Improving Adaptive Hypermedia by Adding Semantics

Anton ANDREJKO

Slovak University of Technology
Faculty of Informatics and Information Technologies
Ilkovičova 3, 842 16 Bratislava, Slovak republic
andrejko@fiit.stuba.sk

Abstract. Recently the Web and its services have changed the way of thinking about them; hereby expectations have changed, too. While in recent times users have been satisfied with presentation of generic information, nowadays they are asking for personalized information, which are adapting to them. This kind of presentation is not sufficient in the information acquiring from databases on the Web. It is needed to append metadata to this information. Metadata express semantics and makes the information easier for computer processing. This paper presents models for adaptive and semantic Web and shows how to improve adaptive presentation with semantics.

1 Introduction

On the Web is the huge amount of the information, which are distributed and structured in various levels. The user can get lost very easy in this environment or he/she spends too much time browsing among information with not very essential value. For this reason is needed to append a new element to hypermedia systems – adaptation.

By adaptive hypermedia systems we mean all hypertext and hypermedia systems, which reflect some features of the user in the user model and apply this model to adapt various visible aspects of the system to the user. In other words, the system should satisfy three criteria: it should be a hypertext or hypermedia system, it should have a user model, and it should be able to adapt the hypermedia using this model [4].

Probably the most used way for accessing the information are catalogs and full-text search engines. These only generate a result from database, which is based on
query given by the user. The search engines do not understand the presented content and do not take care of the language flexion and context in which was the query given by user. Ontologies are the way how to overcome differences between queries and search results. Moreover these results could put more exactly by using adaptive hypermedia techniques described in [4]. In next sections we describe models used in adaptive and semantic Web. The understanding of the models is very important for next research. The aim of this paper is to propose a new adaptive model with added semantics. We try to find an appropriate place in adaptive model for including semantics.

2 Modeling adaptive hypermedia systems

This section describes the evolution from the hypermedia models to the adaptive hypermedia. Models have been developed by designers' in various independent Web systems. In these systems are many similar architecture aspects but they using the different terminology. For further research we need common model and terminology for hypertext field.

2.1 Hypertext Abstract Machine

The Hypertext Abstract Machine (HAM) was the first try to make a common system usable for various hypertext applications. HAM is hierarchical model based on five objects [5]: graphs, contexts, nodes, links and attributes. A graph is the higher-level object. A graph contains information about a topic and consists from one or more contexts. Each context has one parent context and zero or more child contexts. The context consists of nodes and links. Nodes contain arbitrary data. These data are conducted by links. A link defines relationship between source node and destination node in both directions. Attributes (strings, numbers, and user’s types) can be attached to contexts, nodes and links. Attributes give semantics to HAM objects.

2.2 Dexter hypertext reference model

The Dexter reference model is the result of two workshops. The model is an attempt to capture, further develop, and formalize the results of these discussions. Also it tries to find common terminology for hypertext field. Dexter model divides hypertext systems into three layers: run-time, storage and within-component [8]. The run-time layer describes mechanism the user’s interaction with hypertext. The within-component layer covers the content and structures within hypertext nodes. The range of the content or structure is not restricted. Dexter model has become the base for developing others hypertext systems therefore it will be explained in more details.

The main focus is on storage level, which models nodes/links of the network structure that are the base for hypertext. The storage layer describes “database” that is composed of hierarchy of data-contained components interacted by relational links. In different hypertext systems can be used for nodes terms as cards, frames, documents or
articles. Dexter model generalizes the term node. Instead of the node is used the term component. Component can consist from text, graphics, pictures or animation that makes the main content. This layer does not distinguish between text component and graphical component. The layer distinguishes atomic components and composite component. Atomic components are usually nodes in hypertext systems and their structure is the concern of the within-component layer. Composite components are constructed from atomic components.

![Fig. 1. The Dexter Reference Model.](image)

As is shown in the figure 1, presentation specification and anchoring are interfaces between layers these layers. Presentation specifications are the mechanism by which information about how will be component presented encoded to hypertext network at storage layer. Anchoring is a mechanism for addressing item within component.

### 2.3 Adaptive Hypermedia Application Model

In the adaptive hypermedia systems (AHS) are watched preferences and knowledge about domain. These records are stored in the user model that helps to direct user to the relevant information. AHS do that by dynamic adaptation links and content. AHS are adaptive. The user can adjust some preferences explicit.

In Dexter model component covers both nodes and links. In Adaptive Hypermedia Application Model (AHAM) [6] the central term are concept and concept relationships. AHAM extends Dexter model with adaptation based on user model. In this model (see figure 2), Domain model corresponds (roughly) to the storage layer in Dexter model. The domain model comprises of the concepts. The user model describes of which information about the user AHAM keeps a permanent record. The teaching model consists of the rules which define how the domain model and user model are combined to perform the adaptation.
2.4 Munich Reference Model

This model is based on Dexter model and formally described by UML (Unified Modeling Language). The Munich model [10] keeps three-layers structure and appends rule-based adaptation for user aspects. The run-time layer is responsible for interaction with the user.

The adaptation meta-model is appended within storage layer (see figure 3). The adaptation meta-model consists of a set of rules that implement the adaptive
functionality, i.e. personalization of the application. The domain meta-model manages the basic network structure of the hypermedia system. A part of the domain meta-model is the navigation model. This model describes possible navigation through information space determined by the domain meta-model [7]. The user meta-model manages a set of users represented by their attributes. First group are domain dependent attributes (knowledge) and second one are domain independent attributes (preferences). Mentioned models described by UML are more in details in [3, 7].

3 Adding the semantics

Adaptive hypermedia systems are able to adapt to the user or environment but many times are unsatisfactory for users’ expectations although they offer adapted content and/or presentation. Users expect qualitative information with small effort. In many cases information that are presented to the user and that had been adapted claim user’s effort to be useful. Computers do not “understand” the presented content, therefore is needed to add meta-data to the content for better computer-processing. The meta-data describe only the meaning of the content. The most time-consuming user’s activity is looking for relationships between those contents.

The problem may cause designers who can use different meta-data for describing the same content. Shared understanding is necessary to overcome differences in terminology. Ontologies provide a shared understanding of a domain [1]. Ontologies are useful for the organization, navigation and improving the accuracy of the Web search.

In present-day AHS is not described semantics of the content that is stored in within-component layer. In this paper we describe an approach how to add semantics to adaptive hypermedia. To get this goal it is necessary to briefly describe semantic Web structure. Project Hera\(^1\) uses a model-driven methodology for developing semantic web application and divides them into three layers [2]: semantic, application and presentation (see Fig. 4).

\[\text{Fig. 4. The Semantic model in Hera.}\]

\(^{1}\) Hera, http://wwwis.win.tue.nl/~hera/
The presentation layer specifies the layout of the presented content, application layer specifies the navigation view over the conceptual data and consists from slices that specify the structure of navigation nodes. The core of the semantic layer is conceptual model describing the semantics of the data content. This model is expressed in terms of RDFS.

The two upper layers can be found in AHS. If we compare Dexter model as a headstone for AHS to the semantic model, the run-time layer would include the presentation layer and slices from the application layer liken to the components in the storage layer. But models mentioned in the section 2 miss the semantics that is in semantic layer.

Before we add the semantic layer into the adaptive model, we should note that Hera uses personalized presentation by user preferences in application model. To achieve adapting in Hera is used conditional inclusion of slices (fragments) and navigation (links) hiding [9]. In adaptive hypermedia are more techniques for adapting content and navigation [4]. Therefore we prefer adding semantics to the AHS to adding adaptive aspects to the semantic model. This seems to be easier way to fulfill the goal – personalized presentation with the use of semantics.

In our approach we insert the semantic layer between storage and within-component layer (see figure 5). This layer specifies the data content. It also defines the integration process that gathers the data from different sources.

\[
\begin{array}{|c|}
\hline
\text{Run-time layer} \\
\hline
\text{Storage layer} \\
\hline
\text{Semantic layer} \\
\hline
\text{Within-Component layer} \\
\hline
\end{array}
\]

Fig. 5. The Dexter Model with the semantic layer.

The layer creates an interface between layers and according to the user query (for instance in Web search) by using ontology takes into account semantics of the content and offers that content sensitive result from within-component layer to the user. The layer expresses the semantics of the content. For describing the semantics we can use one from ontology languages for instance RDF, RDF Schema or OWL. Any of them seems to be useful. The selection of the appropriate language and inner structure design is the goal for the following research.

4 Conclusions

In this paper we have described hypermedia models and adaptive hypermedia models. We have focused on Dexter reference model that is the base for other AHS. In our research we try to add semantics into AHS and improve the current systems. The semantics layer proposed in this paper connects personalized adaptive hypermedia with
Semantic Web initiative. Semantic layer improve accuracy in Web search and is useful in navigation of Web sites. In following research we are planning deeply describe inner structure of the semantic layer. We are also planning to experiment our approach in prototype.

Acknowledgement: This work has been partially supported by the Grant Agency of Slovak Republic grant No. VG1/ 0162/03.

References