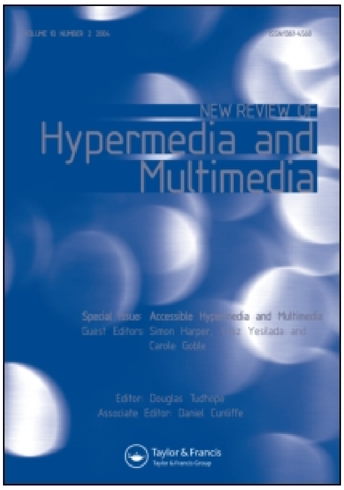


New Review of Hypermedia and Multimedia
Publication details, including instructions for authors and subscription information:
http://www.informaworld.com/smpp/title~content=t713599880

AH 12 years later: a comprehensive survey of adaptive hypermedia methods and techniques

Evgeny Knutov [a]; Paul De Bra [a]; Mykola Pechenizkiy [a]
[a] Department of Mathematics and Computer Science, Eindhoven University of Technology, Eindhoven, The Netherlands

Online Publication Date: 01 April 2009

To cite this Article Knutov, Evgeny, De Bra, Paul and Pechenizkiy, Mykola(2009)'AH 12 years later: a comprehensive survey of adaptive hypermedia methods and techniques',New Review of Hypermedia and Multimedia,15:1,5 — 38

To link to this Article: DOI: 10.1080/13614560902801608

URL: http://dx.doi.org/10.1080/13614560902801608

New Review of Hypermedia and Multimedia,
Vol. 15, No. 1, April 2009, 5–38

AH 12 years later: a comprehensive survey of adaptive hypermedia methods and techniques

EVGENY KNUTOV*, PAUL DE BRA and
MYKOLA PECHENIZKIY
Department of Mathematics and Computer Science, Eindhoven University of Technology,
Eindhoven, The Netherlands

A hypermedia application offers its users much freedom to navigate through a large hyperspace. Adaptive hypermedia (AH) offers personalized content, presentation, and navigation support. Many adaptive hypermedia systems (AHS) are tightly integrated with one specific application and/or use a limited number of techniques and methods. This makes it difficult to capture all of them in one generic model. In this paper we examine adaptation questions stated in the very beginning of the AH era and elaborate on their recent interpretations. We will reconsider design issues for application independent generic AHS, review open questions of system extensibility introduced in adjacent research fields and try to come up with an up-to-date taxonomy of adaptation techniques and an extensive set of requirements for a new adaptive system reference model or architecture, to be developed in the future.

Keywords: Adaptive hypermedia; User modeling; Adaptive techniques; Reference model

1. Introduction

The research field of adaptive hypermedia (AH) and adaptive web-based information systems (AHS for short) has been growing rapidly during the past 15 years and this has resulted in new terms, models, methodologies, and a plethora of new systems. Adaptive systems are becoming more popular as tools for user-driven access to information. Adaptation of an information system or service to a user has been proven to be a powerful and useful concept (Brusilovsky 2001). It is particularly helpful for the reduction of the information overload which is frequently experienced on the Internet or any other information system of a large scale.

Since this explosion in the Adaptive Hypermedia Systems (AHS) area, only a few general overviews of the field have been made to capture all up-to-date techniques, methods, approaches, and applications. The latest was Brusilovsky's (2001) paper that presented an updated survey of AH methods and

*Corresponding author. Email: e.knutov@tue.nl

New Review of Hypermedia and Multimedia
ISSN 1361-4568 print/ISSN 1740-7842 online © 2009 Taylor & Francis
http://www.tandf.co.uk/journals
DOI: 10.1080/13614560902801608

techniques initially introduced (also by Brusilovsky) in 1996, being one of the most influential papers in this field.

In parallel, in 1999, a first reference model for adaptive hypermedia applications, called AHAM (De Bra *et al.* 1999, Wu 2002) was defined, and an implementation closely following this model, called AHA! (De Bra and Calvi 1998, latest publication De Bra *et al.* 2006) was made available to the research community. This reference model unified the AH research community and provided a generic architecture that induced research activities in many directions.

As we have stated in the title of this paper—"AH 12 years later", our first and foremost aim is to provide a comprehensive overview of AH methods and techniques since their introduction 12 years ago and at the same time also come up with a set of requirements and a modular structure that can be used to update the first generic AH model AHAM that was introduced 10 years ago.

Quite a few systems were developed in the past 10–12 years, mostly providing facilities for e-Learning (or Technology-Enhanced Learning as it is sometimes called) which was considered as a primary application area. Examples are KBS Hyperbook (Henze 2000), APeLS (Conlan *et al.* 2002a,b), Interbook (Brusilovsky *et al.* 1998), WINDS (Sprecht *et al.* 2002), MOT (Cristea and De Mooij 2003b), RATH (Hockemeyer *et al.* 1998), etc.

A few attempts have been made to extend the AHAM reference model or provide a new one. The Munich model (Koch and Wirsing 2002) tried to capture all major parts of the system architecture using the Unified Modeling Language (UML) notation. The Goldsmith model (GAHM; Ohene-Djan and Fernandes 2002) was later considered together with AHAM in an attempt to provide a unifying model of all three (AHAM, Munich, and GAHM). The comparison in fact didn't provide a unified description in terms of conceptual representation or adaptive techniques, bringing up mostly implementation and meta-data issues of those systems (Gorle *et al.* 2003).

Most of the new system developments have resulted in new terms, concepts, models, methodologies, and prototypes. All previously described ideas have been transferred to new situations, showing new use cases. The most recent surveys don't give an up-to-date overview of the AHS area and of adaptation principles in particular.

Although AH research has delivered a variety of systems for the same application areas, there is still no consensus as to what is the "ideal" architecture of such adaptive systems. Each development introduced new components, new interfaces, new adaptation techniques, etc. Pursuing the unified approach to AHS we will consider adaptation questions initially raised by Brusilovsky (1996) in respect to the current state of the art, giving explanatory examples of most commonly used AH systems and providing their specific details in comparison to each other, at the same time trying to understand and extract the essence of each adaptation model (AM). This article will cover basics and granularity of a domain model (DM) and user model (UM), peculiarities of the AM, consider goals and context models as terms of new developments and system decomposition. We will also take a look at Adaptive Presentation and Navigation techniques providing a taxonomy update.

As it is almost impossible to grasp all recently proposed and developed AHS, we will consider only the ones we think are most important and interesting in the field and we will also take a brief look at examples that are very representative and may show some specific characteristic of any system. In terms of models we look at developments starting with the Tower Model (De Bra *et al.* 1992), including AHAM (De Bra *et al.* 1999, 2000, Wu 2002), the "Munich" model (Koch 2001, Koch and Wirsing 2002), GAHM (Ohene-Djan 2000, Ohene-Djan and Fernandes 2002), and Layered AHS Authoring Model and their corresponding Algebraic Operators (LAOS) (Cristea and Calvi 2003, Cristea and De Mooij 2003a, Hendrix and Cristea 2008). In terms of systems we consider systems that have a solid base in AH research and that continue to be subject of research and development, including AHA! (De Bra and Calvi 1998, De Bra *et al.* 2006), KBS Hyperbook (Henze 2000), APeLS (Conlan *et al.* 2002a,b), and Interbook (Brusilovsky *et al.* 1998). We will also touch upon important and solid developments such as TANGOW and TANGOW-based systems (Carro *et al.* 1999, Carro *et al.* 2003), GOMAWE (Balik and Jelinek 2007), CoMoLe (Martin *et al.* 2006), and others in order to show clearly expressed differences or provide arguments for the new trends that we consider in Section 3.

As a result we will sketch a modular structure for an AHS reference model that is still to be developed and that will capture the state of the art and the main new trends which may not yet be part of any AHS or may not yet be considered at all as a part of AHS functionality. As well we'll describe best practices and new research methodologies in AHS area that proved their right to exist (being researched and implemented within a number of AH and related projects).

2. Questions of adaptation

The core of adaptation is defined by posing and answering six major questions:

- What can we adapt? (What?)
- What can we adapt to? (To What?)
- Why do we need adaptation? (Why?)
- Where can we apply adaptation? (Where?)
- When can we apply adaptation? (When?)
- How do we adapt? (How?)

This type of classification has been initially introduced in Brusilovsky (1996), where a classification of AH methods and techniques was presented (see figure 1). The main purpose of the current paper is not to just revisit these questions, methods, and techniques but to address the issue of aligning

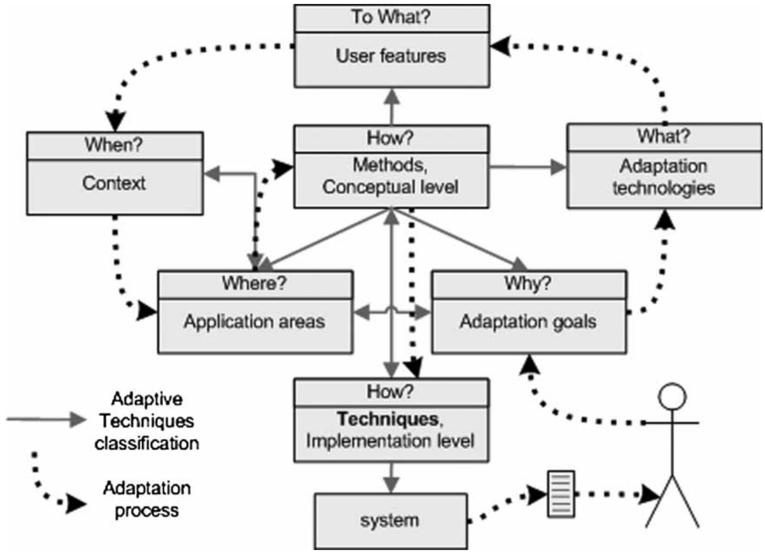


Figure 1. Classification of AH methods and techniques, adaptation process highlights.

all questions (and their answers) in a common, modular structure of a generic purpose AHS architecture. To this end we will also revise the meaning (or definition) of some of these questions in order to capture most recent trends.

Figure 1 considers the sequence in which the questions should be asked (and answered), thus leading to the definition of the adaptation process. By answering major adaptation questions we elaborate adaptation process description outlined in figure 1. This process is usually initiated by the user stating the adaptation goal and thus answering the "Why adaptation is needed?" question. Then in the process we consider the "What?" and "To What?" questions, which emphasize the DM and the UM description. "When?" and "Where?" in this process go next providing context and application area definitions. Lastly, the, "How?" question describing methods and techniques on conceptual and implementation level and finally all together resulting in an AH system description.

Previous (reference) models acknowledged that adaptation in a given application depends on three major factors:

1. The application must be based on a DM, describing how the conceptual representation of an application domain is structured. This model indicates relationships between concepts and how they are connected to content presentation in terms of fragments, pages (De Bra *et al.* 2000), chapters, information units (Henze 2000), pagelets (Conlan *et al.* 2002b), or any other structure encapsulating information about a concept. DM usually answers a "*What?*" question, providing a domain structure and information that needs to be adapted, linking concepts to a corresponding content representation. In this case linking of a concept and content structures should be carefully considered as a separate question as the way

this linking is being done may affect the system architecture, from providing authoring tools to make one-to-one correspondence to bringing up dynamic aspects of open corpus and having a topic resolving query linking concepts and resources.
2. A UM has to be created and kept up-to-date to represent user knowledge, interest, preferences, goals and objectives, action history, type, style, and other relevant properties that might be useful for adaptation. UM usually answers the "*To What?*" question, providing user and usage data using the information from DM. Quite often UM may answer the "*Why?*" question as well, providing information about user objectives using the same conceptual structure.
3. The System has to adapt the presentation, the information content and the navigation structure to the user's level of knowledge, interest, navigational style, goals, objectives, etc. Thus the AM has to be provided, indicating how concept relations in DM affect user navigation and properties update (for instance whether the system should guide the user toward or away from information about certain concepts). AM may be presented as a "teaching model" with pedagogical rules (De Bra *et al.* 1999), a "pedagogical model" (Henze 2000), a "narrative model" (Conlan *et al.* 2002b), or for instance including a glossary structure (Brusilovsky *et al.* 1998). In terms of providing adaptation flexibility, this model may answer the "*When?*" and "*Where?*" questions, as well as bringing a "*What?*" question up again, interpreting constraints on a DM relations structure.

This division into DM, UM, and AM provides a separation of the major AHS questions. However this division is still mixing up some of the questions (since it only has three parts, for six questions). A further specialization of the "parts" or "layers" is needed in order to achieve a better separation of concerns and offer enough granularity in the architectural structure.

2.1 AH reference models and systems

Reference models started having a "layered" architecture with the Dexter Model (Halasz and Schwartz 1990, 1994). In 1992 the *Tower Model*[1] was introduced (De Bra *et al.* 1992, van der Aalst *et al.* 1993). The Tower Model was an extensible data model for Hyperdocuments intended to serve as the basis for integrating hypermedia systems with other information sources, such as DBMS, IR systems, CAD tools, etc. To this end it had functional structures that can express adaptive and dynamic hypertext systems and applications. The Tower Model considered a layered structure, just like Dexter, but considered very explicitly the view (or projection) of each individual object through the individual layers, which led to the definition of the *Tower*. The model provided definitions of *nodes*, *links*, and *anchors* as first-class citizens, and offered modeling constructors to build complex information representations, such as *composite objects and cities.*

The tower constructor packaged together the multiple levels at which an object was described. These levels would include, among others, a structural description level, and a visual presentation level. For example, a text node tower description in figure 2.

These objects and functionality within the hyperspace were multidimensional, encapsulating different aspects in a "tower" from an issue or problem decision to a graphical rendering or a text representation. These different dimensions corresponded to different levels of a hyperdocument description and defined belonging to a different conceptual spaces, resembling one of the goals we stated to achieve a clear layer separation in a generic AHS. As a more advanced structure the "City" comprised by a number of towers gave an opportunity of viewing a hyperdocument from different perspectives. The "tower" and "city" constructs can be considered the basis for later models like LAOS. Having presented a set of modeling constructs that made it possible to integrate a wide variety of information sources into a hypermedia systems, The Tower Model predicted and provisioned the structure and dynamics of AH systems, indicated that a layered structure of hypertext systems that can be used to provide flexibility and interoperability of the system within different concept spaces.

In the following models and systems like AHAM, LAOS, KBS Hyperbook and others we can clearly identify where the major adaptation factors (described in Section 2) belong. The AHAM model layout matching the basic adaptation questions is presented in figure 3. We can do the same for the UML based Munich model in figure 4 and LAOS authoring model in figure 5. All figures underline the presence of major adaptation factors in each model and to some extent represent layered structure according to these factors.

In order to perform adaptation based on domain and user knowledge an "author" is required to specify how the system interaction results in different information presentation units based on DM. In AHAM, this is done by means of an AM consisting of adaptation rules. An adaptation engine (AE) interprets these rules to handle link anchoring and to generate the presentation specifications. AHAM uses Event–Condition–Action (ECA)

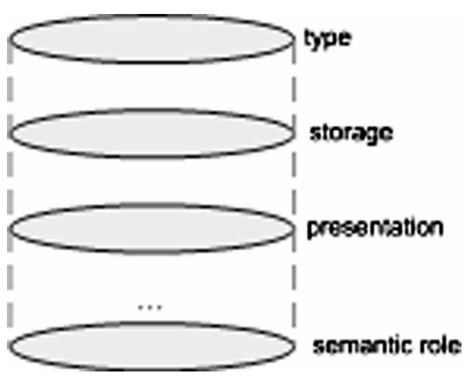


Figure 2. Tower model: a tower for a text node.

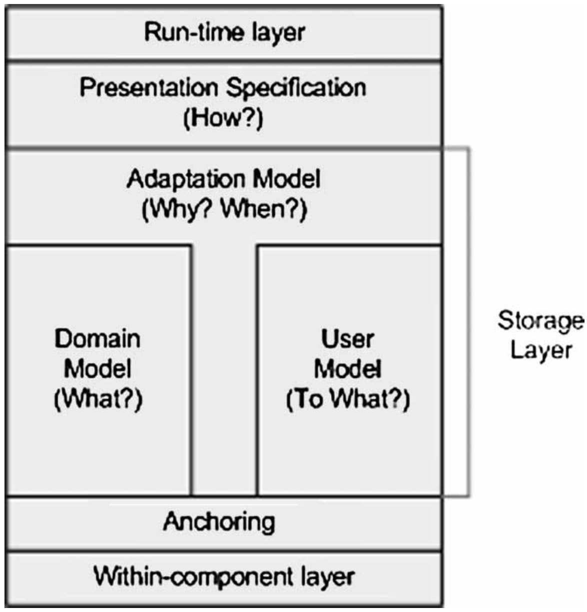


Figure 3. Adaptive Hypermedia Applications Model.

rules to describe the UM update and the adaptation processes, without requiring represented systems to actually do the same. In Wu (2002), the associated problems of termination and confluence problems have been considered, proposing static analysis of rules and a simple strategy for dynamic enforcement. The AE in the AHA! systems uses ECA rules. The same reasoning approach is applied in KBS Hyperbook, introducing also deduction rules, which are based upon the object-oriented conceptual modeling Telos language. The APeLS system uses Java Expert System Shell (JESS) which in fact represents facts that make certain rules applicable and then asserting them, which is of an ECA reasoning type.

Since ECA rules are low level they are difficult for authors to understand. Therefore, some AHS provide authoring tools that hide the actual ECA rules and offer higher-level constructs, which correspond to "concept relationships" and "concept relationship types" in AHAM. There can be also work-flow based constructs, concept type based rules or programming based constructs such as LAG or LAG-XLS languages (Cristea and Verschoor 2004, Stash 2007).

Below we are going to elaborate on the six main questions and place them within the context for the DM/UM/AM parts of the AHS.

2.2 What? The domain model (DM)

The domain model of an AHA usually consists of concepts and concept relationships. A concept represents an abstract information item from the application domain. In most of the systems the concepts form a hierarchy. As

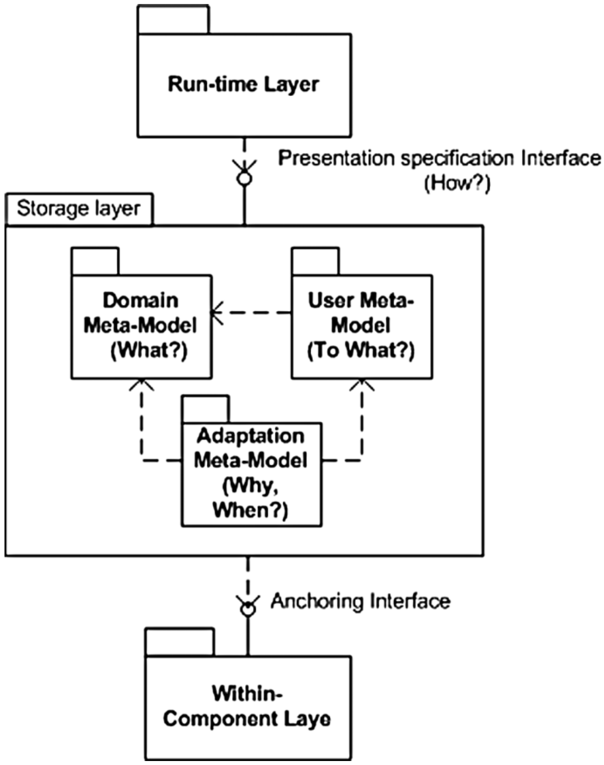


Figure 4. Architecture of Adaptive Hypermedia Applications (Munich model): UML notation.

a result each concept can be either an atomic (primitive) concept or a composite concept that has child concepts (sub-concepts) and a description of how they fit together. Some systems and corresponding authoring tools allow graph-based approaches. More complex ways of connecting concepts are also possible, as often done in defining subject domain ontologies.

In many AHS concept hierarchies and their representation may vary from system to system, providing indexing facilities like in Interbook (Brusilovsky *et al.* 1998), mapping domain concepts onto a document space which contains documents and test items (and the concepts themselves). Each textbook is structured as a hierarchy of chapters and sections with atomic presentations, tests, or examples. Interbook applies adaptive navigation support (but no content adaptation). The same hierarchical presentation can be traced in KBS Hyperbook (Henze 2000, Henze and Nejdl 2004), where the system uses a knowledge base which consists of so-called "*Knowledge Items*" or essentially concepts. In this respect each document from the document space is indexed by some concepts from the knowledge base which describe the content representation and hierarchical structure. In APeLS the concepts are encapsulated into a "*Narrative*" structure where each narrative can be hierarchically split into sub-narratives.

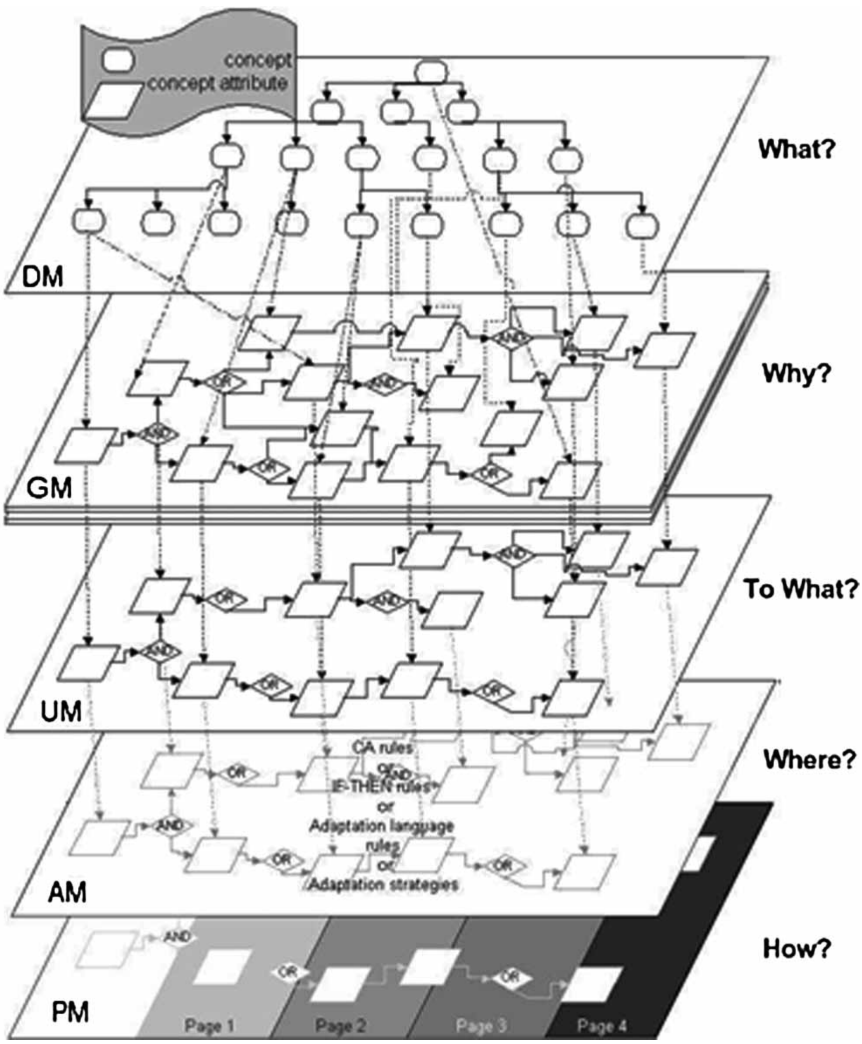


Figure 5. LAOS five level AHS authoring model.

Providing this type of DM structure (where all concepts are fine grained and hierarchically structured down to low-level representation primitives) makes it possible to apply adaptive techniques, working with fragments of fine-grained information units representing each concept and making adaptive presentation and navigation come into play. In general structures follow the same scheme of concept hierarchy (presented as a directed acyclic graph), providing arbitrary number of object enclosures. This allows us to apply adaptive techniques directly to a low level structure of fragments and pages performing user adaptive navigation and presentation support. Each system proposes its own way to encapsulate content information: in the form of a *Pagelet* (in APeLS), which contains content and a content model, representing general, pedagogical and technical information, and which may be assigned to a certain content group. Or it may be an *Information Unit* just encapsulating content information as in KBS Hyperbook. And these Information Units are indexed to map the *Knowledge Items* structure. In the AHAM model and in the AHA! system content representation is based on *pages* consisting of *fragments* (see figure 6). Whereas most systems have a fixed concept structure in TANGOW concept structures are reconfigured in different ways according to the rules and depending on the user for which this website/course is intended.

A concept relationship is a *meaningful* relationship between concepts. In AHAM it is represented as an object (with a unique identifier and attribute–value pairs) that relates a sequence of two or more concepts. Each concept relationship has a type (e.g. direct link, inhibitor, "part of", or prerequisite) which may play a role in the adaptation. Such a DM structure representation applies to most AHS systems. In KBS Hyperbook we may see the dependency graph of all the knowledge items (KIs), in AHA! We have binary relationships of arbitrary types (De Bra and Calvi 1998), and in APeLS we have a form of relationships map in a Narrative Model. In some of the systems or models (for instance LAOS) the "prerequisite" type is withdrawn from DM as it is more related to a certain variant of content interpretation.

In the Munich reference model (Koch 2001, Koch and Wirsing 2002) a more formal UML notation DM view can be found, presenting all relations and entities in terms of UML associations, compositions, interfaces, links, and packages, providing a formalized overall intuitive visual representation and a formal unambiguous specification of an AHS model. Two basic classes of a DM are *Component* and *Domain*. The Component structure is represented by an abstract *Component* class that can be either a *Concept*

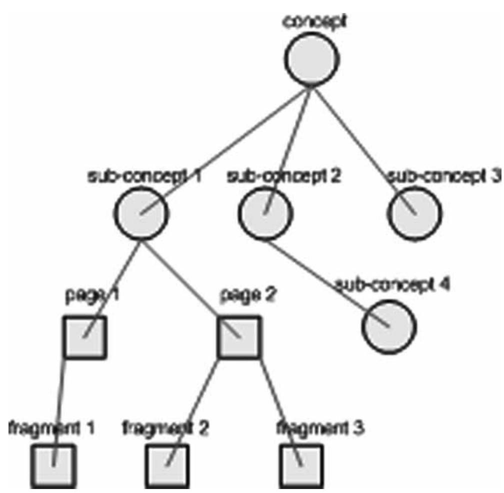


Figure 6. Concept hierarchy as represented in AHAM and AHA!.

(class–concept), which in turn can represent *Atom* or *Composite* class or concept relationship (*class–concept relationship*).

Considering yet another generic purpose AHS model GAHM we can see that here personalization is essentially carried out by handling *hyperpages*, which are defined to be a sequence of certain chunks, each of which is comprised of a *content specification* or so-called *C-Spec*, which may be presented in a form of data values or requests to a database and may be associated with a set of *template variables*, which can be marked as a placeholder for the content. Carefully considering this combination of *C-Spec* and *R-Spec* or rendering specification (which in turn describes how content has to be rendered) we may conclude that the aforementioned specifications to some extent can be mapped (by providing a description of functionality overlap of each system's sub-components/models) onto the AHAM and Munich models. In respect to a DM template variables and content specifications represent the conceptual structure of a Domain.

The (above) typical approach at defining a domain model as a set of concepts and concept relationships does not take into account dynamic aspects such as the construction of *goals* or *tasks* as structures over domain model concepts. Since the adaptation is moving toward a more intelligent process taking into account user interaction toward certain objectives, perhaps following a certain workflow in a highly dynamic context we see an emerging need for a separate model or layer to handle the "Why?" question, which we deal with in the "Goals and Tasks" Section below.

Another trend is to attempt to utilize the DM as an ontology or vice versa. For instance providing an integration model (IM) and integration model ontology (IMO), which allows specifying a DM and ontology mapping as mentioned in Vdovjak and Houben (2002) and Aroyo *et al*. (2004). Alternatively the GOMAWE system, Balik and Jelinek (2007) proposes a model based on a semantic data representation which can be easily utilized in process automation and knowledge reuse across applications. AHAM can almost handle the single ontology case (because it considers concepts and arbitrary concept relationships), however, it has no provisioning dealing with multiple ontologies. In this respect making the reasoning on the Semantic Web is becoming more challenging than initially thought. Research into reasoning over different ontologies will become one of the core AH research areas, because one cannot assume that different applications of which the adaptation must be combined are using the same ontology (Vdovjak and Houben 2002, Aroyo *et al*. 2004, Carmagnola *et al*. 2005, Balik and Jelinek 2007, Dietze *et al*. 2007).

In general the DM is considered to be a static structure being defined and authored by a domain expert, which implies that adaptation can be provided only within the bounds of a Domain modeled knowledge space. However moving toward open corpus adaptive systems defined in Brusilovsky and Henze (2007) is becoming one of the challenges in the AHS research field. It aims to extend AHS with the possibility to operate on an open corpus of documents, which is not known at design time and in addition to this can be constantly changing and expanding (Brusilovsky 2008).

If an AHS has to deal with the "open corpus" document space the problem of mapping concept(s) to content arises. Having a great variety of content structures we may have: (1) one-to-one concept to content matching; (2) selection of content resources which results in one-to-many relations; (3) a link which is represented in a query (concept query) (e.g. topic resolving query) to provide content resources to be mapped with a certain concept; or (4) one resource may be a (partial) match to different concepts. Even though when combining just two DMs the task of reasoning is becoming challenging, in the case of "Open corpus" it becomes even more difficult in terms of concept and content alignment. As the "consensus" as to how concepts and content (resources) match may change over time the concept to content mapping problem is related to the research topic of concept drift (Morita *et al.* 2006, Tsymbal *et al.* 2008, Xie *et al.* 2008).

Having done a brief overview of core adaptive functionality in terms of rules (basically represented by ECA type rules) which are interpreted by adaptive engine to deliver user navigation and presentation support, we didn't mention a challenging idea of *higher order adaptation*. Although most of the systems adapt to one parameter (recommender systems adapt to what they think the user interests are, learning systems adapt to what they think the user's knowledge is, some systems perform device adaptation), more advanced systems can do adaptation to more than one parameter at once or can "monitor" the user in order to decide to change they way in which the adaptation works. For instance in Stash (2007), the user's *learning style* is monitored, and as the observed learning style is detected (or changed) the way in which the system adapts also changes. More in general a *second order adaptive system* would use machine learning techniques to discover usage patterns and adapt the way in which it adapts to the user or provide the following information to a domain expert for more accurate refinement. In general there may be no limit to adaptation orders: a system may learn how to adapt the way in which it learns how to adapt its adaptation strategy, etc.

In table 1 we present a summary of the DM functionality and specification approaches used in AHAM (and the AHA! implementation), KBS Hyperbook, APeLS, and Interbook, as the most representative comparison examples, even though we realize that other systems with some additional properties exist. Each row in the table presents a description of a particular system, its properties and aspects which we consider describe more or less the same system functionality in comparison to other systems. On the other hand the table shows all the differences both in approach, implementation and composition of each system, as well as a difference or similarity in terms used to describe system functionality. E.g. "Content Grouping" (is present only in APeLS and KBS Hyperbook) is implemented either in a way of grouping similar content pagelets in APeLS content groups or grouping a sequence of concept and associated content in Project Units fulfilling similar user objectives. In other systems grouping is not possible which we denote by "n/a".

Table 1. Summary of the domain model properties.

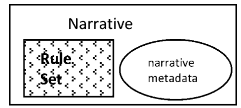
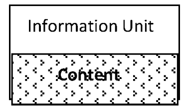
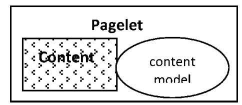
			AHAM	AHA!	KBS Hyperbook	ApeLS	Interbook
	Concept	An abstract representation of an information item from the application domain	Concept information: – <attribute, value> pairs; – sequence of anchors; – presentation specification; atomic concepts – represent single fragment of information; composite concepts use child attribute to specify sequence of composite concepts	Concepts like in AHAM with restrictions also have type; and associated with a template, (can have only fixed number of attributes)	Knowledge item (KI) – abstract representation of domain knowledge (e.g. if, class, run_method) (may also be a compound structure)	Encapsulated in narrative model metadata (each narrative may add a new concept and corresponding narrative rules) 	Glossary entries = domain concepts
Concept	Concept relationship	Represents semantic relationship between concepts	Authored semantic linking between concepts in a form of: <C1, C2, T, A> where (T = type) – link; prerequisite; inhibit; part (compositional) and (A = attribute value)	Types of relationships: fragment/link/contain	Dependency graph of the KIs = ≥semantic links between information units (IU). Each IU is connected to one or more KI presenting which concept represents corresponding content in IU	Is presented in a form of a relationships map in the narrative model	Concept relationships (*navigational paths between glossary items*) types: (1) first-page; (2) sub-section; (3) domain_ concept; (4) bookset; (5) loginpage; (6) requirement; (7) outcome; and (8) fragment

Table 1 (*Continued*)

			AHAM	AHA!	KBS Hyperbook	ApeLS	Interbook
	Indexing	Explicit indexing options: mapping concepts, projects, etc.	n/a	n/a	Knowledge items (KI) – index Project units and information units	n/a	Glossary entries index domain concepts. Concepts are indexed on textbooks (bookshelves)
Content	Content data presentation	Content unit structures	Pages and fragments (page may consist of several fragments)	Pages and fragments	Information units (IU) 	Content information is presented in a form of a pagelet which may belong to a certain content group (see below) 	Content info is presented in a Textbook (shelf of textbooks). Glossary (glossary entries provide link to a certain textbook and connection to a certain domain concept)
	Content grouping	Content grouping according to similarity of presentation, objectives, etc.	n/a	n/a	Project units are mapped on information units. Project unit defines a number of KI that has to be learnt to fulfill project goal	Content group (content pagelets are organized in a group fulfilling the same learning objective (LO)).	n/a
	Storage	Content storage part of the AHS	Within-component layer	Within-component layer	Domain model	Content domain	Textbooks/ textbook shelves

2.3 To what? The user model (UM)

As an initial approach the adaptation process in adaptive systems was made based on user characteristics represented in the UM. Since that time many systems used their own approaches and/or adapted to something else rather than user characteristics. Kobsa (2001) suggested how to distinguish adaptation to user data, usage data, and environment data. User data points the way toward the adaptation goal. Usage data is a comprised data about the user interaction that still could be used to influence the adaptation process. Environment data comprised all aspects of the user's environment that are not related to the UM or usage process or behavior.

The UM usually consists of entities for which we store a number of attribute–value pairs. For each entity there may be different attributes, but in practice most entities will have the same attributes. Therefore, it can be a *table structure*, in which for each entity the attribute values for that concept are stored. Most entities in UM represent *concepts* from DM. Some entity instances may represent a user's background, preferences, interest, learning style, or even a platform or environment specific properties.

Usually the analogy between the structure of UM and DM is that UM is an *overlay structure* over DM, mapping the user's domain-specific characteristics like knowledge over the domain knowledge space. This is typically done by associating attribute values with each identifiable piece of user knowledge, interest, or other characteristic for each concept of a given domain. When considering different aspects about different concepts the table representation (using a *universal instance*) would result in a sparsely filled table, but alternative implementation structures do not suffer from this problem.

In general we have domain dependent and independent properties. Domain dependent properties usually are: user knowledge, test results, learning objectives, problem-solving tasks, or short-term objectives. Domain independent properties are: user credentials, preferences, cognitive and learning styles, user environment (time, place, equipment, etc.), and group affiliation if any. In this respect user experience or background can be considered as domain independent properties, however, in the case of overlap between domains background knowledge may be fitted again within the DM structure.

Dealing with an Overlay Model, the LAOS model tried to eliminate the classical UM overlay structure by avoiding "hidden" adaptation rules, representing UM as a concept map in such a way that relations between the variables in UM can be expressed explicitly (without the need to express UM concept relationships as DM relationships). LAOS also uses goals and constraints model (GM; Cristea and De Mooij 2003a) (uniforming ontological representation, Lassila and Swick 1999) to express goals and constraints separately from DM.

From another angle user properties are considered as static (covering user personal characteristics, such as age, gender, grade, or capabilities) and dynamic (which is the information about user interaction with the system such as knowledge, skills, motivation, plan, activity, or goal). An AHS must handle static and dynamic UM properties in a different way: it can just "use" static properties but it must "monitor" (changes to) dynamic properties as well as use them for the adaptation.

Considering this generic and quite popular overlay approach we may easily identify that for instance in KBS Hyperbook the user is modeled as a current state of his/her knowledge snapshot at each time (overlapping DM KI vector structure). The Learner Model in APeLS is authored to meet a DM structure and is represented in a set of concepts and user knowledge of each concept. It may also contain user prior-knowledge, learning style and user objectives. However, we will consider the user goal question separately in Section 2.3 to have a clear separation of AHS system layers (similar to the LAOS system). There are no strict rules saying that we can't use task representation within UM and treat it together with user characteristics. However, we should first consider the question of system usability and therefore treat task representations and application models separately, in order to pursue system models independence and interoperability.

We may also identify the property reflecting the way knowledge is deduced and stored in UM. Most of the systems (such as AHAM and Interbook) use a conventional scheme of updating knowledge level basing on the DM concept competence and keeping it in UM overlay. Others (KBS Hyperbook) use probabilistic approach by means of Bayesian network calculating the conditional probability that knowledge "x" is known to the student under the condition which is denoted by previously detected information about this student.

As defined in the AHAM reference model the user model may also consist of a *persistent* part and a *volatile* part. For each concept attributes of which the value is maintained were considered (for instance page was read or what is the level of knowledge). In this respect an AHS could recalculate some other attribute values on the fly. Some AHS may verify prerequisites satisfaction for a concept each time it is accessed or when a link to it is shown (backward reasoning), while another AHS may calculate and store prerequisite satisfaction each time it changes, for instance as a *ready-to-access* attribute in UM. For the future we foresee a new scenario where the AHS may already precompute (and store) the UM states that would result from future possible interactions like following a link. This thus allows the system to serve adapted information more quickly than when the (forward or backward) reasoning only starts when the user actually performs that interaction (follows that link). A smart system may predict the most likely future interactions and precompute several steps into the future (almost like what a chess program does). From a model point of view this can be considered as performance optimization. In future, however, the predictions may also be shown to the users at which time they will influence the interaction and thus also need to be incorporated in the model of the system architecture.

In table 2 we present a summary of domain dependent and independent UM properties as they are presented in AHAM (and the AHA! implementation), APeLS, KBS Hyperbook, and Interbook systems/models. We also consider goals and objectives here as a part of UM, though in the following section we discuss a question of Goal model separately. Note that we concentrate on the *representation* of the user model, not on the process of *obtaining* or even *deducing* that information. That itself could fill an entire journal article. It would also necessitate considering interaction other than reading and navigation.

2.4 Why? Goals and tasks

In the previous section we already mentioned that the user goal can be considered a user property that can be stored in the UM. However, we also saw that this is not the most natural approach. When considering goal-driven adaptation in existing AHS we cannot achieve good adaptation by just considering goals as user model properties. A goal is becoming not just an objective that has to be fulfilled, but evolves into a hierarchical structure of goals, objectives, tasks, requirements, workflows, depicting a more task oriented, and procedural approach. A "Goal Model" thus deserves to be a separate part of any up to date AH model.

An attempt to catch goal-driven adaptation has been made in KBS Hyperbook, where user defined or proposed system tasks have been mapped onto "Projects" units, each representing an index of "knowledge items" (essentially concepts presentation in the system), providing an elaborated task approach, where "projects" are meant to be real application issues that can be faced by performing a certain sequence of tasks (learning in terms of the e-Learning approach and orientation of KBS-Hyperbook application), each consisting of dealing with a new concept. Thus having a diverse structure of "projects" one may fulfill different application goals having basically the same DM and UM structures, being used all over in AHS.

Quite a similar approach has been followed in APeLS, where a "learning object" (LO) instance was able to fulfill a learning requirement, which was mapped to a certain content group, providing a choice of content depending on the user's objective. At the same time LO are coupled with "narratives" to provide a domain dependent structure.

In the LAOS framework (Hendrix and Cristea 2008) a goal separation approach has been considered more elaborately, proposing the "Goals and Constrains Model" (GM). This model essentially filters useful domain concepts and groups them together, according to the goal. Because GM is a separate layer it allows the formation of goals that deal with more than one DM. The GM defines concept relationships that do not belong to the domain model but only define structures needed to satisfy a goal.

In TANGOW/COL-TANGOW (Carro *et al.* 1999, 2003) or CoMoLe (Martín *et al.* 2006) systems there is a set of tasks to be accomplished by users. These tasks are proposed at different times to different users according to the state of their UM and context, which makes the task dependent not only on the UM but context of usage as well, which can not always be expressed through UM properties. We will consider context questions in Section 2.4.

As a generalization of goal centric approach one may think of creating a hierarchy of goals and corresponding tasks comprising this goal, workflows

Table 2. Summary of domain independent and domain dependent user model properties.

		AHAM	AHA!	KBS hyperbook	APeLS	Interbook
User goal/objectives	Overall learning goal stated by interaction with user	User follows a link to a (different) page	User follows a link to a (different) page	1 – for direct guidance; 2 – for goal based learning: knowledge items (KI) to be learnt are selected by user. goal (with triggering event for AE) consists of KI array. 3 – for project-based learning: goal and project repository	Learning objective – state the goal of learning procedure	User stated/ assigned learning goal
User goal statement	Goal statement by the user	n/a	n/a	1 – user defined; (2) proposed	1 – user defined	1 – user defined
System internal objective	Goal interpreted in terms of adaptive engine (AE) and domain model (DM)	Concept to learn (one step at a time) (stated with triggering event for AE)	Concept to learn (one step at a time)	Project (consists of project units mapped on KI) or KI to learn for guidance tour to reach a certain goal	LO is mapped to a certain content group that has to be learned (decision on LO can be done runtime (based on learner and environment information)	Represented as a set of concepts to be learned
Properties Domain independent	User common static parameters	Yes	Yes + authored attributes	Yes	Yes	Yes
	Experience/ background	n/a (not stated explicitly, but can be considered and expressed in UM)	n/a (not stated explicitly)	n/a	n/a	n/a
	Preferences (font types, pictures, examples, size, etc.)	n/a	Link coloring (default or defined)	n/a	n/a	n/a

Table 2 (*Continued*)

		AHAM	AHA!	KBS hyperbook	APeLS	Interbook
	Cognitive/learning style	n/a	Can be authored (not offered as a default option)	n/a	Supported via narratives (each narrative supports different pedagogical approach dealing with the same course meeting the same LO)	n/a
	Explicit user environment settings (time, place, etc.)	n/a	n/a	n/a	(for example, device dependent narratives mentioned	n/a
Domain dependent	Knowledge	Represented by an array of concept and a number of attributes for each content entity (<attribute, value >pairs) representing user knowledge of each concept (knowledge, interest, etc.)	Represented by an array of concept and a number of attributes for each content entity	Knowledge vector – KV = array of knowledge items [K1, K2, … Kn], each is weighted according to user confidence in this knowledge	Competencies learned – describes users prior-knowledge described with the same vocabulary (concepts) as narrative (DM)	Knowledge attribute – value estimating users knowledge on each concept
	Learning objectives	n/a (tracked by AE)	n/a (tracked by AE)	n/a	Competencies required – describes user learning goal (minimum knowledge learner should acquire to complete a course)	
	Problem-solving task (short-term user goal)	Yes (next page guidance = local guidance)	Yes (next page guidance = local guidance)	Direct guidance	Yes (course authoring dependent)	

that need to be followed to complete a requirement. Such a hierarchical structure should be aligned with a DM to describe mappings between Models in order to have better adaptation results. In this case we may think of a Goal Model which might have the same structure as an overlay with the DM and UM correspondingly to provide concept sequences for higher-level goals representation.

As another aspect of the goal-driven paradigm of adaptive systems we can consider deducing a goal from what other users have been doing within a given hyperspace. This may provide goal inference and recommendations to follow or just leading the user by previously discovered navigation patterns. In this case we may say that this goal has been learnt from other users' interaction with the system, which is opposite to the classical example where goals and corresponding task are usually assigned to or chosen by users from a known set of objectives. So a new generation of AHS system should be capable of versatile goal assignment, either when it was created and given by a domain expert or proposed/recommended by the system itself (e.g. modeled in Mei and Easterbrook 2007).

2.5 Where? and when? Application and context models

When talking about AH Systems in general, there is a wide range of application areas; however, the major one still remains e-Learning or Educational hypermedia with a great diversity of systems. On-line information systems, which cover fields from cultural heritage (for example, Bohnert *et al.* 2008, Rutledge *et al.* 2008, Stash *et al.* 2008) to TV guides (Bellekens *et al.* 2008, Tintarev and Masthoff 2008) or Social web aspects (Farzan and Brusilovsky 2006, Priedhorsky *et al.* 2007). This diversity is becoming richer each year. One may think of providing adaptation in consumer devices or medical industry. We will not cover every application area of AHS, moreover it is becoming even more difficult to capture the whole scope of constantly appearing systems and system approaches.

We will rather focus on context issues that started playing an important role in AH systems. Context aware systems gain popularity, however, context awareness is usually very field-dependent. Most of the time these are context-sensitive user interactions (Ardissono *et al.* 2008), providing context-based navigation or presentation support (Paris *et al.* 2004, Stober and Nurnberger 2006) or context-aware adaptive process management (Ardissono *et al.* 2007). Context awareness in some sense may replace the definition of application area or environment, allowing the system to be decoupled from a narrow field of application to a broader concept of context which may vary, providing system flexibility with different or evolving context. An Adaptive System therefore should be able to track this dynamic and evolving context.

Combining Adaptive architectures and Ubiquitous Computing results in a semantic interoperation-based approach to creating context-informed adaptive applications that makes maximum use of rich content as presented in O'Connor and Wade (2006), or context sensitivity within content-based filtering recommender systems like in Chedrawy and Abidi (2006). This may also be a context-based recommender system which is based on different adaptation filters to recommend individual or collaborative activities to the users according to different type of user features, behavior, and usage context like in CoMoLe system (Martín *et al.* 2006). Though context may not always be applied in terms of AHS, a reference model of such a system should be designed taking into account context awareness and sensitivity aspects.

As should be clear from the description above the term "context" applies to both the application (context) in which adaptation can be applied and to the environment (context) in which the application is used. The application-dependent adaptation decisions correspond to the question *where* the adaptation is done (this also conforms with the original Brusilovsky's classification of AH methods and techniques), whereas the application-independent environment of use context adaptation (e.g. time, day of the week, network bandwidth, etc.) corresponds to the question *when* the adaptation is done. In the new model, because of this difference, the *where* and *when* questions thus belong in a different layer.

2.6 How? Adaptive techniques and methods

Adaptive techniques and methods refer to methods of providing adaptation and their generalization correspondingly. Techniques are usually a part of implementation layer of an AHS and can be characterized by a specific approach or algorithm. Methods represent generalizations of a technique. Every single method shows a clear idea of adaptation approach, but at the same time each method can be implemented by a number of different techniques. Likewise some techniques may be used to implement several methods using the same knowledge representation. This set of techniques and methods comprises a toolkit of AH (Brusilovsky 1996). Both techniques and methods can be applied to content, presentation and navigation adaptation. In Brusilovsky (1996) adaptation to presentation was not considered separately.[2] In this paper we distinguish *adaptive presentation* far beyond Brusilovsky's *content and navigation techniques.* Some forms of content adaptation really only change the presentation, and some forms of adaptive navigation support do not change the possible navigation but only change "suggestions" by changing the presentation. We decided to differentiate the three forms of adaptation and to present them in a single diagram in figure 7.

The use of adaptive techniques has changed as AHS have matured. Especially in the field of education AHS have their origin in Intelligent Tutoring Systems where all the adaptation decisions (like what to show to the user and which steps the user should take next) were taken solely by the system. Some adaptation techniques still enforce a system decision upon the user, like hiding a fragment of text or removing a link. But in AHS the trend is to offer users more and more control. This has resulted in the techniques that we show below as "adaptive presentation". They do not change the

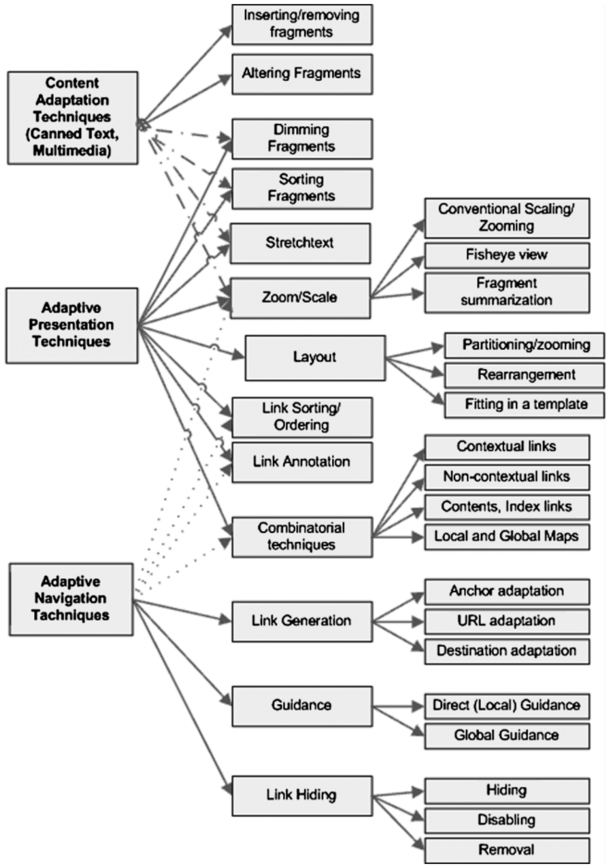


Figure 7. The new taxonomy of adaptation techniques.

information or the possible navigation, but only use presentation variations to make suggestions to the user.

The use of particular adaptive techniques is also influenced by the increasing use of online assessment to more accurately measure user knowledge. When compared to early systems modern systems measure user knowledge as well as other properties more accurately (Barbosa Leon *et al.* 2005, Challis 2005). Therefore, having more precise measurements, more observable characteristics-AHS can use a wider range of techniques best suitable for each stored instance of information or user profile properties to provide better adaptation results.

2.6.1 Content adaptation support. The presentation of information can be influenced essentially in two ways: by showing/hiding the information or by emphasizing/deemphasizing it. The essential difference here is whether the information is accessible or not. When inserting, removing, or altering fragments the information content is really changed. Other techniques: dimming, sorting, zooming, and stretchtext keep the same information available but suggest to the user to only read part of it. This suggestion is made through changes in the presentation, which is why we also place them under "adaptive presentation techniques". The techniques of zooming, which is recently introduced in Theophanis and Schraefel (2003) and stretchtext are useful for additional explanations which need not be read by every user. We would also like to distinguish three different types of scaling/zooming technique. The one mentioned before is a conventional technique providing content (irrespectively to the information type) scaling or zooming, changing the text font, zooming in or out a complete web page or only a pictorial part of it, or scaling down images that appear in the presentation. A "fisheye view" allows us to have a different view on information content or a link structure; that's why we have associated this technique with adaptive navigation support as well. In a fisheye view certain details are kept visible (readable), whereas other details are scaled down a lot, aggregated or deleted entirely. The last one is a "fragment summarization" (for example, text summarization when text is analyzed statistically and linguistically and a summary text is generated from these important sentences). In stretchtext only the title is shown whereas in zooming/scaling the entire information content is shown, but it may be scaled down (zoomed out) so much that it becomes unreadable using a conventional scaling technique. The user can decide to select/open the information so that it becomes readable (in full size). This is like placing a magnifying glass over the presentation that was scaled down. Accessing the information may also cause user model updates so as to influence the adaptive selection of zoomed information in the future.

2.6.2 Adaptive navigation support. The most complete revision of adaptive navigation instruments can be found in Brusilovsky (2007). That paper provides an extensive overview of adaptive navigation techniques and methods that are becoming increasingly important in various aspects of adaptive applications from web-based hypermedia to virtual reality. It reviews all major approaches, technologies and mechanisms giving illustrative examples. In this respect we will provide just a taxonomy of adaptive navigation techniques and mechanism used in AHS.

There are two ways in which the user's navigation can be influenced: enforced or suggested. The "guidance" techniques present recommended links, which can be obtained either through adaptively selecting links from a larger list (and hiding/removing the non-recommended links) or by generating destinations for predefined link anchors. In all these cases the structure of possible navigation paths (and links) is altered in a way that forces the user to select a link from a "computed" set of links.

Most adaptive navigation research focuses on adaptive navigation support that does not restrict the user but rather provides suggestions as to which links or paths are more appropriate than others. Sorted lists of links (placing the strongest recommendations at the top) and link annotations using colors and/or icons help the user in deciding which links are appropriate and which are not, but the user is not forced to follow these recommendations. The recommendations are made by changing the presentation, which explains why these techniques also fall under the "adaptive presentation" category.

Direct guidance using not just a single step but whole suggested paths were introduced for instance in KBS Hyperbook, where users were provided with "guiding trails". Adaptive link sorting is beginning to show up in personalized search engines. Link hiding, with its variants of hiding, disabling and removal, is most commonly used in the AHA! system. Link annotation is used in ELM-ART (Brusilovsky *et al.* 1996) and its descendents, including Interbook (Brusilovsky *et al.* 1998). The link generation technique can be found in Yan *et al.* (1996) and Lutkenhouse *et al.* (2005), but is essentially also the technique used by Amazon.com to provide its recommendations. A more complete survey of Recommender systems can be found in Adomavicius and Tuzhilin (2005). We can anticipate the use of three types of "link generation" techniques which may result in "anchor adaptation", "URL adaptation", and "destination adaptation". (All three are possible in AHA! for instance, but have mainly been used just to show their existence.) Initially introduced in Brusilovsky's (1996) paper the "page variants" technique can be explained as a case of destination adaptation. The main difference between "URL adaptation" and "destination adaptation" is that with the former the decision as to which link destination to use is made when the page containing the link is generated, whereas the latter always shows the same link destination (URL), but when the link is accessed the server will decide which actual destination (or page variant) to return.

More powerful techniques can be defined as combinations of previously mentioned approaches to link adaptation. These are "contextual links" embedded into the context of the page, "local non-contextual links" which may include all types of links on a regular page (like links, buttons, lists, pop-ups, etc.). "Links on local and global hyperspace maps" provide graphical representation of local or global hyperspace navigational structure in a network form of nodes. The same approach of a global map structure can be seen in providing linking from table of contents or index page, which in fact does support a kind of "pre-defined" navigation, but it can be useful in a particular type of application.

2.6.3 Adaptive presentation support. As we saw above, adaptively changing the presentation can be used to either emphasize/deemphasize part of the content (that is all accessible) or to suggest links to users. However, there is also adaptation to the presentation that is applied for entirely different reasons, like device adaptation or layout preferences.

Layout adaptation can be needed because content (especially in open corpus applications) needs to be presented within a predefined presentation format. Research in the GRAPPLE[3] project aims at integrating adaptive learning environments (ALEs) which is an adaptive system supporting teaching and learning in an educational setting, with learning management systems (LMS), which are used to deliver, track, and manage training process (De Bra *et al.* 2008). Depending on the LMS an information page to be presented may need to be placed in different frames/windows, and automatically generated view on the navigation structure may be included or may need to be omitted.

Another situation in which layout adaptation is needed is when adaptive presentations need to be adapted to devices with limited capabilities. A large presentation can for instance be scaled down, with the ability to zoom in to parts of it (one at a time), or the presentation may be partitioned into subpages that can be selected and viewed one at a time. Parts of information may also be presented within a predefined template layout which is reflected in presentation specification (for example, using CSS web site templates with two columns and left navigation or just one column and right navigation).

2.6.4 Adaptive multimedia presentation. Nowadays a lot of photographic and multimedia content is described with extended metadata that can be used for/by adaptation. Moreover constantly extending image repositories, web services, tagging techniques, basic image operations which most of the devices are capable of, starting from computer software to embedded devices, and internet applications have appeared. Even if these new technologies or image metadata are not available everywhere (e.g. on a handheld device) it is still possible to make use of image basics—width and height. Having a look at the aforementioned taxonomy of content adaptation we see the part that applies to adaptation in a multimedia context. The techniques that apply to textual content adaptation apply (viewed at an abstract level) to pictorial information as well.

We show a few use cases of adaptation to pictorial information below:

Conditional image inclusion may be quite useful in device adaptation, where only a key part (tagged with some concept; may be a thumbnail) or just resized image will be shown on a small-screen device or a device with low-bandwidth capabilities for example. In case of image resizing generating adaptive presentation becomes very simple since it doesn't require any extensive metadata from an image, but uses only image dimensions (see figure 8). In this case the Zoom/Scaling technique is the best to be used.

As a "stretchtext/stretchimage" technique example we can think of expanding a single image to a set of pictures or a picture timeline, extending presentation to provide rich multimedia experience and fulfill curious user goals (see figure 9) or just present a thumbnail of an image. In Stash (2007), adaptation to the visual/verbal learning style dimension uses the "stretch-image" technique.

In Hanisch *et al.* (2006), there has been an attempt to extend the taxonomy with multimedia components. It has been presented as a number of multimedia components used by the altering (fragments) technique, such as: models, views, controllers, widgets, graphics items, scripts, and strategies. Providing component alteration may result in a system that can change its internal representation (model), its specific view, and/or the one of the controls. Altering similar widgets may change the user interface, as well as changing different graphical items.

Figure 8. Conditional image device resizing.

2.6.5 Tools adaptation. Nowadays AH applications include not only content to be read, but also tools to interact with different resources. Although the question of tools adaptation is very specific and usually treated in a context of every single application and apart from application adaptation it may play an important role in the user modeling and adaptation process. Usually tools adaptation results in providing a different set of features to the different types of users: novice versus advanced users or group of users. For instance, tools used in collaborative workspaces (Carro *et al*. 2003) can be either selected or adapted to support collaborative task accomplishment. Also, in the AES-CS system (Triantafillou *et al*. 2004) field-dependent and field-independent users were provided with different orientation support tools (such as a concept map and path indicator). Despite the fact that tools adaptation is still a very specific field, a generic AM should be extensible to accommodate tools adaptation techniques and in instruments in a single adaptation process.

3. Summarizing new trends for a vision of future generic adaptive hypermedia systems (AHS)

Having given a brief review of existing and new approaches to building an AH system, we would like to summarize our vision on the future that will result in an updated AHS reference architecture, highlighting key points, which will incorporate new trends in AH research to provide greater adaptivity and

Figure 9. Stretch image to a picture set.

flexibility of the system. Several of the items shown below are already showing up in AHS, mostly in isolation, but in the future they will all need to be present in any general-purpose or generic AHS.

3.1 Ontologies

In many AHS authors create not only the information space but also the concept space for applications. In order to start combining the adaptation from different applications, taking advantage of what one AHS has learnt about the user in another AHS, the meaning of the concepts must be agreed upon. Therefore, instead of arbitrary conceptual structures adaptive applications are becoming based on ontologies. Combining the user models and the adaptation from different applications based on the same ontology is a feasible problem, but when different ontologies are used, the problem of ontology mapping must be tackled first, making the reasoning on the Semantic Web (Berners-Lee *et al*. 2001, Aroyo *et al*. 2004, Aroyo *et al*. 2007, Balik and Jelinek 2007) within the boundaries of AH field more challenging. Research into reasoning over different ontologies will become important for the AH research. For example the AHAM reference model can handle the single ontology case (as it allows arbitrary relationships between concepts) but has no provision for dealing with multiple ontologies.

3.2 Open corpus adaptation

Most AHS deal with a known set of information items, whether it is a single course, a "bookshelf" or a whole encyclopedia. In such applications a concept space can be mapped onto the document space by the author. Even though open corpus is not a completely new research field, adaptive applications increasingly consider open corpus adaptation, where resources come from search results in large and dynamic LO repositories or from a Web search engine. In order to perform adaptation to an unknown document space, the mapping between concepts and documents can only be done at run-time, bringing the fields of hypermedia, databases, and information retrieval together. One of the strongest research threads in parallel to AH since the very beginning was the Open Hypermedia (OH) research aiming to separate links from documents in order to handle hyperstructure separately from the media it relates to and trying to provide an alternative view of the AH from a contextually aware OH perspective (Bailey *et al*. 2002, Bailey *et al.* 2007). Recently defined in Brusilovsky and Henze (2007) open corpus adaptation in terms of, AH is receiving more and more attention providing new ideas and models in this area. Most of them introduce new approaches of adaptive navigation support in an open corpus space (Brusilovsky 2008) or trying to model linked open hyperspace from open-corpus resources, providing indexing for open corpus resources (both manual or automatic keyword-based) or introduce some community-based approaches. Currently no applications of content adaptation exist for open corpus applications (to the best of our knowledge). This is not surprising because it is difficult to foresee how the content (or presentation) should be adapted of pages that are newly found and are outside the control of the author of an adaptive application. In the future, however, we envision that content adaptation will become an integral part of open corpus applications, by extending technologies from natural language processing, as used for instance in automatic text summarization.

3.3 Group adaptation

With few exceptions AHS perform adaptation to individual users. However, this process can be significantly extended by taking into account actions undertaken by other users and the adaptation has been performed for other users, perhaps with a similar profile or belonging to the same (manually or automatically created) group. Determining the best partitioning of users into groups (that can be also done through collaborative tools adapted to each group features) and finally fitting this within AM is another challenge and subject of ongoing research. Although a few developments have dealt with automatic group formation (considering user features and actions) and adaptive generation of collaborative workplaces (e.g. COL-TANGOW) the main issue here for a new reference model is the existence of (group) models that are not associated with a (single) user, and rules for individual user actions generating updates of these models and of the models influencing adaptation performed for a user (belonging to the group), which is different from known Stereotype AHS. In the ALS[4] project an extension to AHA! system was designed to deal with group formation and adaptation, which allows us to model users belonging to groups as well as groups consisting of users, without the need to create a new and separate way to handle groups versus users.

3.4 Information retrieval and data mining

The behavior of user groups may provide information that can be used to improve the navigation structure of an application. Data mining is a valuable tool in this respect. For example, clustering users into groups based on their navigational patterns can be used to automatically suggest hyperlinks or products to a user or customer, based on the common interests of the members of the group (Yan *et al*. 1996). For an overview of web mining for website personalization (see Eirinaki and Vazirgiannis 2003). Similar research in this direction, providing hints for reorganization of sites, was described in Casteleyn (2005). The application of data mining in AH research has been started mostly in the area of e-Learning (Romero *et al*. 2003, Romero and Ventura 2006), but the need and potential benefits of data mining in the all of AHS areas are obvious. The main consequence of the introduction of data mining in adaptive applications is that the traditional AM based on ECA rules no longer covers all the possible ways in which the needed adaptation can be determined. Whereas ECA rules cover the calculation of the "immediate" adaptation, data mining can potentially be used to capture longer term effects. Ideally the outcome of data mining for adaptation would be the automatic generation or updating of the ECA rules that drive the AE.

3.5 Higher order adaptation

As mentioned in Section 2.1 we are beginning to see applications that not only monitor the user's behavior in order to perform adaptation, but also to decide to adapt the adaptation behavior. Monitoring the user and the adaptation process will allow systems to deduce either directly or indirectly (after data mining) how to refine existing rules or construct new ones. Higher adaptation orders will allow systems to do adaptation to more than one parameter at once, though considering several aspects of adaptation is inherently difficult because they may influence each other.

3.6 Context awareness

On the one hand shifting from Application Model to Context Awareness will help to decouple and make AH systems and applications less integrated with and dependent upon the environment in which they are used. On the other hand, considering a context model will allow the system to be sensitive and adapt in many other ways, rather than following a certain number of fixed adaptation rules. In this respect adaptation to context may also be referred to as a higher order of adaptation, providing monitored results to devise new rules in a particular context.

3.7 Multimedia adaptation

We have already mentioned the possibility of mapping existing content adaptation techniques on multimedia content, which results in a certain level of technique abstraction, irrespectively, to a content type. Future systems should provision this content type independence at every application level: authoring, AE, or presentation generation. This will help to generalize techniques and methods use and broaden application deployment.

4. Conclusion

The coming years will bring more and more use-cases of how AHS can provide adaptation, what techniques will be introduced, and what research areas will be adjacent to AH field and introduce new technologies in its evolution.

However, as a result of investigation even now we can foresee some further developments and research strategies of AH and thus tried to come up with an up-to-date review of AH research for the past 12 years and the resulting requirements for a modular composition of a new AHS reference model that will capture all new trends and adjoining technologies to support users within rich and diverse hyperspaces, bringing a new level of adaptation to the user experience.

The aim of this paper was twofold: first and foremost this paper presented a survey of AH architectures, and defined a new taxonomy of adaptation techniques. Secondly, the paper shows that using the results of this analysis we have obtained many requirements for a new reference model that we will design and that builds on the experience gained with existing models including the Tower Model, the AHAM reference model, the multi-layer LAOS model, and others, and that draws from the many new research ideas that show up in (prototype) adaptive systems.

Acknowledgements

This work has been supported by the NWO GAF Project. We would like to thank the journal editors and anonymous reviewers for providing detailed and constructive comments that helped us to improve this paper.

Notes

[1] In De Bra *et al.* (1992), the model actually did not have a name, but its main construct was the "tower", hence our naming here and in later publications about the model.

[2] In Brusilovsky, the term "adaptive presentation" was used for what we mainly consider to be "content adaptation".

[3] GRAPPLE EU FP7 STREP project—http://www.grapple-project.org/.

[4] Adaptive Learning Spaces project under the Minerva Program.

References

G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions", *IEEE Transactions on Knowledge and Data Engineering*, 17(6), pp. 734–749, 2005.

L. Ardissono, A. Goy and G. Petrone, "A framework for the development of distributed, context-aware adaptive hypermedia applications", in *AH'08 Proceedings of the Fifth International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, W. Nejdl, J. Kay, P. Pu and E. Herder (Eds), Berlin/Heidelberg: Springer, pp. 259–262, 2008.

L. Ardissono, R. Furnari, A. Goy, G. Petrone and M. Segnan, "Context-aware workflow management", in *ICWE'07 Proceedings Of The 7th International Conference On Web Engineering*, L. Baresi, P. Fraternali and G-J. Houben (Eds), Berlin/Heidelberg: Springer, pp. 47–52, 2007.

L. Aroyo, N. Stash, Y. Wang, P. Gorgels and L. Rutledge, "CHIP demonstrator: Semantics-driven recommendations and museum tour generation", in *Proceedings of the 6th International Semantic Web Conference*, K. Aberer, K.-S. Choi, N. Noy, D. Allemang, K-I. Lee, L. Nixon, J. Golbeck, P. Mika, D. Maynard, R. Mizoguchi, G. Schreiber and P. Cudre-Mauroux (Eds), Berlin/Heidelberg: Springer, pp. 879–886, 2007.

L. Aroyo, P. De Bra, G-J. Houben and R. Vdovjak, "Embedding information retrieval in adaptive hypermedia: IR meets AHA!", *Hypermedia*, 10(1), pp. 53–76, 2004.

L. Aroyo, P. De Bra and V. Chepegin, "Semantic web-based adaptive hypermedia", in *Proceedings of the WWW2004 Workshop on Application Design, Development and Implementation Issues in the Semantic Web*, C. Bussler, S. Decker, D. Schwabe and O. Pastor (Eds), Aachen: CEUR-WS.org, 2004.

C. Bailey, W. Hall, D.E. Millard and M.J. Weal, "Towards open adaptive hypermedia", in *AH'02 Proceedings of the Second International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, P.D. Bra, P. Brusilovsky and R. Conejo (Eds), London, UK: Springer-Verlag, pp. 36–46, 2002.

C. Bailey, W. Hall, D.E. Millard and M.J. Weal, "Adaptive hypermedia through contextualised open hypermedia structures", *ACM Transactions on Information Systems (TOIS)*, 25(4), 2007.

M. Balik and I. Jelinek, "General architecture of adaptive and adaptable information systems", in *Proceedings of the Ninth International Conference on Enterprise Information Systems, ICEIS'07: Doctoral Consortium*, J. Cardoso, J. Cordeiro and J. Filipe (Eds), Berlin/Heidelberg: Springer, pp. 29–34, 2007.

H. Barbosa Leon, H. Rego, T. Moreira and F. Garcia Penalvo, "A model for online assessment in adaptive e-learning platform". in *Recent Research Developments In Learning Technologies*, A. Méndez-Vilas, B. González-Pereira, J. Mesa González and J.A. Mesa González (Eds), Badajoz, Spain: FORMATEX, pp. 1–5, 2005.

P. Bellekens, V.D. Sluijs, K., L. Aroyo, G-J. Houben, "Convergence of web and TV broadcast data for adaptive content access and navigation", in *AH'08 Proceedings of the Fifth International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, W. Nejdl, J. Kay, P. Pu and E. Herder (Eds), Berlin/Heidelberg: Springer, pp. 361–365, 2008.

T. Berners-Lee, J. Hendler and O. Lassila, "The Semantic Web", *Scientific American*, May, 2001.

F. Bohnert, I. Zukerman, S. Berkovsky, T. Baldwin and L. Sonenberg, "Using collaborative models to adaptively predict visitor locations in museum", in *AH'08 Proceedings of the Fifth International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, W. Nejdl, J. Kay, P. Pu and E. Herder (Eds), Berlin/Heidelberg: Springer, pp. 42–51, 2008.

P. Brusilovsky, "Methods and techniques of adaptive hypermedia", *User Modeling and User Adapted Interaction*, 6(2–3), pp. 87–129, 1996.

P. Brusilovsky, "Adaptive hypermedia", *User Modeling and User-Adapted Interaction*, 11(1–2), pp. 87–110, 2001.

P. Brusilovsky, "Adaptive navigation support", in *The Adaptive Web*, P. Brusilovsky, A. Kobsa and W. Nejdl (Eds), Berlin/Heidelberg: Springer, pp. 263–290, 2007.

P. Brusilovsky, "Adaptive navigation support for open corpus hypermedia systems", in *AH'08 Proceedings of the Fifth International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, W. Nejdl, J. Kay, P. Pu and E. Herder (Eds), Berlin/Heidelberg: Springer, pp. 6–8, 2008.

P. Brusilovsky, E. Schwarz and G. Weber, "ELM-ART: An intelligent tutoring system on World Wide Web", in *ITS'96: Proceedings of Third International Conference on Intelligent Tutoring Systems*, C. Frasson, G. Gauthier and A. Lesgold (Eds), London, UK: Springer-Verlag, pp. 261–269, 1996.

P. Brusilovsky, J. Eklund and E. Schwarz, "Web-based education for all: A tool for developing adaptive courseware", *Computer Networks*, 30(1–7), pp. 291–300, 1998.

P. Brusilovsky and N. Henze, "Open corpus adaptive educational hypermedia", in *The Adaptive Web*, P. Brusilovsky, A. Kobsa and W. Nejdl (Eds), Berlin/Heidelberg: Springer, pp. 671–696, 2007.

F. Carmagnola, F. Cena, C. Gena and I. Torre, "A semantic framework for adaptive web-based systems", in *Proceedings of SWAP 2005, The 2nd Italian Semantic Web Workshop*, Paolo Bouquet and Giovanni Tummarello (Eds), CEVR Workshop Proceedings, 14–16 December 2005, Trento, Italy, 2005.

R.M. Carro, A. Ortigosa, E. Martín and J. Schlichter, "Dynamic generation of adaptive web-based collaborative courses", in *Groupware: Design, Implementation and Use*, D. Decouchant and J. Favela (Eds), London: Springer-Verlag, pp. 191–198, 2003.

R.M. Carro, E. Pulido and P. Rodríguez, "TANGOW: Task-based adaptive learner guidance on the web", in *Proceedings of the Second Workshop on Adaptive Systems and User Modeling on the Web. TUE Computting Science Report 99–07*, P. Brusilovsky and P. De Bra (Eds), pp. 49–57, 1999.

S. Casteleyn, *Designer specified self re-organizing websites*, PhD Dissertation, Vrije Universiteit Brussels, 2005.

D. Challis, "Committing to quality learning through adaptive online assessment", *Assessment and Evaluation in Higher Education*, 30(5), pp. 519–527, 2005.

Z. Chedrawy and S.S.R. Abidi, "An adaptive personalized recommendation strategy featuring context sensitive content adaptation", in *AH'06 Proceedings of the Fourth International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, V.P. Wade, H. Ashman and B. Smyth (Eds), Berlin/Heidelberg: Springer, pp. 61–70, 2006.

O. Conlan, C. Hockemeyer, V. Wade and D. Albert, "Metadata driven approaches to facilitate adaptivity in personalized eLearning systems", *Journal of the Japanese Society for Information and Systems in Education*, 1, pp. 38–45, 2002a.

O. Conlan, V. Wade, C. Bruen and M. Gargan, "Multi-model, metadata driven approach to adaptive hypermedia services for personalized eLearning", in *AH'02 Proceedings of the Second International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, P.D. Bra, P. Brusilovsky and R. Conejo (Eds), London: Springer-Verlag, pp. 100–111, 2002b.

A. Cristea and A. De Mooij, "LAOS: Layered WWW AHS authoring model and their corresponding algebraic operators", in *WWW'03 Proceedings of World Wide Web International Conference (Alternate Paper Tracks)*, New York, NY: ACM, 2003a.

A. Cristea and A. De Mooij, "Evaluation of MOT, an AHS authoring tool", in *CATE 2003: International Conference on Computers and Advanced Technology in Education*, V. Uskov (Ed), Calgary, Alta., Canada: ACTA-Press, pp. 241–246, 2003b.

A. Cristea and L. Calvi, "The three layers of adaptation granularity", in *UM'03: 9th International Conference on User Modeling*, J.G. Carbonell and J. Siekmann (Eds), Berlin/Heidelberg: Springer, p. 4–14, 2003.

A. Cristea and M. Verschoor, "The LAG grammar for authoring the adaptive web", in *Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'04)*, Washington, DC: IEEE Computer Society, p. 382–386, 2004.

P. De Bra, D. Smits and N. Stash, "The design of AHA!", in *Hypertext'06: Proceedings of the 17th Conference on Hypertext and Hypermedia*, New York, NY: ACM, pp. 133–134, 2006.

P. De Bra, G-J. Houben and H. Wu, "AHAM: A dexter-based reference model for adaptive hypermedia", in *Hypertext'99: Proceedings of the Tenth ACM Conference on Hypertext and Hypermedia: Returning to our Diverse Roots: Returning to our Diverse Roots*, J. Westbomke, U.K. Wiil, J.J. Leggett, K. Tochtermann and J.M. Haake (Eds), New York, NY: ACM, pp. 147–156, 1999.

P. De Bra, G-J. Houben and Y. Kornatzky, "An extensible data model for hyperdocuments", in *Proceedings of the ACM Conference on Hypertext,* New York, NY: ACM, pp. 222–231, 1992.

P. De Bra, H. Wu and G-J. Houben, "Supporting user adaptation in adaptive hypermedia applications", in *Proceedings of InfWet'00 Conference*, 2000.

P. De Bra and L. Calvi, "AHA! an open adaptive hypermedia architecture", *Hypermedia*, 4(1), pp. 115–139, 1998.

P. De Bra, M. Pechenizkiy, K. van der Sluijs, D. Smits, "GRAPPLE: Integrating adaptive learning into learning management system", in *Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications 2008*, Chesapeake, VA: AACE, pp. 5183–5188, 2008.

P. De Bra and W. Nejdl, *Adaptive Hypermedia and Adaptive Web-Based Systems, Third International Conference, AH 2004, Eindhoven, The Netherlands, August 23–26, 2004, Proceedings*, Berlin/Heidelberg: Springer, 2004.

S. Dietze, A. Gugliotta and J. Domingue, "A semantic web service oriented framework for adaptive learning environment", in *Proceedings of 4th European Semantic Web Conference, ESWC 2007*, Berlin/Heidelberg: Springer, pp. 701–715, 2007.

M. Eirinaki and M. Vazirgiannis, "Web mining for web personalization", *ACM Transactions on Internet Technology (TOIT)*, 3(1), pp. 1–27, 2003.

R. Farzan and P. Brusilovsky, "Social navigation support in a course recommendation system", in *AH'06 Proceedings of the Fourth International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, V.P. Wade, H. Ashman and B. Smyth (Eds), Berlin/Heidelberg: Springer, pp. 91–100, 2006.

M. Gorle, J. Ohene-Djan, C. Bailey, G. Wills and H. Davis, "Is it possible to devise a unifying model of adaptive hypermedia and is one necessary?", in *AH2003: Proceedings of the Workshop on Adaptive Hypermedia and Adaptive Web-Based Systems, Fourteenth Conference on Hypertext and Hypermedia*, P. De Bra and M. Schraefel (Eds), New York, NY: ACM, pp. 167–183.

F. Halasz and M. Schwartz, "The dexter reference model", in *Proceedings of the NIST Hypertext Standardization Workshop*, J. Moline (Ed), pp. 95–133, 1990.

F. Halasz and M. Schwartz, "The dexter hypertext reference model", *Communications of the ACM*, 37(2), pp. 30–39, 1994.

F. Hanisch, M. Muckenhaupt, F. Kurfess and W. Strasser, "The AHES taxonomy: Extending adaptive hypermedia to software components", in *AH'06 Proceedings of the Fourth International Conference on*

Adaptive Hypermedia and Adaptive Web-Based Systems, V.P. Wade, H. Ashman and B. Smyth (Eds), Berlin/Heidelberg: Springer, pp. 342–345, 2006.

M. Hendrix and A. Cristea, "Meta-levels of adaptation in education", in CATE'08: *Proceedings of 11th IASTED International Conference on Computers and Advanced Technology in Education*, V. Uskov (Ed.), IASTED, 2008.

N. Henze, *Adaptive hyperbooks: Adaptation for project-based learning resources.* PhD Dissertation, University of Hannover, 2000.

N. Henze and W. Nejdl, "A logical characterization of adaptive educational hypermedia", *Hypermedia*, 10(1), pp. 77–113, 2004.

Hockemeyer, T. Held and D. Albert, "RATH-A relational adaptive tutoring hypertext WWW-Environment based on knowledge space theory", in *CALISCE'98: Proceedings of the Fourth International Conference on Computer Aided Learning in Science and Engineering*, C. Alvegård (Ed), Sydney: UniServe Science, pp. 417–423, 1998.

Kobsa, "Generic user modeling systems", *User Modeling and User Adapted Interaction*, 11(1–2), pp. 49–63, 2001.

N. Koch, *Software engineering for adaptive hypermedia systems.* PhD Dissertation, Ludwig-Maximilians University of Munich, 2001.

N. Koch and M. Wirsing, "The Munich reference model for adaptive hypermedia applications", in *AH'02 Proceedings of the Second International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, P.D. Bra, P. Brusilovsky and R. Conejo (Eds), London: Springer-Verlag, pp. 213–222, 2002.

M. Kravčík and D. Gaëevič, "Adaptive hypermedia for the semantic web", in *Proceedings of the Joint International Workshop on Adaptivity, Personalization and the Semantic Web*, New York, NY: ACM, pp. 3–10, 2006.

O. Lassila and R.R. Swick, 1999. Resource description framework (rdf) model and syntax specification. W3C Recommendation 22 February 1999. W3C. Available online at: http://www.w3.org/TR/1999/REC-rdf-syntax-19990222/ (accessed 10 November 2008).

T. Lutkenhouse, M.L. Nelson and J. Bollen, "Distributed, real-time computation of community preferences", in *Hypertext'05: Proceedings of the 16th ACM Conference on Hypertext and Hypermedia*, New York, NY: ACM, pp. 88–97, 2005.

E. Martín, R.M. Carro and P. Rodríguez, "A mechanism to support context-based adaptation in M-learning", in *Innovative Approaches for Learning and Knowledge Sharing*, W. Nejdl and K. Tochtermann (Eds), Berlin/Heidelberg: Springer, pp. 302–315, 2006.

L. Mei and S. Easterbrook, "Evaluating user-centric adaptation with goal models", in *Proceedings of the 29th International Conference on Software Engineering Workshops*, Washington, DC: IEEE Computer Society, p. 192, 2007.

T. Morita, Y. Shigeta, N. Sugiura, N. Fukuta, N. Izumi and T. Yamaguchi, "DODDLE-OWL: A domain ontology construction tool with OWL", in *ASWC'06: Proceedings of the First Asian Semantic Web Conference*, R. Mizoguchi, Z. Shi and F. Giunchiglia (Eds), Berlin/Heidelberg: Springer, pp. 537–551, 2006.

O'Connor and V. Wade, "Informing context to support adaptive services", in *AH'06 Proceedings of the Fourth International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, V.P. Wade, H. Ashman and B. Smyth (Eds), Berlin/Heidelberg: Springer, pp. 366–369, 2006.

J. Ohene-Djan, *A formal approach to personalisable adaptive hyperlink-based interaction*. PhD Dissertation, University of London, 2000.

J. Ohene-Djan and A. Fernandes, "Modeling personalisable hypermedia: The goldsmith model", *New Review Hypermedia Multimedia*, 8(1), pp. 99–137, 2002.

Paris, M. Wu, K. Vander linden, M. Post and S. Lu, "Myriad: An architecture for contextualized information retrieval and delivery", in *AH'04 Proceedings of the Third International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, P.D. Bra and W. Nejdl (Eds), Berlin/Heidelberg: Springer, pp. 205–214, 2004.

R. Priedhorsky, J. Chen, S. Lam, K. Panciera, L. Terveen and J. Riedl, "Creating, destroying, and restoring value in wikipedia, in *GROUP'07: Proceedings of the 2007 international ACM Conference on Supporting Group Work*, New York, NY: ACM, pp. 259–268, 2007.

Romero and S. Ventura, *Data Mining in E-Learning*, Wessex, UK: WIT Press, 2006.

Romero, S. Ventura, P. De Bra and C. De Castro, "Discovering prediction rules in AHA! Courses", in *UM'03: 9th International Conference on User Modeling*, J.G. Carbonell and J. Siekmann (Eds), Berlin/Heidelberg: Springer, pp. 35–44, 2003.

L. Rutledge, N. Stash, Y. Wang and L. Aroyo, "Accuracy in rating and recommending item features", in *AH'08 Proceedings of the Fifth International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, W. Nejdl, J. Kay, P. Pu and E. Herder (Eds), Berlin/Heidelberg: Springer, pp. 163–172, 2008.

M. Sprecht, M. Kravcik, R. Klemke, L. Pesin and R. Huttenhain, "Adaptive learning environment for teaching and learning in WINDS", in *AH'02 Proceedings of the Second International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, P.D. Bra, P. Brusilovsky and R. Conejo (Eds), London, UK: Springer-Verlag, pp. 572–575, 2002.

N. Stash, *Incorporating cognitive/learning styles in a general-purpose adaptive hypermedia system*. PhD Dissertation, Eindhoven University of Technology, 2007.

N. Stash, L. Aroyo, Y. Wang, L. Rutledge and P. Gorgels, "Semantics-driven recommendations in cross-media museum applications", in *AH'08 Proceedings of the Fifth International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, W. Nejdl, J. Kay, P. Pu and E. Herder (Eds), Berlin/Heidelberg: Springer, 2008.

S. Stober and A. Nurnberger, "Context-based navigational support in hypermedia", in *AH'06 Proceedings of the Fourth International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, V.P. Wade, H. Ashman and B. Smyth (Eds), Berlin/Heidelberg: Springer, pp. 328–332, 2006.

T. Theophanis and M.C. Schraefel, "Adaptive presentation supporting focus and context", in Proceedings of AH2003: *Workshop on Adaptive Hypermedia and Adaptive Web-Based Systems, 14th Conference on Hypertext and Hypermedia*, P. De Bra (Ed), 2003.

N. Tintarev and J. Masthoff, "The effectiveness of personalized movie explanations: An experiment using commercial meta-data", in *AH'08 Proceedings of the Fifth International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, W. Nejdl, J. Kay, P. Pu and E. Herder (Eds), Berlin/Heidelberg: Springer, pp. 204–213, 2008.

Triantafillou, A. Pomportis, S. Demetriadis and E. Georgiadou, "The value of adaptivity based on cognitive style: An empirical study", *British Journal of Educational Technology*, 35(1), pp. 95–106, 2004.

Tsymbal, M. Pechenizkiy, P. Cunningham and S. Puuronen, "Dynamic integration of classifiers for handling concept drift", *Information Fusion*, 9(1), pp. 56–68, 2008.

W. van der Aalst, P. De Bra, G-J. Houben and Y. Kornatzky, "Browsing semantics in the "Tower" model", *Computer Science Notes*, 93(47), pp. 2–19, 1993.

R. Vdovjak and G-J. Houben, "Providing the semantic layer for WIS design", in *CAiSE'02: 14th International Conference on Advanced Information System Engineering*, G. Goos, J. Hartmanis, and J. van Leeuwen (Eds), Berlin/Heidelberg: Springer, pp. 584–599, 2002.

V. Wade, H. Ashman and B. Smyth, "Adaptive hypermedia and adaptive web-based systems", in *4th International Conference, AH 2006, Dublin, Ireland, June 21–23, Proceedings.* Berlin/Heidelberg: Springer, 2006.

H. Wu, *A reference architecture for adaptive hypermedia application*. PhD Dissertation, Eindhoven University of Technology, 2002.

Y. Xie, A. Ravichandran, H. Haddad and K. Jayasimha, "Capturing concepts and detecting concept-drift from potential unbounded, ever-evolving and high dimensional data streams", *Studies in Computational Intelligence (SCI)*, 118, pp. 485–499, 2008.

T-W. Yan, M. Jacobsen, H. Garcia-Molina and U. Dayal, "From user access patterns to dynamic hypertext linking", *Computer Networks and ISDN Systems*, 28(7–11), pp. 1007–1014, 1996.