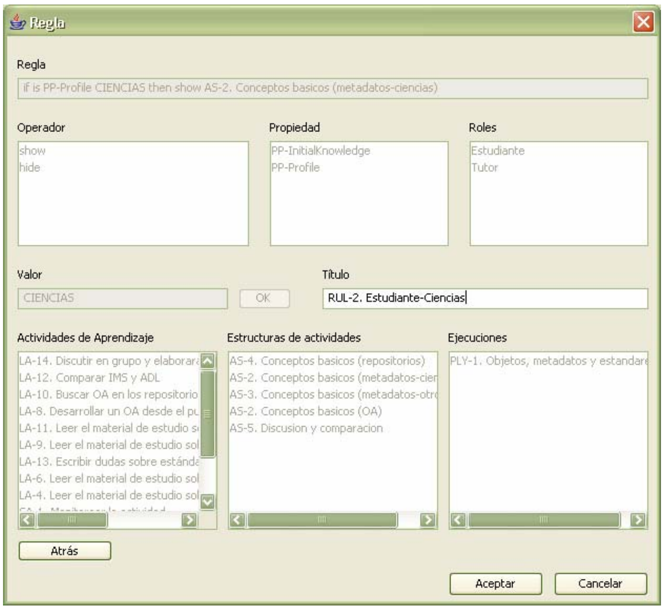

Figure 10: HyCo-LD - expression-builder tool for adaptive rules

4.4 HyCo-LD evaluation, reusability and interoperability

An evaluation of the adaptivity behaviour of the ALD component has been conducted [Berlanga, 06]. The aim of this evaluation was to test the adaptivity behaviour of a particular ALD, as well as the learners' perception about it. We use as an experimental setting a postgraduate course of the University of Salamanca, which is directed to learners from a wide range of backgrounds. The HyCo-LD wizard was used to define a direct guidance technique that, considering the prerequisites and the initial knowledge of the learner, showed different learning activities. Furthermore, the expression builder tool was used to create an adaptive rule that showed a complementary learning activity if the learner wanted to go deeper on that topic. During the experiment, learners interacted with the ALD using the Coppercore player integrated in HyCo-LD.

The evaluation showed that most of the learners found the ALD adequate to their background. Nevertheless, most of them also expressed their desire to explore by themselves the existing material and learning activities and, then, decide the learning flows they would like to follow.

Regarding the reusability of the components created in HyCo-LD, several use cases have been conducted. Particularly, using the learning activities created for the experiment mentioned above, we created a new ALD intended for advanced learners with computer background. Almost the same learning activities, activity structures and acts where used, but a new adaptation rule and a more technical learning activity (i.e., programming in XML) were included. This verification showed existing learning activities could be incorporated into new ALD.

Concerning the interoperability of ALD, since HyCo-LD integrates the IMS LD compliant player Coppercore for running ALD, it is safe to assume that ALD are interoperable components. Besides, other IMS LD authoring tools have been used to check the interoperability of ALD. Particularly, the well-known IMS LD editor Reload [Reload, 06] has been used to open and edit existing ALD. Although, the adaptivity rules defined in HyCo-LD cannot be defined in the same manner in Reload, they can be edited and modified.

5 Discussion

IMS LD has the potential to become a common notational method for developing adaptivity features [Towle, 05] [Burgos, 06]. Specifically, and as it has been argued above, the characteristics of this specification do allow the definition of adaptivity behaviour for AEHS. In the work we carried out, we analysed until what extent IMS LD can be used to model features for performing adaptivity and adaptive hypermedia techniques of AEHS.

The features for performing adaptivity can be easily modelled using the IMS LD element <properties>. As this element is defined in the specification following a meta-definition structure, any type of feature can be described. As explained before, features for performing adaptivity in AEHS include, for instance, learner knowledge, preferences or learning styles. Furthermore, properties in IMS LD can be viewed, set, modified and included in adaptive conditions or rules.

Combining IMS LD elements such as <properties> and <conditions> makes possible to model, until certain extent, adaptive hypermedia techniques. The element <conditions> has sub-elements such as <show> and <hide > that are ideal to define adaptive navigation support techniques like direct guidance, adaptive link hiding, and adaptive link annotation. However, IMS LD does not model the learning content; therefore the manipulation of the text is not covered and, as a consequence, adaptive presentation techniques such as stretchtext or dimming fragments cannot be modelled. However, adaptation techniques such as inserting/removing fragments or adaptation of modality can be represented in IMS LD if fragments or types of media are included in different learning activities that, accordingly to certain conditions, are showed or hidden.

Although, the use of IMS LD can be somehow restrictive, it may bring the following benefits:

- Incorporation of an existing annotation (i.e., ontology) to describe learning knowledge and pedagogical strategies into AEHS
- Assure the separation between pedagogical strategies, learning flows, adaptive logics, and content
- Feasible reutilization and interoperation of resources, learning elements and learning designs amongst courses and AEHS
- Quick AEHS prototyping and testing. For instance, a cycle for prototyping a particular adaptivity behaviour might consist of: creating a prototypical ALD, testing it using IMS LD tools and verify if the results are as expected, making the necessary changes to the ALD, testing it again, etc. In this way

specific components that are needed can be tested and validated without having to make major changes in a course or AEHS.

There are, however, three issues that should be pointed out: the way a unit of learning has to be delivered, the definition of learning objectives, prerequisites and feedbacks, and the definition of roles.

For delivering a unit of learning, IMS LD requires to embed the learning design in a manifest compliant with IMS CP [IMS-CP, 04]. This makes it impossible to change the learning strategies once the learners are interacting with them. Thereby, all the alternatives the learner could follow have to be defined at design time. At run time, the learning flow is controlled only by the system. This makes the authoring process extremely time-consuming and takes the control of the learning flow away from the learner.

The second issue is the definition of learning objectives, prerequisites and feedbacks. Rather than on a learning flow level, in IMS LD the definition of these elements is done on a descriptive level. These elements participate in the learning flow only as information resources described by attached learning objects or URLs; they are not described as triggered elements that, once they have been completed, viewed or assessed, they could activate additional actions like presenting additional learning activities, activating new actions, or marking an activity as completed. Nonetheless, it is possible to use an artificial alternative and create learning sequences that represent prerequisites. For instance, if the learning activity A is a prerequisite of the learning activity B, an activity structure that contains A and B and sequences them (A then B), has to be defined. However, this is not the optimal solution because the prerequisite of the learning activity B will not be semantically stated and, then, it cannot be included as an element to define adaptive rules nor retrieved automatically.

The last issue is the definition of roles. In IMS LD the definition of roles is not based on properties. This impedes personalizing the learning flow for roles that have certain characteristics (e.g., knowledge, preferences) or dynamically change the role of the learner.

Finally, it should be stressed that IMS LD is a machine-readable educational modelling language, which is not intended to be used as a "programming language" for creating units of learning nor adaptive hypermedia techniques. This specification should play a "back-stage" role. Definitively, evaluation results of the designers' perception about the creation of ALD and the tools proposed in this paper, as well as current research on visual authoring tools for non-expert users [Tattersall, 07] might bring some insights regarding this matter.

6 Conclusions

It is clear that pushing forward the benefits of AEHS for a wide range of applications and systems requires using a common notational method. As it has been explained in this paper, IMS LD can be used for this purpose.

We hold, however, that from the pedagogical point of view, the way of delivering learning designs in IMS LD and the philosophy of the AEHS need to go one step further in order to give learners freedom for building their own learning flows and, as a result, reduce the pre-design workload of tutors and teachers.

Hitherto, the philosophy behind (most of the) AEHS is providing each learner with the content he or she may need according to his/her characteristics. This personalization approach is focused on suggesting learners the next best resource or learning activity to follow, instead of let them freely explore the learning material, learn from other peers, and decide which learning flow to follow.

An alternative approach is showing all the existing learning activities and following learner's footsteps. This tracing can include, for instance, tracking the learner's behaviour with the system, his/her interaction with the ALD or unit of learning, his/her successful completed learning activities, or the learning flows peers with the same characteristics have followed. Based on this information, a new emerging ALD can be suggested. As this new ALD will not be pre-designed, it should be created at run time. An engine, thus, should look for the most appropriate learning activities to automatically generate the new ALD or IMS LD package. The Lego metaphor explained earlier can be useful to select, at run time and form different servers, which learning activities should be considered.

Tracking learner behaviour and his/her interaction with the learning activities can also facilitate the recommendation of the relevant learning activities to peers that have the same characteristics, as well as provide learners with information about the reasons (e.g., characteristics of the learner, characteristics of the learning activity, highly rated by peers, etc.) of the recommendation provided.

Furthermore, this tracking might facilitate, as well, the evaluation and analysis of the most followed and efficient paths and, based on that, generate adaptive rules that can be used, later on, for personalizing recommendations and provide navigation support [Hummel, 07]. The features for performing adaptivity and the definition of adaptive rules explained in this paper can be used to generate the IMS LD structure of these rules and, then, store them as separate objects that can be combined following different conditions to provide further recommendations.

We believe that tracing learner's behaviour, making recommendation based on peers interaction, providing learners with information about the reasons of the recommendation, and evaluating learning flows, is a bottom-up approach for defining emerging ALD or learning designs that will foster learner control and diminish author's work load. This is a challenging approach that, certainly, deserves further research.

Acknowledgements

Authors would like to express thanks to the blind reviewers for their valuable comments and remarks on the previous versions of this paper. Special thanks to Jorge Carabias for the development of HyCo-LD and, in general, to the other members of the research GRoup of InterAction and eLearning (GRIAL) of the University of Salamanca. Adriana Berlanga thanks the support of the Mexican Council of Science and Technology (CONACyT). This work is partially supported by the Spanish Ministry of Education and Science (ref. TSI2005-00960).

References

[Berlanga, 05a] Berlanga, A., García, F.: “Learning technology specifications: semantic objects for adaptive learning environments”; International Journal of Learning Technology, 1, 4 (2005), 58-472.

[Berlanga, 05b] Berlanga, A.J., García, F.: “IMS LD reusable elements for adaptive learning designs”; Journal of Interactive Media in Education, 11 (2005). Retrieved August 10, 2007 from http://jime.open.ac.uk/2005/11.

[Berlanga, 06] Berlanga, A.J., García, F., Carabias, J.: “Authoring Adaptive Learning Designs Using IMS LD”; Proceedings of 4th International Conference, AH 2006, Lecture Notes in Computer Science. Springer, Berlin (2006), 31-40.

[Brusilovsky, 96a] Brusilovsky, P.: “Methods and techniques of adaptive hypermedia”; User Modeling and User Adapted Interaction, 6, 2-3 (1996), 87-129.

[Brusilovsky, 96b] Brusilovsky, P., Weber, G., Schwarz, E.: “A Tool for Developing Adaptive Electronic Textbooks on WWW”; Proc. WebNet'96, AACE Publishing, Vancouver (1996), 64-69.

[Brusilovsky, 01] Brusilovsky, P.: “Adaptive Hypermedia”; User Modeling and User Adapted Interaction, 11, 1-2 (2001), 87-110.

[Burgos, 06] Burgos, D., Tattersall, C., Koper, R.: “Representing adaptive eLearning strategies in IMS Learning Design”. Learning Networks for Lifelong Competence Development TENCompetence Workshop, March, 30-31, 2006, Sofia, Bulgaria. Available on line: http://hdl.handle.net/1820/601

[Carro, 99] Carro, R. M., Pulido, E., Rodríguez, P.: “TANGOW: Task-based Adaptive learNer Guidance on the WWW”; Proc. 2nd Workshop on Adaptive Systems and User Modeling on the Web at the 8th International World Wide Web Conference (1999), 49-57.

[De Bra, 01] De Bra, P., Ruiter, J. P.: “AHA! Adaptive Hypermedia for All”. Proc. WebNet 2001, AACE Publishing, Vancouver (2001), 262-268.

[De Bra, 05] De Bra, P., Stash, N., Smits, D.: “Creating Adaptive Web-Based Applications”; Tutorial at the 10th International Conference on User Modeling, Edinburgh, (2005). Retrieved July 24, 2007, http://wwwis.win.tue.nl/~debra/um2005/tutorial.pdf

[Felder, 88] Felder, R. M., Silverman, L. K.: “Learning and Teaching Styles in Engineering Education”; Engineering Education, 78, 7 (1988), 674-681.

[Gagné, 79] Gagné, R., Briggs, L.: “Principles of Instructional Design”; Holt, Rinehart & Winston Inc., New York (1979).

[García, 06] García, F. J., Berlanga, A. J., Carabias, J., Gil, A. B., García, J.: “HyCo Authoring Features”; In R. Navarro-Prieto & J. Lorés-Vidal (Eds.), HCI Related Papers of Interacción 2004, Springer, The Netherlands (2006), 71-79.

[Henze, 99] Henze, N., Nejdl, W.: “Adaptivity in the KBS Hyperbook System”; Proc. 2nd Workshop on Adaptive Systems and User Modeling on the Web at the 8th International World Wide Web Conference (1999), 67-74.

[Honey, 92] Honey, P., Mumford, A.: “The manual of Learning Styles”; P. Honey Publications, Maidenhead (1992).

[Hummel, 07] Hummel, H.G.K., Van den Berg, B., Berlanga, A. J., Drachsler, H., Janssen, J., Nadolski, R.J., & Koper, E. J. R.: "Combining Social- and Information-based Approaches for Personalised Recommendation on Sequencing Learning Activities"; Int. J. Learning Technology, 3, 2 (2007), 152–168.

[IMS-CP, 04] IMS: "IMS Content Packaging v1.1.4 final specification"; (2004). Retrieved June 14, 2007, from http://www.imsglobal.org/content/packaging.

[IMS-LD, 03] IMS: "Learning Design specification v1"; (2003). Retrieved June 14, 2007, from http://www.imsglobal.org/learningdesign.

[IMS-LIP, 03] IMS: "The Accessibility for Learner Information Package v1 Final Specification"; (2003). Retrieved June 14, 2007, from http://www.imsglobal.org/profiles.

[IMS-LOM, 04] IMS: "IMS Meta-Data v1.3 Public Draft specification"; (2004). Retrieved June 14, 2007, from http://www.imsglobal.org/metadata.

[IMS-QTI, 05] IMS: "IMS Question and Test Interoperability v2.0 Final Specification"; (2005). Retrieved June 14, 2007, from http://www.imsglobal.org/question.

[Kobsa, 01] Kobsa, A., Koenemann, J., Pohl, W.: "Personalized Hypermedia Presentation Techniques for Improving Online Customer Relationships"; The Knowledge Engineering Review, 16, 2 (2001), 111-155.

[Koper, 05] Koper, R.: "An Introduction to Learning Design"; Learning Design. A Handbook on Modelling and Delivering Networked Education and Training, Springer, The Netherlands (2005), 3-20.

[Martens, 04] Martens, H., Vogten, H., Rosmalen, P. v., & Koper, E. J. R.: *CopperCore*. Retrieved June 14, 2007, from SourceForge: http://coppercore.org

[Merrill, 94] Merrill, D.: "Instructional Design Theory". Educational Technology Publications, Englewood Cliffs (1994).

[Murray, 04] Murray, T.: "Design Tradeoffs in Usability and Power for Advanced Educational Software Authoring Tools"; Educational Technology, 44, 5 (2004), 10-16.

[Papanikolaou, 03] Papanikolaou, K. A., Grigoriadou, M., Kornilakis, H., Magoulas, G. D.: "Personalizing the Interaction in a Web-based Educational Hypermedia System: The case of INSPIRE"; User Modeling and User Adapted Interaction, 13, 3 (2003), 213-267.

[Reigeluth, 99] Reigeluth, C. M. (Ed): "Instructional Design Theories and Models: A New Paradigm of Instructional Theory"; Vol. II. Lawrence Erlbaum Associates, Mahwah (1999).

[Reload, 06] Reload Learning Design Editor, version 2.1.3
http://www.reload.ac.uk/ldeditor.html.

[Specht, 02] Specht, M., Kravcik, M., Klemke, R., Pesin, L., Hüttenhain, R.: "Adaptive Learning Environment for Teaching and Learning in WINDS"; Proc. AH2002, Springer, Berlin (2000), 572–575.

[Spiro, 90] Spiro, R., Jihn-Chang, J.: "Cognitive Flexibility and Hypertext: Theory and Technology for the Nonlinear and Multidimensional Traversal of Complex Subject Matter"; Cognition, Education, and Multimedia: Exploring ideas in high technology, Lawrence Erlbaum Associates, Hillsdale (1990), 163-205.

[Stash, 07] Stash, N.: "Incorporating Cognitive/Learning Styles in a General-Purpose Adaptive Hypermedia System"; (2007). PhD doctoral Thesis, Technische Universiteit Eindhoven, Eindhoven.

[Tattersall, 04] Tattersall, C., Manderveld, J., Hummel, H., Sloep, P., Koper, R., De Vries, F.: "IMS Learning Design Frequently Asked Questions"; (2004). Retrieved May, 2006, from http://hdl.handle.net/1820/116/.

[Tattersall, 07] Tattersall, C., Sodhi, T., Burgos, D., Koper, R.: "Using the IMS Learning Design notation for the modelling and delivery of education"; In L. Botturi (Ed.). Handbook of Visual Languages for Instructional Design: Theories and Practices (in press).

[Towle, 05] Towle, B., & Halme, M.: "Designing adaptive learning environments with learning design"; In R. Koper & K. Tattersall (Eds.), Learning Design, Springer, Berlin (2005), 215-226.

[Van Merriënboer, 97] Van Merriënboer, J.J.G.: "Training complex cognitive skills. A four component instructional design model for technical training"; Educational Technology Publications, Englewood Cliffs (1997).

[Weibelzahl, 05] Weibelzahl, S.: "Problems and pitfalls in the evaluation of adaptive systems"; In S. Chen & G. Magoulas (Eds.). Adaptable and Adaptive Hypermedia Systems. Hershey, PA: IRM Press (2005), 285-299.

[Wu, 00] Wu, H., De Bra, P., Aerts, A., Houben, G. J.: "Adaptation Control in Adaptive Hypermedia Systems"; Proc. AH 2000, Springer, Berlin (2000), 250-259.