Proposed Evaluation Framework for Adaptive Hypermedia Systems

Arpita Gupta, P. S. Grover

Department of Computer Science, University of Delhi,
Delhi – 110 007, India
arpita_aggarwal@hotmail.com, psgdelhi@yahoo.com

Abstract. Although a number of frameworks exist for the evaluation of Adaptive Hypermedia Systems (AHS), recently suggested layered frameworks have proved useful in identifying the exact cause of the adaptation failure or any other error in the system. This paper presents an evaluation framework for AHS for internet which is an extension of the layered frameworks and adds new dimensions to them. It treats evaluation as an integral part of development process of AHS and also evaluates the successful access of AHS on the internet. The framework has four dimensions which are orthogonal to each other – Environment – the environment in which AHS is accessed, Adaptation – the type of adaptation used, Development Process – software engineering life cycle steps used for developing AHS and the Evaluation Modules – the layers of AHS which are evaluated in context of other dimensions.

1 Introduction

Adaptive Hypermedia Systems (AHS) are designed and built with the intention of providing tailor made information to individual users according to their preferences, goals and knowledge. With the advent of internet as a common source of information, they have found a platform to reach heterogeneous groups of users using different devices for assessing AHS. With this increases the challenge of catering to a wide variety of users in differing environments and also the added responsibility of working without making mistakes since a single mistake can make the user lose trust in the system – maybe forever. Therefore, evaluating the AHS is of utmost importance. Moreover, it is equally important to have a correct method of evaluation since an incorrect method can lead to wrong conclusions [3].

Earlier evaluation studies compared adaptive versions of the system with the non-adaptive versions [2, 4]. A major criticism of this approach was that the non-adaptive versions – usually implemented using adaptive version with their adaptivity switched off – were not "optimal" [11].

Recently some layered evaluation frameworks were suggested which do not treat evaluation as a "monolithic" process but instead divide it into layers [3, 20, 23]. This approach helps in identifying the exact cause of the adaptation failure or any other error. These evaluation frameworks basically differ in layer granularity and do not take "extensibility or maintainability" of the AHS into consideration [10]. Another evaluation framework – Extended Abstract Categorization Map (E-ACM) [21] has been suggested to guide adaptation evaluation and design. Most of them do not take the development process into consideration resulting in detection of errors and weaknesses in the system which can prove to be expensive to correct at a later stage. Moreover issues like maintenance of the system, the environment in which they will be used – location and devices accessing AHS, etc., have not been addressed in the current frameworks.

There can be a number of factors which affect the evaluation process of an AHS. Internet provides the opportunity of accessing AHS using a variety of devices like desktops, mobiles, PDAs, in any location of the world, to the users having diverse skills, capabilities and knowledge. The AHS itself can belong to any application domain having some specific characteristics. All these elements form the environment of the AHS. The adaptation in the AHS can be static or dynamic depending on the time and process of adaptation.

Similar to software engineering, AHS also involves the analysis, design, implementation and maintenance phases of development process. During these phases, they should be evaluated for the validity of input acquisition, correctness of inferences drawn from these inputs, correctness of various models created by the AHS, the adaptation decisions taken based on these models and their final presentation to the user.

We propose an evaluation framework treating the evaluation as an integral part of the development process of AHS and taking the accessing environment and the type of adaptation provided by AHS into consideration while evaluating individual modules of AHS. The framework consists of four orthogonal dimensions: Environment, Adaptation, Development process and the Evaluation modules. Next Section describes the framework.

2 Proposed Evaluation Framework

The proposed framework is an extension of layered evaluation frameworks where the layers need to be evaluated in context with other dimensions. The benefits of the framework are: (i) it allows a structured, layered view to better understand the various aspects of AHS (ii) it can be used as a conceptual framework for evaluating existing approaches for AHS (iii) it may be used during the development of next generation AHS using software engineering steps.

The framework consists of 4 dimensions - Environment, Adaptation, Development Process, and Evaluation Modules. These dimensions are orthogonal to each other i.e. all the evaluation modules should address all the components of environment and adaptation during each phase of development process.

Figure 1 shows our proposed evaluation framework and the following sections describe these dimensions and their components.

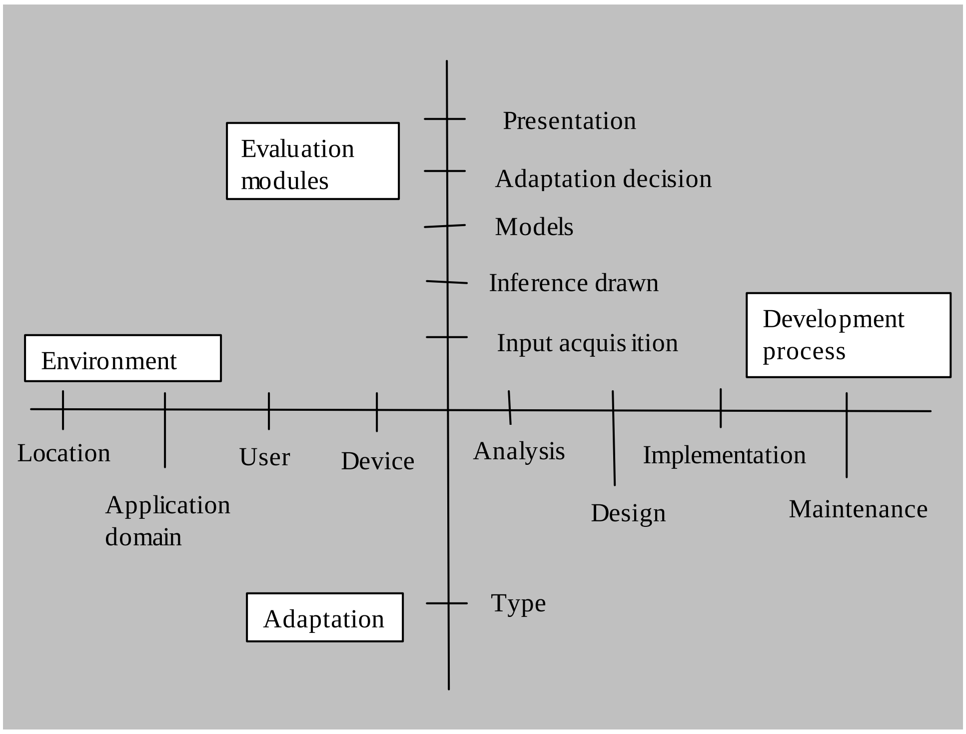

Figure 1. Proposed Evaluation Framework for AHS

2.1 Environment

The first dimension – environment – is the set of conditions to which AHS has to adapt itself i.e. they are the circumstances of consumption for AHS. There can be innumerable variables affecting the environment; we group them under the following components:

2.1.1 Device: Advent of web-capable appliances with limited abilities such as PDAs and mobile telephones along with desktops – have made one-size-fit-all paradigm obsolete since the range of hardware and software used at client side of web-based systems is extremely wide. AHS need to be evaluated for the correct acquisition of the device characteristics and smooth running with hardware features like display sizes, local storage size, method of input, processing speed and software features like browser versions, available plug-ins, Java and JavaScript etc. Kobsa et al [16] have discussed acquisition of such data by the AHS.

2.1.2 User: The personal characteristics of user such as demographic data, user knowledge, user skills and capabilities, user interests and preferences, his goals and plans have been used by many AHS for adaptation [2, 22]. Along with these features, AHS should be evaluated for adaptation according to all users including disabled and elderly users [15].

2.1.3 Application Domain: AHS can be developed for a wide range of applications which differ in characteristics. Brusilovsky [5] have identified several application domains and existing AHS for them. The application specific characteristics constitute critical parameters while evaluating AHS and should be evaluated along with general characteristics of AHS since the traditional concerns change with the application domain.

2.1.4 Location: Information about the geographical location of the accessing device can be used to filter and adapt or recommend the content of AHS and should be evaluated for the correct delivery of the same. Kobsa et al [16] have given methods for acquisition of information about the location and their use in adaptation procedure.

2.2 Adaptation

The second dimension of the framework is adaptation which can be of two types: **static** adaptation and **dynamic** adaptation, depending upon the time and process of adaptation. Static adaptations are specified by the author at the design time or determined once only at the startup of the application. Fink et al [9] used static adaptation in AVANTI project. Dynamic adaptation occurs during runtime depending on various factors like inputs given by the users during use, changes in user model, adaptation decision taken by AHS etc. Kappel et al [13] have distinguished three options for dynamic adaptation – immediate dynamic adaptation i.e. adaptation done as soon as context changes, deferred dynamic adaptation i.e. adaptation done only after the user has requested the page which is subject to adaptation, and periodic adaptation i.e. adaptation is done periodically. Example of system having dynamic adaptation is AHA [8].

2.3 Evaluation Modules

The third dimension of evaluation framework consists of evaluation modules which need to be considered for evaluation of AHS. These modules have been suggested in the layered frameworks [3, 20, 23]. Our perspective is to evaluate them with respect to other dimensions of the framework.

2.3.1 Input Acquisition: Inputs are required from the environment as well as from the user. These can be taken manually (i.e. user feeds them), automatically (i.e. system takes the input itself e.g. Type of device, its screen size, location etc) or semi automatically (i.e. combination of both – some input through user, some automatically) [13].

The inputs taken by the system – either manually or automatically might not carry any semantic information, but they need to be evaluated for the reliability, accuracy, precision, latency, sampling rate, so that the inferences drawn from them are valuable.

This needs to be done at all stages of development process – analysis, design, implementation and maintenance phases, for both static and dynamic adaptations, for all intended devices, users, locations and application domains.

2.3.2 Inferences Drawn: Previous layer was involved with the data collection, this layer gives "meaning" or "semantics" to it i.e. it draws inferences from it. Evaluators need to check if these inferences or the conclusions drawn by the system concerning the user-computer interaction are correct since it is not necessary that there will be a direct – one to one mapping between raw data and their semantically meaningful counterparts.

Moreover, inputs given by various users of different devices or application domain might need different interpretations. Evaluators need to check if all such interpretations have been analyzed, designed and implemented in the AHS for both static and dynamic adaptations.

2.3.3 Models: For achieving the required adaptations, various models are created by the system. Benyon and Murray [1] specified three models – user model, domain model, interaction model. Nora Koch [18] has described four models for carrying out the adaptation – user model, navigation model, presentation model, and adaptation model.

These models are based on the inferences drawn in the previous stage and are suppose to imitate the real world. They need to be evaluated for validity i.e. correct representation of the entity being modeled, comprehensiveness of model, redundancy of model, precision of the model, sensitivity of the modeling process [20].

2.3.4 Adaptation Decision: Given a set of properties in the user model, sometimes there can be more than one adaptation possible. In this module, evaluation of the most "optimal" adaptation is done using criteria like necessity of adaptation, appropriateness of adaptation, acceptance of adaptation [20]. Careful evaluation is needed to ascertain that increase in adaptation is not resulting in decreased usability [5].

2.3.5 Presentation: This module involves the human-computer interaction and needs to be evaluated for criteria like completeness of the presentation, coherence of presentation, timeliness of adaptation, user control over adaptation [20].

2.4 Development Process

The fourth dimension of the framework is the development process comprising of phases of software life cycle i.e. analysis, design, implementation and maintenance. Benyon and Murray [1] gave a star approach to interactive system development taking evaluation as central element and system analysis, specification of user requirements, design, prototype and implementation as the peripheral elements.

During each phase of this dimension, evaluation of individual elements of environment, adaptation and evaluation modules is done with respect to each other.

2.4.1 Analysis: This phase involves gathering information about the problems of current system, and/or identifying the requirements and constraints of the system to be developed. Main components of this phase are:

- *Functional analysis*: establishes the main functions that the system is expected to perform and how it is to perform.
- *Environment analysis*: This analyzes the environment in which the system is expected to be accessed – including the physical aspects like location, device and other aspects like application domain, type of user.
- *User and task analysis*: This determines the scope of cognitive characteristics like user's preferences, goals, knowledge and other attributes required in user model e.g. search strategy required, assumed mental model etc.
- *Interface analysis*: It identifies features like effectiveness, learnability, flexibility and attitude required of the system.
- *Data analysis*: This involves the analysis of input acquisition to identifying the data to be stored and manipulated by the system, and to understand and represent the meaning and structure of data in the AHS.
- *Analysis of Models*: This involves the analysis of various models maintained by the system such as user model, domain model, navigation model, adaptation model.

To evaluate this phase, checklists can be prepared for different types of analysis mentioned above, corresponding to various components of different dimensions of the evaluation framework. For example, a checklist is prepared for the desktop computer's functional requirements of static adaptation for input acquisition; another checklist for the PDAs for the same specifications is prepared.

2.4.2 Design: Design phase defines the overall system organization by transforming the functions and tasks defined during analysis phase into software components and their externally visible properties of those components and their relationships. It is recommended to design the adaptive parts of the system in parallel with the whole system so as to have a successful adaptation [11]. Some of the components for this phase are:

- *Architectural Design*: Many architectural designs have been suggested for the adaptive systems [1, 12, 19]. The modular architecture model presented in figure 2 is especially designed for our evaluation purpose.

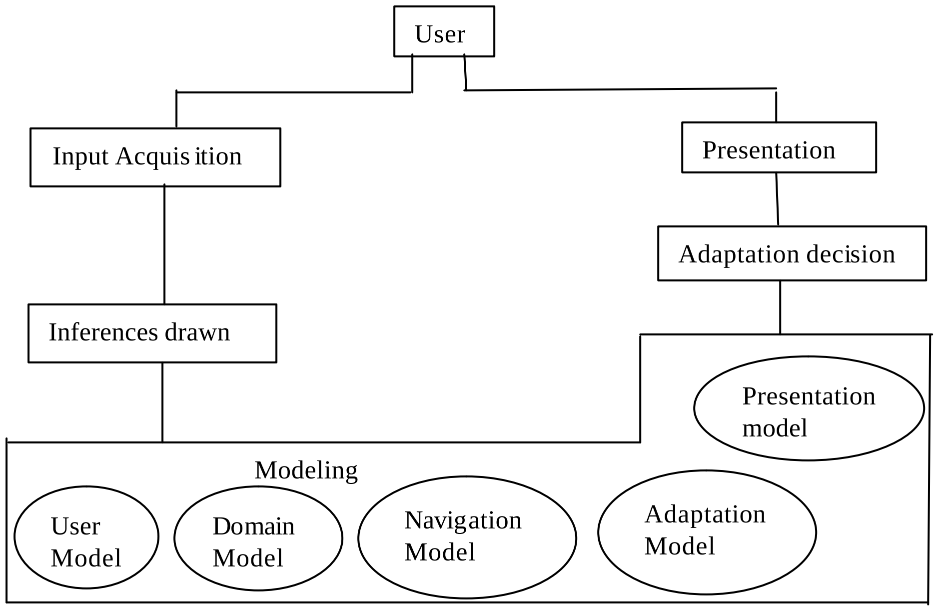

Figure 2. Architecture model for evaluation of adaptive hypermedia system

In this model, "Input acquisition" module acquires input by the user through keyboard or mouse or takes it from the environmental factors such as the device and location accessing the AHS. This input is verified and then passed on to the next module "Inferences drawn" for drawing logical inferences and conclusions which are then stored in some dynamic models created by the system such as user model. The "modeling" module consists of various models – both static and dynamic such as user model, domain model, navigation model, presentation model and adaptation model. Different AHS can have one or more such models which are used at various levels of adaptation. Depending on some these models, "adaptation decision" module takes the decision so as to choose the best adaptation techniques out of the available ones in the system. Then the "presentation" module presents the final content and links to the user.

Evaluation of architectural design is important because it defines constraints on implementation and maintenance, making it easier to reason about and manage changes, and defines system's quality attributes.

- *Content Design*: Content forms one of the most important aspect of the AHS since it is for the content that the AHS exists. Therefore, it should be verified for accuracy and precision. Moreover, there is a need to identify and evaluate the content which would be visible to different users (according to their individual user model), on different devices and location.

- *Navigation Design*: Since a number of navigational techniques are present such as direct guidance, link hiding etc [5] which can be used with various navigation aids like icons, graphicals, there is a need to ascertain that the chosen one is appropriate for content and consistent with the heuristics leading to high quality interface designs and also ensure that aids like site maps and table of contents are designed into the system.

  Evaluation of navigational design is needed to ensure the maintenance of coherence of the overall system as the user moves across application since it is related to AHS's underlying communicative performance.

- *Interface Design*: Since user interface is the "first impression" of the system, a poorly designed interface might disappoint user regardless of the value of content, sophistication of architectural design and navigation techniques used. Therefore, careful evaluation is necessary for well structured ness and ergonomically sound interface designs.

Design phase can be evaluated by using metrics such as structural complexity metrics, navigational metrics, usability metrics etc. Moreover, a good architecture exhibits low coupling and high cohesion in terms of some decomposition of functionality [14].

2.4.3 Implementation: During the actual implementation of the system, evaluation of each module of AHS should be carried out individually and then in integration with the other modules for the successful adaptation at various levels – both static and dynamic, for different users, on different devices. Some metrics like behavioral complexity, reliability metrics, precision, software size and length metrics help in evaluation of the system as a whole. Moreover, evaluation of tools can be done in relation to the activities in which they are used to indicate their availability for each kind of activity and how the tool supports it.

2.4.4 Maintenance: AHS might require updation at any moment; therefore, during the design phase care should be taken to design hyperspace in modular way so that change in structure can be made by changing the relations among these modules. Automatic link generation should be preferred over static links to preserve the link coherence. Maintenance of static links is very complex as any change in a node position in hyperspace necessitates it to revise all the documents that include links to this node, in order to update them.

Content maintenance can be made easy by storing contents apart from the concept structure or the navigational options since contents are external and easily updateable. Finally, database should be used to store information about different system components to facilitate the management and maintenance of these components and guarantee data consistency [7]. Checklists can be prepared for these and metrics can be used to measure the ease of maintenance such as complexity metrics, reuse metrics or expandability metrics.

Table 1 gives an example to illustrate the way in which to use the framework. While developing an adaptive tutoring system for internet, during the implementation phase, the precision of the content is measured for the presentation module and the values are filled in the table 1. The values are filled by using various metrics appropriate for different stages and can be in any unit of measurement. For the complete evaluation, many such tables would be required for various phases.

Table 1. An Example to show how to use the framework dimensions (values are only for demonstration purpose and are not exact).

Precision of content during implementation phase of presentation module		Adaptation		
			Static Adaptation	Dynamic Adaptation
Environment	Device	PDA	85%	80%
		Mobile	80%	80%
		Desktop	95%	96%
	User	Novice	70%	76%
		Average	90%	90%
		Expert	80%	85%
	Location	.		
		.		
		.		
	Application Domain	.		
		.		
		.		

3 Conclusion

Our proposed evaluation framework integrates the AHS development process, the accessing environment, the different types and levels of adaptations involved in AHS and the evaluation modules of layered frameworks. Several factors that impact the AHS evaluation can be organized around these framework perspectives.

The framework is a mechanism to gain insight into and an understanding of AHS for internet. It can be used for summational evaluations once the AHS has been completed by replacing the “development process” with “initial goals and achieved goals” and checking with the rest of the three dimensions. It can also be used for formative evaluations during the development of the system by establishing goals for each phase and then compare the actual results.

The dimensions and their elements suggested in the framework have been addressed more or less globally. Subfactors can be established for each element which can be evaluated objectively or subjectively.

We are in the process of developing and evaluating an adaptive tutoring system with the authoring tool AHA using this framework where analysis phase has check-lists and set of goals prepared according to the requirements. Metrics largely are being used during design, implementation and maintenance phases for the purpose of evaluation. The results of this study will be reported later on.

4 References

1. Benyon D., Murray D.: Adaptive Systems: from intelligent tutoring to autonomous agents. Knowledge-Based Systems, 6(4), 197-219 (1993)
2. Boyle C., Encarnacion A. O.: Metadoc: An Adaptive Hypertext Reading System. In P. Brusilovsky et al. (Eds.), Adaptive Hypertext and Hypermedia, ©1998 Kluwer Academic Publishers, pg 71-89, (1994)
3. Brusilovsky P., Karagiannidis C., Sampson D.: The Benefits of Layered Evaluation of Adaptive Applications and Services. In Weibelzahl S., Chin D. N., and Weber G. (eds), Empirical evaluation of Adaptive systems, Proceedings of workshop at the eighth international conference on user modeling, UM2001, pg 1-8, Freiburg
4. Brusilovsky P., Eklund J.: A Study of User Model Based Link Annotation in Educational Hypermedia. Journal of Universal Computer Science, vol. 4, no. 4 (1998), pg 429-448, (1998)
5. Brusilovsky P.: Methods and Techniques of Adaptive Hypermedia, P. Brusilovsky et al (eds.), Adaptive Hypertext and Hypermedia, 1-43, Kluwer Academic Publishers, 1998, printed in the Netherlands
6. Brusilovsky P.: Efficient Techniques for Adaptive Hypermedia, In C. Nicholas and J. Mayfield (eds): Intelligent hypertext: Advanced techniques for the world wide web. LNCS, 1326, Berlin:Springer-Verlag, 12-30
7. Carro, R.M.: Adaptive Hypermedia in Education: New Considerations and Trends. Proceedings of the 6th World Multiconference on Systemics, Cybernetics and Informatics (Orlando, Florida), Vol. 2, ISBN: 980-07-8150-1, 452-458, (2002)
8. De Bra, P., Smits D., Stash N.: AHA! The next generation. ACM Conference on Hypertext and Hypermedia, May 2002
9. Fink, J., Kobsa, A, Nill, A.: Adaptable and adaptive information provision for all users, including disabled and elderly people. In The New Review of Hypermedia and Multimedia, 4, 163-188
10. Gupta A., Grover P. S.: Comparison of Evaluation Frameworks for Adaptive Hypermedia. To be published in Proc. of 2nd International Conference on Quality, Reliability and Information Technology (ICQRIT Dec 2003), New Delhi
11. Höök K.: Steps to take before intelligent user interfaces become real. Interacting with Computers, 12, pg 409-426, (2000)
12. Jameson, A.: Systems That Adapt to Their Users: An Integrative Perspective. Saarbrücken: Saarland University, (2001)

13. Kappel, B. Pröll, W. Retschitzegger, W. Schwinger: Customisation for Ubiquitous Web Applications - A Comparison of Approaches. Int. Journal of Web Engineering and Technology (IJWET), Volume 1, No. 1, 2003, pp. 79-111, [ISSN 1476-1289]
14. Kazman R., Clements P., Bass, L.: Software architecture in Practice. Addison-Wesley, 1998, pg 218
15. Kobsa, A., Stephanidis, C. : Adaptable and Adaptive Information Access for All Users, Including Disabled and Elderly People. Proceedings of 2nd Workshop on Adaptive Hypertext and Hypermedia, HYPERTEXT'98, Pittsburg, USA, June 20-24 (1998)
16. Kobsa A., Koenemann J., Pohl W.: Personalised Hypermedia Presentation Techniques for Improving Online Customer Relationships. The Knowledge Engineering Review, Vol. 16:2, 111-155, (2001), Cambridge University Press
17. Mendes E., Hall W., Harrison R.: Applying Metrics to the evaluation of Educational Hypermedia Applications, Journal of Universal Computer Science, vol. 4, no. 4 (1998), pg 382-403
18. Nora Koch, PhD Thesis: Software Engineering for Adaptive Hypermedia Systems: Reference Model, Modeling Techniques and Development Process. Ludwig-Maximilians-University of Munich, Germany, December 2000
19. Oppermann, R.: Adaptively supported adaptability. International Journal of Human Computer Studies, 40(3), 455-472, (1994)
20. Paramythis A., Totter A., Stephanidis C.: A Modular Approach to the Evaluation of Adaptive User Interfaces, in Weibelzahl, S. Chin, D.N., Weber, G. (eds), Empirical Evaluation of Adaptive Systems, Proceedings of workshop at the eighth International Conference on User Modeling, UM2001, pg 9-24, Freiburg, (2001)
21. Tobar, C. M.: Yet Another Evaluation Framework. In: Weibelzahl, S. and Paramythis, A. (eds.). Proceedings of the Second Workshop on Empirical Evaluation of Adaptive Systems, held at the 9th International Conference on User Modeling UM2003, Pittsburgh, (2003), pp. 15-24
22. Weber, G, Specht, M.: User modeling and adaptive navigation support in WWW-based tutoring systems.. In a Jameson, C Paris and C Tasso (eds) User Modeling: Proceedings of the sixth International Conference Springer-Verlag, (1997), 289-300
23. Weibelzahl S., Lippitsch S., Weber G. (2002): Advantages, Opportunities, and Limits of Empirical Evaluations: Evaluating Adaptive Systems, Künstliche Intelligenz, 3/02, 17-20