The next big thing: adaptive web-based systems

De Bra, P.M.E.; Aroyo, L.M.; Chepegin, V.

Published in:
Journal of Digital Information

Published: 01/01/2004

Document Version
Publisher’s PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:
- A submitted manuscript is the author’s version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher’s website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

Citation for published version (APA):

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal?

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.
The Next Big Thing: Adaptive Web-Based Systems

Paul De Bra, Lora Aroyo and Vadim Chepegin
Department of Computer Science, Eindhoven University of Technology, Eindhoven, The Netherlands
Email: {debra, laroyo, vchepegi}@win.tue.nl

Abstract

At the ACM Hypertext Conference a panel discussed "The Next Big Thing Inc." in the area of hypermedia. The Web has been the "Big Thing" during the past 10 years, but its success has also brought a host of problems. In this paper we argue in favor of adaptation but at the same time we made two predictions for the future (within three to five years from publication):

1. the emergence of a mass market for hyperbass,
2. the integration of hyperbass and other computer facilities.

It has taken a bit longer, but other than that the predictions were spot on. To some extent World Wide Web technology has been used in a more effective way. The Web itself is the best illustration of the mass market emerge true. With over 3 billion pages available online as well as available online and sometimes even only online. Traditional help systems, like Microsoft Windows help, also have hyperbass functionality has been effectively integrated with other computer facilities.

Hypertext, and the Web in particular, offers three ways to find information:

- When you know the precise location of the information you want, you type a URL in the browser's location field, and you have located the information. A bookmark list can help to remember the location of information you found before.
- When you have a good description of what it is you wish to find, the modern search engines are very good at finding the location of the information. For instance, hoping to find the sorts of things you are interested in at that time. When you start browsing the search results for "Paul De Bra", the abov appear to be among the first 500 hits (out of over 2000). Browsing may provide access to all the information (that is linked) available about topics you are interested in. The core problem in finding the information you want, in all the above cases, is describing what you want. Results from search engine most search requests are too short and unspecific to yield good results. Once a Web site with interesting information is found, it is often interesting pages only, because the site can only be navigated using its predefined link structure, independently of the search request...
The community of user modelling and adaptive hypermedia offers solutions for this problem: using information gathered about the user, change the information content and link structure on-the-fly. User modeling captures the mental state of the user, and thus allows to determine precisely what the user is looking for. To support the design of this user model-based adaptive hypermedia systems (AHSs) to show how an AHS processes a user’s request, using its architecture, and for a more dynamic and sharable framework for automated personalization and adaptation. In this way, current adaptation can be extended with powerful reasoning at the level of standardized concept schemes, as opposed to the traditional hand-crafting by author relations, thus allowing adaptation generation in open information spaces on a higher schema level.

This paper first briefly recalls and summarises the overall architecture of (Web-based) adaptive hypermedia systems (AHSs) to show how architecture to enable different AHSs to work together at different levels: the conceptual structure, the user model, and the adapta- tion system. The former category of systems is called adaptable, the latter adaptive. A lot of research has been performed in the field of adaptive hypermedia on coming up with new ways of adaptation, to offer the user better, more suitable, more acceptable and more visible guide- ance. To become “The Next Big Thing”, adaptive hypermedia systems need to open their architecture to allow collaboration between sites to be currently possible. The latest developments within the scope of the Semantic Web are leading adaptive hypermedia towards the Adaptable Adaptation, architecture, and for a more dynamic and sharable framework for automated personalization and adaptation. In this way, current adaptation can be extended with powerful reasoning at the level of standardized concept schemes, as opposed to the traditional hand-crafting by author relations, thus allowing adaptation generation in open information spaces on a higher schema level.

2 Adaptive hypermedia architecture

On the Web personalization is being realized in two ways:

- Some Web sites require users to register and provide information about their interests. The site takes the interests into account when presenting content to the user. The user’s profile is only updated when the user revisits the site to make changes.
- Some Web sites require users to register only so they can be identified. Identification is also possible through cookies, in which information about the user’s behavior on the site is stored. The site monitors the user’s browsing behavior (and in particular the pages that are visited) in order to adapt the presentation to the user’s interests and knowledge. The presentation is updated according to the user model and the model is updated when the user revisits it to make changes.

The former category of systems is called adaptable, the latter adaptive. A lot of research has been performed in the field of adaptive hypermedia on coming up with new ways of adaptation, to offer the user better, more suitable, more acceptable and more visible guidance. To become “The Next Big Thing”, adaptive hypermedia systems need to open their architecture to allow collaboration between sites to be currently possible. The latest developments within the scope of the Semantic Web are leading adaptive hypermedia towards the Adaptable Adaptation, architecture, and for a more dynamic and sharable framework for automated personalization and adaptation. In this way, current adaptation can be extended with powerful reasoning at the level of standardized concept schemes, as opposed to the traditional hand-crafting by author relations, thus allowing adaptation generation in open information spaces on a higher schema level.

This paper first briefly recalls and summarises the overall architecture of (Web-based) adaptive hypermedia systems (AHSs) to show how architecture to enable different AHSs to work together at different levels: the conceptual structure, the user model, and the adaptations for a more dynamic and sharable framework for automated personalization and adaptation. In this way, current adaptation can be extended with powerful reasoning at the level of standardized concept schemes, as opposed to the traditional hand-crafting by author relations, thus allowing adaptation generation in open information spaces on a higher schema level.

2 Adaptive hypermedia architecture

On the Web personalization is being realized in two ways:

- Some Web sites require users to register and provide information about their interests. The site takes the interests into account when presenting content to the user. The user’s profile is only updated when the user revisits the site to make changes.
- Some Web sites require users to register only so they can be identified. Identification is also possible through cookies, in which information about the user’s behavior on the site is stored. The site monitors the user’s browsing behavior (and in particular the pages that are visited) in order to adapt the presentation to the user’s interests and knowledge. The presentation is updated according to the user model and the model is updated when the user revisits it to make changes.

The former category of systems is called adaptable, the latter adaptive. A lot of research has been performed in the field of adaptive hypermedia on coming up with new ways of adaptation, to offer the user better, more suitable, more acceptable and more visible guidance. To become “The Next Big Thing”, adaptive hypermedia systems need to open their architecture to allow collaboration between sites to be currently possible. The latest developments within the scope of the Semantic Web are leading adaptive hypermedia towards the Adaptable Adaptation, architecture, and for a more dynamic and sharable framework for automated personalization and adaptation. In this way, current adaptation can be extended with powerful reasoning at the level of standardized concept schemes, as opposed to the traditional hand-crafting by author relations, thus allowing adaptation generation in open information spaces on a higher schema level.

This paper first briefly recalls and summarises the overall architecture of (Web-based) adaptive hypermedia systems (AHSs) to show how architecture to enable different AHSs to work together at different levels: the conceptual structure, the user model, and the adaptations for a more dynamic and sharable framework for automated personalization and adaptation. In this way, current adaptation can be extended with powerful reasoning at the level of standardized concept schemes, as opposed to the traditional hand-crafting by author relations, thus allowing adaptation generation in open information spaces on a higher schema level.

The links that emanate from the page can be manipulated. Links to pages that are considered not suitable can be hidden, or even removed. The link destination can be changed as well. The information content of the page can be altered, e.g. by conditionally including or hiding fragments. When a pre-defined query is matched, a new fragment can be inserted, or when an interest value is high, additional details can be shown. Another possible adaptation is to change the information overlay model. The overlay model can be extended with powerful reasoning at the level of standardized concept schemes, as opposed to the traditional hand-crafting by author relations, thus allowing adaptation generation in open information spaces on a higher schema level.
Figure 1 shows that the three submodels (DM, UM, AM) all reside on a central Web server, together with the "local pages" (content fra handles page requests. Pages can come from external servers, but the corresponding concepts must exist in the local domain model, updates to the user model. That user model is used to perform the adaptation.

The centralized architecture of AHA! (and several other systems) has the advantage that the DM, AM and UM can work closely together retrieve the DM and AM when the server starts, and the UM when the user logs on, and to keep all that information cached in memory UM to disk (or database). For adaptive systems to work together, as suggested in the introduction, a new, decentralized and modular presented in Figure 2. The issue of optimization is transferred towards the bridging protocols and the information request coordinatio that architecture. We recognize that performance remains very important as a key to the success of adaptive information services in performant as non-adaptive services.

Apart from the issue of collaboration between adaptive systems, the centralized architecture has some other drawbacks: the semantic domain is usually stored inside a local domain and adaptation model. Relationships between concepts are "hidden" inside adaptation r graphical interface - i.e. Graph Author in AHA!, Figure 1 - for the authors to define the concepts and relationships and make high-leve difficult to "export" the semantic information contained in the DM/AM combination to applications that deal with concepts and relati difficult to "import" semantic information, e.g. an ontology, into an AHA! application. Also, the user model cannot be accessed by ext by applications running on the same AHA! server. Knowledge gained by the user through an online course cannot be used by an online department in order to initialize the user model there.

The next section looks at various research efforts attempting to provide solutions for these problems while facilitating adaptation to\ and distributed open information space. Methods are proposed to improve the quality, consistency and linking of Web documents whi authors when creating the adaptation. Reflecting on this work we propose a new, modular architecture of an open Web-based adapt above difficulties.

3 Modular adaptive hypermedia

While ubiquitous computing was still a far fetched vision when introduced by Weiser at the Computer Science Lab at Xerox PARC in Y1 and a promising part of our everyday lives. After the age of mainframes (one machine - many people), and after the boom of personal person), now we are moving towards "the age of calm technology", as described by Weiser and Brown (1996) (one person - many comp refers to it: the "Third Paradigm" of computing.

The essence of ubiquitous computing is to offer to users systems with "invisible" (ambient) intelligence, which will be able to know ab desired service, content and presentation, without intrusion and unnecessary human computer interaction. Adaptation across ac and realized with the notion of adaptive Web-based systems, defines the goal today: to provide adaptation within software environm interact simultaneously with various applications. For this we need open and modularized architectures, which are able to interact, e components. A fundamental issue in such architectures relates to the coordination, handling and control of all the components (servic biggest challenges in this context is the sharing, synchronization and interpretation of the user model among the different applicatio within each system will be permanently evaluated and more detailed, and richer user models will be achieved in order to allow for or personalization of the content. This paper presents our approach for achieving this openness and modularity of the adaptive hyperme different AHSs to work together at different levels (e.g. conceptual, user model, and adaptation) we see the need for four main asp:

- supporting a strict separation of domain model, application model, adaptation model and user model, to ensure good modulariz
- maintaining generic sharable (dynamic) models, such as the user model, to serve as a communication point for different AHSs;
The basic idea is that by augmenting encapsulated system modules with rich formal descriptions of their competence, we can further aspects of the system management. The role of sharable models such as the dynamic user model is crucial for enabling inter-system sharability and reusability of user-related modules. Finally, by applying open Web standards we can enable interoperability and wide hypermedia systems.

Currently research in the area of the Semantic Web, originating primarily from the knowledge engineering and artificial intelligence field of ontologies and Web services, provides a number of standards and accompanying solutions which can be used to achieve the above. We have the notion of ontology, which plays a role in facilitating the sharing of meaning and semantics of information between number and representational formats that have been proposed as W3C standards for ontology and metadata representation. The most current existing Web standards (e.g., XML, RDF and RDF(S) and add the primitives of description logic as powerful means for reasoning services. Initiatives to illustrate this is the SHOE metadata annotation of Web content (Heflin et al. 1999). The idea was further elaborated in et al. 2001) with use of ontology concept instances, considering evolving ontologies and offering annotation in a semi-automated way.

Several other annotation tools are known to the research community, e.g. the Amaya Web editor for RDF-based mark-up of resources points out the advantages of a centralized server, and its unapplicability in "open" environments. It exemplifies a good scenario for co-"closed" content spaces, and illustrates its difficult implementation in an "open" information space. Other examples of annotation system project (Mulholland et al. 2000), the CREAM-based Ont-O-Mat/Annotizer (Handschuh et al. 2001), WeM (Vargas-Vera et al. 2002), Lett... As the annotation is only part of the content authoring, another rather labor-intensive part is the process of linking the annotated content to a Hypermedia Services Environment (COHSE) developed by Carr et al. (2001) introduces an ontological reasoning service over domain cc combination with a Web-based open hypermedia link service: This enables documents to be linked via metadata describing their context.

Another recent project inspired by COHSE and applying ontologies and semantic services for the automation of semantic content (Carr et al. 2003). This project facilitates various interpretation views on the same content with no prior mark-up, but by means of adding an so-called "virtual" mark-up. On top of this semantic layer Magpie deploys semantic services provided to the user as a physically independent resource.

Thus, the next step in the process of opening up AHS architectures is applying a Web services perspective on the system components. Above-mentioned semantics and offer means for flexible composition of services (system components) through automatic selection, verification of service properties, and execution.

Another relevant approach for describing the role of Web services in system architecture is the Web Service Modeling Framework (WSMF). That research shows that Web services appear to be a useful solution for achieving modularity. We can achieve reasonable autonomy of main aspects of Web services (e.g. Web service location, composition and mediation) by extending them with rich formal description standardized languages such as RDF or OWL). In this way we can allow adaptive Web-based systems to reason about the functionality services, to locate the best ones for solving a particular problem, and automatically to compose the relevant Web services for dynamic needs.

An interesting approach that could serve as the basis for a successful application of the Web service perspective on AHS architecture framework like the Internet Reasoning Service (IRS-II) introduced by Hendtlass et al. (2003). They show how we can support the public administration execution of heterogeneous semantic-rich Web services. The service uses UPML (Unified Problem-solving Method description Language) knowledge-based systems by defining how we can build elementary components and how these components can be integrated into an IRS-II approach supports capability-driven service invocation (e.g. find a service that can solve problem X) by task specifications (the problems which need to be solved), method specifications (the ways to solve problems), and domain models (problems need to be solved). This separation of system components actually fits quite nicely with the requirements for exploiting the context of AHSs, as shown by Figure 2.
Figure 2 illustrates our vision of the modular architecture for adaptive Web-based systems (Chepegin et al. 2003). One of the first consequences is that the different system components are all equipped with facilities to communicate with each other. Bridges are used in accordance with the UPML framework connector defined by Fensel et al. (1999), in order to specify mappings between components within the architecture. Ontologies also play an important role to define and unify the system’s terminology and properties to describe service. Each service can be specified by means of a corresponding ontology, providing common ground for knowledge sharing, exploiting the services. This leads to a highly modular architecture which offers a high degree of flexibility.

In the case of adaptive systems access to the user model via a Web service is a good example of this flexibility. It means that designers interact with or react to the user intelligently without knowing anything in advance about that user, but merely using the knowledge c interpreting it in the context of the current application. Crucial for achieving such a flexible architecture is the need for a standardization of information about users. As mentioned above, open standards (e.g. XML, RDF, OWL) allow for the specification of ontologies to stand to enable reuse and interoperability. Another key aspect is to facilitate mobile user models that follow the user across applications. If agents provide implementation views supporting mobility and autonomous behavior. A final but important requirement is for the user reuse system and knowledge components, and thus benefit from other applications.

A second characteristic seen in Figure 2 is the separation of the different components. In the traditional AHS approach, exemplified by a distinguished feature (section 2). When transforming these components into services the need for a fourth component, the application model lies in the fact that in the traditional AHS approach the adaptation model unites the actual process of how to adapt with the decision: applications, e.g. those realized in AHA!, the designer’s knowledge about why the user is served in a certain way is more or less left or implemented the way in which the information is adapted to the user and does so based on the designer’s decisions, which are not made to share and exchange the different functionalities between systems, it becomes relevant to separate the “how” and “why” in the designer’s intentions about the roles, goals and tasks in the application (related to domain model concepts and user model values) itself to the actual realization of the adaptation to follow the directions given by the application model. In fact this aligns well with the approach, mentioned above. The application model service contains a generic description of the user tasks in the context of a Role-G.

It is clear that this gives the application model service a crucial role in the system architecture. It divides the adaptation process into technical adaptation performed by the adaptation model service, while the management of the service process is coordinated from the entire architecture as displayed in Figure 2 emphasizes the fact that the core knowledge about the application processes and the goals lies in the application model service. In the interaction with the application the user is represented by a particular role (e.g. g.

Figure 2. Architecture for adaptive Web-based systems (adapted from Motta et al. 2003).
This role defines for him/her a corresponding behavior in terms of goals to achieve. To accomplish the user’s (applications) are used, which realize one or several corresponding methods. The adaptation model service receives the direct user in application model service in order to define the context for the user input for its most precise adaptation. Further the adaptation model service in order to select the relevant content to be presented to the user. The domain model service is responsible for the module domain knowledge in terms of concepts of a domain ontology. Finally, it updates the user model with new values. For instance, when adaptation, every action they perform on the user interface is communicated to the user model service, which is responsible for updating the user model. The user model is shared and used by the different system services. The next section provides a list of noteworthy issues.

## 4 Legal issues in adaptive hypermedia

We do not claim to have any meaningful expertise in legal matters. However, before people start envisioning a great future for user companies to perform personalization and adaptation for new and existing customers based on a “big brother” type of server that sells

development to customers. As a result it is wise to abide to the strictest laws when creating

<table>
<thead>
<tr>
<th>noteworthy issues are:</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Usage logs must be deleted after each session. This implies that adaptive systems may store the effect of user actions in a user’s personal profile. AHA! version 1.0 for instance kept a permanent log of user IDs and timestamps of every access to a page and of the user’s navigation. In the future version of AHA! version 2.0 the logging functionality has been removed.</td>
</tr>
<tr>
<td>- Usage logs of different services may not be combined. It is still unclear what the impact of such laws is on the proposed modular model becomes a service that can be used by different and distributed applications. We would of course like to create user models that are gathered from different sites, but this may or may not be allowed depending on the interpretation of the word “service” course program is one service, thus justifying the sharing of user model information between applications or courses within that departments of an electronic shopping mall that interact with customers independently could be considered one service, and the user’s independence remains to be seen whether this interpretation will be followed by the law.</td>
</tr>
<tr>
<td>- Users should have a real, voluntary choice between an “anonymous” login (or login using a pseudonym) and a normal login using the version 2.0 anonymous logs in possible, yielding the same adaptation as non-anonymous logs. It helps to provide users with advance notice about the data that is to be collected, and to indicate the purposes for which this is being done.</td>
</tr>
<tr>
<td>- User modeling should be transparent, or “scrutable”. Users should be able to inspect their user model and understand what it contains or erase data, especially the assumptions the system inferred about them.</td>
</tr>
<tr>
<td>- The user models should be protected using adequate security measures. When user modeling becomes a service, accessible by non-trivial users, we need to consider the impact of such laws. We have identified the following possible disadvantages and legal problems raised by this open modularized architecture with respect to access by various collaborating parties.</td>
</tr>
</tbody>
</table>

This short list is far from complete, but it already shows that for adaptive Web-based systems to become a “big thing” there are a number of issues to clarify and then implemented into the systems.

## 5 Conclusions and future work

Adaptive Web-based systems are ready to make the jump from single applications to modular distributed frameworks in which multiple models and adaptation rules. The challenge for the future is to get research groups to work together to develop standards for exchanging user model information. While new also need clarity in the legal issues involved in sharing user modeling information. We are opposed to a “big brother” style of user model with voluntary user consent some sharing will be allowed under controlled circumstances. With many applications of adaptive Web-based technologies, the idea of adaptive systems can still become a “next big thing”.

## Acknowledgements

We would like to thank the anonymous reviewers for their helpful comments, and also Geert-Jan for clarifying and improving the description of the modular article.

## References


courseware”. *Proceedings of 7th International World Wide Web Conference*, Brisbane, April


Links

Amaya Web Editor http://www.w3.org/Amaya/

Annotea http://www.w3.org/2001/Annotea/


DAML Services http://www.daml.org/services/