An Adaptation Method by Feedback in an Evolutionary Hypermedia System

Nuria Medina Medina
University of Granada
E.T.S.I.I., Dept. LSI, 25
Granada – 18071. Spain
(+34) 958 240634
nmedina@ugr.es

Fernando Molina Ortiz
University of Granada
E.T.S.I.I., Dept. LSI, 31
Granada – 18071. Spain
(+34) 958 243180
fmo@ugr.es

Lina García Cabrera
University of Jaén
Campus Las Lagunillas, A-3,132
Jaén - 23071. Spain
(+34) 953 212475
lina@ujaen.es

ABSTRACT

In this paper, we describe a model for hypermedia systems that aims to reduce the main problems present in the development of these systems. This model, called SEM-HP, is defined as systemic, semantic, adaptive and evolutionary. It is systemic because it separates the aspects of representation, presentation, navigation and adaptation of the hypermedia into four subsystems; semantic because it makes explicit the meaning of the offered information; adaptive because its functioning varies according to the features of the user navigating it; and evolutionary because it supports mechanisms that allow the author to restructure the hypermedia in a flexible and consistent way. The paper also presents and details a method called Adaptation by Feedback, which analyzes and integrates the navigational behaviour of the users browsing the hypermedia, compares the structures traced by the users with those previously defined by the author, and suggests the necessary modifications so that by using the evolutionary capacity of the system the author can bring both structures closer.

Categories and Subject Descriptors

H.5.4 [Information Interfaces and Presentation]: **Hypertext/Hypermedia**.

General Terms

Design, Human Factors and Theory.

Keywords

Hypermedia Systems, User Adaptation and Software Evolution.

1. INTRODUCTION

Although there has been a massive increase in hypermedia systems over the last twenty years, they do in fact date back to the late thirties. It was at this time that Vannevar Bush wrote a draft of the MEMEX system [1] which presented the use of a principle of association between informational resources in such a way that the user could access the resources, regardless of their type, simply by association of ideas.

For technical reasons, hypermedia systems did not become a reality until the sixties in research projects such as the one directed by Doug Engelbart at the Stanford Research Institute [2]. Many authors have defined the term "hypermedia", coined by Nelson in 1965 [3]. For example, in 1989, Shneiderman [4] described the hypermedia as a "database with active cross-references that allow the reader, according to his wishes, to jump to other places in the database".

Since then, the state of the art in hypermedia systems has advanced significantly: the word hypermedia has grown in popularity, hypermedia systems are proving to be a very powerful tool for representing information, and several authors have proposed reference models with the aim of guaranteeing that they are correctly designed. We can mention Campbell's HAM model (1987) [5], the Trellis metamodel, proposed by Furuta and Stotts (1989) [6], or Halasz's Dexter model (1990) [7].

Beginning in the 90s, the interest in adapting the presentation and navigation of hypermedia systems to the particular features of their users has resulted in a research field with different models and architectures for adaptive hypermedia systems. As a reference, we can mention the AHAM model [8] (proposed by Paul de Bra) and the AHA architecture which is based on it [9], or, with a similar architecture, the Munich Reference Model [10] (proposed by Nora Koch).

At the present time, we cannot deny that hypermedia systems are widely and successfully used for assisting formative and educative processes in every field. Nevertheless, we believe that it is still necessary to continue working in the following aspects:

- Making explicit the semantics in the representation of the information offered by the system.
- Separating different concepts such as information representation and information navigation.
- Sufficiently contemplating user adaptation so the functioning of the system really depends on each user.
- Providing evolutionary mechanisms that allow the developer (or author) to modify the structure of the hypermedia system in an easy, flexible and consistent way.

In order to meet these goals, we propose the SEM-HP model, which takes into account both the adaptation and evolution of the hypermedia system, separating the different aspects and making its semantic explicit. We also provide an author tool, JSEM-HP,

ICWE'06, July 11-14, 2006, Palo Alto, California, USA.
ACM 1-59593-352-2/06/0007.

which eases the development and maintenance of the adaptive hypermedia system, hiding part of the complexity of the model.

An important problem associated to the evolution of the hypermedia system is the fact that the modifications made by the author in the hypermedia system are usually caused by changes in the author's particular perception of the knowledge domain and does not take user perceptions into account. Nevertheless, users will undoubtedly feel more comfortable using a hypermedia system whose structure contemplates the knowledge model in a way that is close to their mental conception.

With this objective, in this paper we will present the Adaptation by Feedback method. This method obtains the mental structure that users have about the system's knowledge domain by analyzing the navigation strategy they mostly use. It then provides the author with useful information for changing the system structure so that author and user perceptions get closer.

The proposed process is therefore a semi-automatic process, driven by the system users, suggested by the system itself, and finally directed by the author. In addition, since the redefined structures are provided to new users, the knowledge extracted from the group of users is used to guide future users. Novice users can therefore benefit from the process of adaptation to the group, learning from the navigation strategies of the more experienced users [11].

Section 2 of this paper briefly describes the SEM-HP model. Section 3 focuses on the Adaptation by Feedback method. Finally, our conclusions, related work and further work are presented in the remaining sections.

2. SEM-HP MODEL

The SEM-HP model [12][13] is a semantic, systemic, evolutionary and adaptive model that divides a hypermedia system into four related and interacting subsystems (Figure 1).

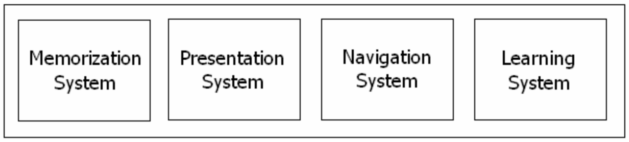


Figure 1. SEM-HP model

This division into subsystems aims to separate the aspects of representation, presentation, navigation and user adaptation, easing both the development process of the hypermedia system and its subsequent maintenance. In addition, it allows evolutionary mechanisms to be defined that guarantee the integrity of the hypermedia system after any change it may suffer.

2.1 Memorization Subsystem

The Memorization Subsystem stores the hypermedia system's knowledge domain. In order to do this, the representation model used is a network that structures the information according to its meaning, cataloguing each available resource by means of a real or abstract concept.

We call this semantic network the **Conceptual Structure of Memorization** (CS_M) and it has two different kinds of nodes: concepts and items. Concepts are semantically labelled ideas that are represented with circles, and are associated with each other by means of conceptual relations. Items are information resources represented by rectangles and associated to the concepts they explain through functional associations ("example", "definition", "introduction", "opinion", etc.).

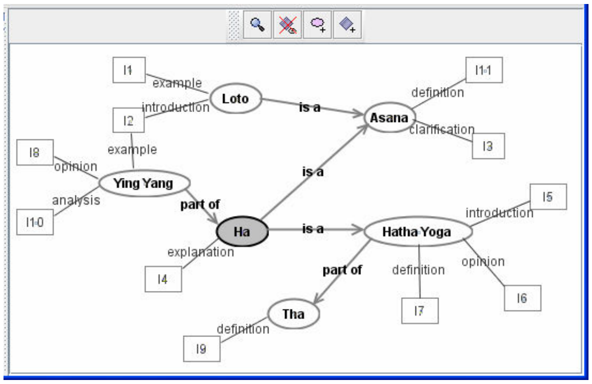


Figure 2. Yoga CS_M

Figure 2 shows the CS_M created by the author in order to represent a Yoga-based knowledge domain in a simple way. For example, the concept "Asana" has two associated information items: I11 (which defines this concept) and I3 (which clarifies it).

2.2 Presentation Subsystem

The Presentation Subsystem allows different views to be built of the knowledge domain. The author creates each view, called a Conceptual Structure of Presentation (CS_P), by selecting concepts, relations and items in the CS_M. A CS_P is therefore a subset of the CS_M which focuses on a particular knowledge subdomain (Figure 3). In order to make this information explicit, the author labels each presentation indicating the knowledge subdomains it captures.

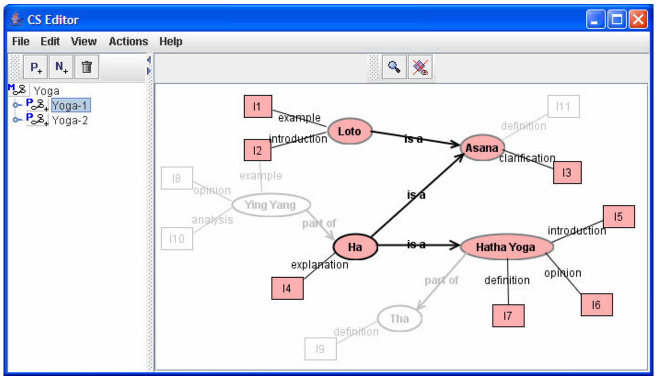


Figure 3. Yoga-1 CS_P

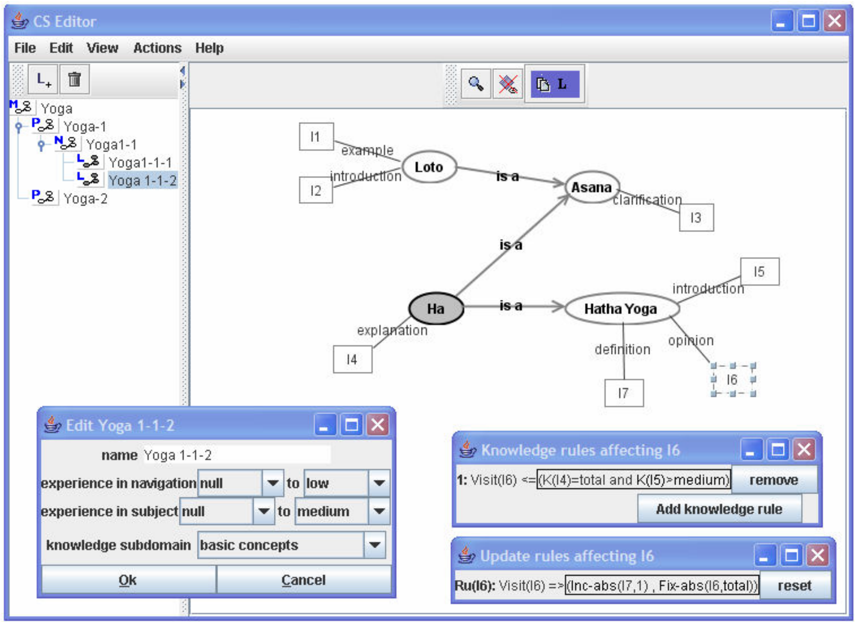
2.3 Navigation Subsystem

The Navigation Subsystem automatically generates a set of **order rules**, which requires the information resources to be visited in an order that is coherent with the semantic relationships defined between the concepts. This means that an item Ij can only be visited if the user has previously visited an item associated to a concept from which there is a relation whose destination is a concept to which Ij is associated. Accordingly, in the example CS_P, in order to visit I3 the previous item visited by the user must have been I1, I2 or I4. Further discussion about the order rules and how they can be modified by the author can be found in [14].

2.4 Learning Subsystem

The Learning Subsystem allows the author to define other navigation rules which are not based directly on the visits performed by the user but on the knowledge acquired after them. The so-called **knowledge rules** therefore allow the author to establish the pedagogic prerequisites needed in order to visit an item in optimal learning conditions. More specifically, we work with five knowledge degrees associated to the following semantic labels: “null”, “low”, “medium”, “high” and “total”.

Figure 4 shows a Conceptual Structure of Learning (CS_L) created from the example CS_P. The knowledge rule defined for item I6 establishes that, in order to adequately understand the information it contains, the user knowledge about I4 must be “total” and about I5 it must be higher than “medium”.



name Yoga 1-1-2
experience in navigation null to low
experience in subject null to medium
knowledge subdomain basic concepts

1: Visit(I6) <= (K(I4)=total and K(I5)>medium)

Ru(I6): Visit(I6) => (Inc-abs(I7,1) , Fix-abs(I6,total))

Figure 4. Yoga 1-1-2 CS_L

Another set of rules, called **update rules**, allows the system to infer the user’s knowledge acquisition whenever an information item is read. By default, the generated update rule for an item assigns a “total” knowledge about the item to the user once it has been visited. However, the system provides mechanisms so that the author can define another type of knowledge acquisition about the visited item and even about other related items. In the example, we can see that the author has associated to the visit of I6 an increment of one degree of knowledge about I7.

Having defined the knowledge and update rules, the author labels the CS_L indicating the user profile for which it is suitable. More specifically, the author sets an interval of the degree of experience in the subject and an interval of the degree of experience in navigation. In the example, the CS_L has been designed for novice users since the recommended navigation experience is between “null” and “low”, and the experience in the subject between “null” and “medium”.

2.5 Use of the Hypermedia System

The division into subsystems allows an incremental development process, i.e. on the basis of a CS_M the author can create multiple presentations, and for each CS_P several navigation schemes can be created, and different learning structures can be defined for each one, thereby obtaining a wide CS_L set.

The CS_L is the structure provided to the user in order to browse the hypermedia system, and the variety of available CS_L enables the one best fitting the user's profile to be chosen (Figure 5). This selection is performed by comparing the CS_L’s labels with the user's experience in the subject and in navigation, and with the knowledge subdomain he is most interested in [13].

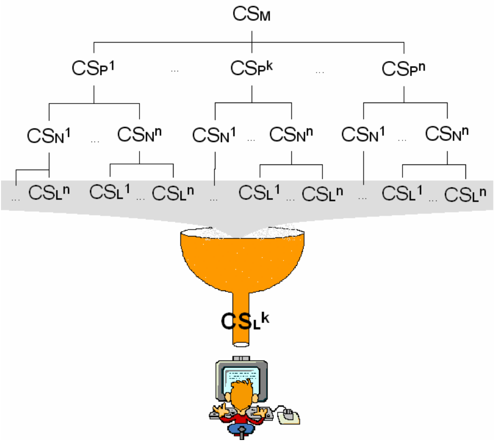


Figure 5. Choosing a CS_L for a user

The CS_L offered to the user is also adapted while he browses the system. More specifically, methods of guiding and orientation are applied [15], and this becomes effective through techniques such as hiding and disabling items the user is not yet ready to understand [13], or generating guided routes that enable the user to achieve a knowledge goal [16].

In order to perform the adaptation, the system maintains a user model which stores the following pieces of information (among others) about the user: subject experience, navigation experience, subdomain of interest, preferences about the information items (language, date, author, etc.), degree of knowledge about each information item (current knowledge state), and the knowledge state the user aims to achieve (goal knowledge state).

The user's knowledge state is automatically calculated by running the update rules defined in the Learning Subsystem for the current CS_L. Knowledge rules are used to decide if the user is ready, at a given moment, to assimilate the information provided by an item, which will decide the guided route that is offered to achieve the user's goal, or which items are hidden in the current structure in order to avoid problems of comprehension and disorientation [13][16].

2.6 Evolution of the Hypermedia System

Adaptation involves a change in the functioning of the system so that it adjusts itself to the features of the current user. However, hypermedia systems frequently need to change their structure in order to include new information, structure it in a better way, or change the way in which it can be browsed.

The problem is that if there is no predefined evolutionary mechanism, the system may be left in an inconsistent state after any structural change has been made to it. For example, what would happen to the CS_M if the author were to remove the concept Loto in the Memorization Subsystem? Would items I1 and I2 be removed? What would happen to the relations associated to the concept Loto? How would this affect the CS_P defined in the Presentation Subsystem? And the rules in the Navigation and Learning Subsystems?

In order to solve these issues, the SEM-HP model separates each subsystem into two abstraction levels: system and metasystem (Figure 6).

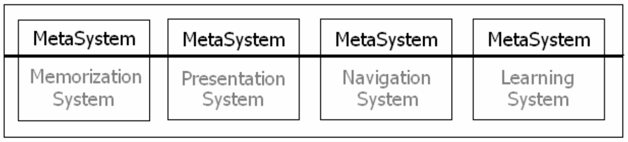


Figure 6. SEM-HP meta-model

The meta level is therefore responsible for the evolution of the system, for which it defines a set of evolutionary actions that allow structural modifications to be performed in the relevant subsystem, thereby guaranteeing that both this subsystem and others that may be affected by the change are subsequently left in a consistent state. More details about how the evolutionary process is performed are provided in [14].

3. ADAPTATION BY FEEDBACK

The hypermedia system is usually restructured based only on the author's personal perception with no user feedback; at most, some sort of questionnaire may be carried out, but few users will generally respond, and not particularly willingly. Nevertheless, it is very important that both the hypermedia system's behaviour and also its structure are adapted to its users. With this objective, we propose a method which, from the data obtained from the constant use of the system by many different users, enables inference of the mental structures which system users have about the knowledge domain provided. We call this the Adaptation by Feedback method, even though it is not a purely adaptive method since once the users' perceptions about the system have been obtained, they are then compared with the author's, and the necessary structural modifications are identified (but not automatically carried out) so that author and user perceptions may be brought closer.

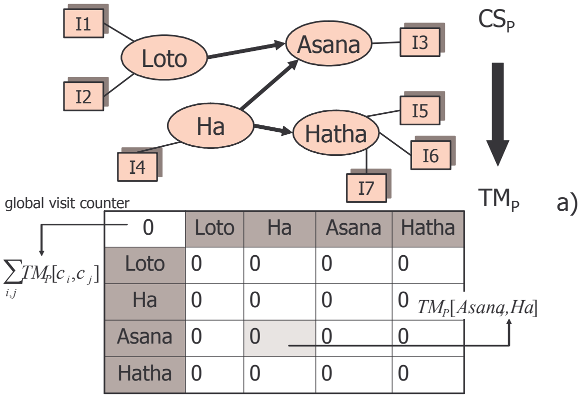
Any change in the structure of the hypermedia system can have serious repercussions and will affect every user, and must therefore be carefully considered by the author. The necessary modifications to match the author's representations with the users' mental conception are therefore only suggested to the author, who may carry them out through the corresponding evolutionary actions. In the following subsections, we will explain in further detail the process followed in order to obtain these suggestions.

3.1 Presentation Transition Matrix

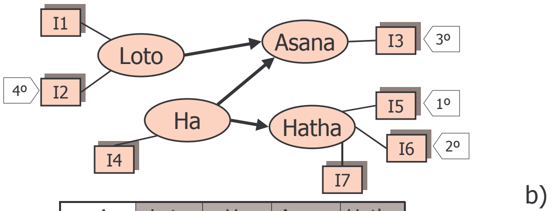
We consider the users of a CS_P to be all the users who use one of the learning structures defined from it. The users navigate directly on the graphical representation of the semantic net, selecting items on it. For each CS_P the system creates a **Transition Matrix** (TM_P), which will be used to identify the conceptual relationships which the users have in mind while navigating, since they are not forced to follow the already established semantic associations among concepts.

The transition matrix TM_P identifies the CS_P that the users have in mind. In other words, if most users visit an item associated to the concept cj immediately after visiting an item linked to ci, this means that most users (either consciously or not) accept a semantic relationship between both concepts (ci → cj).

The TM_P associated to a CS_P will have a row and a column for each concept included in the presentation. The cell TM_P[ci, cj] represents the number of times any user has followed the conceptual relation ci → cj by visiting an item linked to ci just before one linked to cj. Figure 7 shows the TM_P for the example CS_P in its initial state (a) and after a user has visited four items in the indicated order (b).



0	Loto	Ha	Asana	Hatha
Loto	0	0	0	0
Ha	0	0	0	0
Asana	0	0	0	0
Hatha	0	0	0	0



4	Loto	Ha	Asana	Hatha
Loto	0	0	0	0
Ha	0	0	0	0
Asana	1	0	0	0
Hatha	0	0	1	2

Figure 7. Initial TM_P (a) and TM_P after four visits (b)

The total number of visits carried out in a CS_P can be calculated by adding all the cells of its TM_P. Comparing this global visit counter between different CS_P of the same CS_M, the system can know which presentations are most visited (and can therefore know what kind of user uses the system most frequently), and it can detect presentations that may be removed due to a very low visit counter (this may mean that the features of this presentation do not match almost any user profile).

In addition, as indicated in Equation 1, by combining the number of visits carried out in a CS_P, the total number of visits in the hypermedia system, and the number of available presentations, the system obtains a relative index of visits for each presentation.

$$Ivisits\left(CS_P\right)=\frac{visits\left(CS_P\right)}{\left[\frac{\sum_{k=1}^{r} visits\left(CS_P^k\right)}{r}\right]}$$

with r being the number of presentations of CS_M **(1)**

This index is used to form two sets: P+ (with the most visited CS_P) and P- (with the least visited CS_P) depending on whether the obtained value exceeds a threshold defined by the author or not. By default, the threshold is 1, which means that a presentation is included in P+ if its number of visits is higher than the theoretical average obtained by dividing the total number of visits performed in the system by the number of presentations available.

Both sets are analyzed by the system and useful information is obtained for the author, such as the knowledge subdomains captured in the presentations in the P+ set, or the experience interval (in the subject or in navigation) that includes all the intervals that label a learning structure defined upon a presentation in P-. The first item permits the author to identify subdomains of large collective interest, while the second enables detection of the lack of CS_L based on CS_P for users with a given degree of experience. For example, a low number of visits in a presentation could be justified if the author has defined only CS_L for users with a ["medium" - "total"] experience of the subject and the majority of system users are novices.

3.2 Memorization Transition Matrix

In addition, by combining the TM_P of all the presentations of a CS_M, the system can build the transition matrix of the CS_M the users have in mind and describe using their navigation pattern (Figure 8). Upon the author's request, the system will infer the differences between the **real CS_M**, created by the author in the memorization phase, and the **virtual CS_M**, defined through navigation by the users browsing the different presentations of the real CS_M.

The transition matrix that represents the virtual CS_M (TM_M_users) is built by adding all the TM_P, each of which is associated to a CS_P defined from the real CS_M. This matrix will have a row and a column for each concept in the real CS_M. Since a presentation will not usually include all the concepts in the CS_M, in the calculation of the cell TM_M_users[ci,cj] only the transition matrices TM_P that include both concepts (so that they have the cell TM_P[ci,cj]) will be used. There may be no CS_P that includes both concepts ci and cj, and in this case the cell TM_M_users[ci,cj] is marked with a special symbol (∝). Otherwise, the value for the cell TM_M_users[ci,cj] represents the number of times a user of any CS_P has followed the conceptual relation ci → cj.

200	Loto	Ha	Asana	Hatha
Loto	5	0	0	0
Ha	0	0	90	5
Asana	0	0	0	0
Hatha	0	80	15	5

100	Ha	Hatha	Tha
Ha	0	0	0
Hatha	70	5	0
Tha	25	0	0

TM_P

TM_M_users

300	Loto	Ha	Asana	Hatha	Y-Y	Tha
Loto	5	0	0	0	∝	∝
Ha	0	0	90	5	∝	0
Asana	0	0	0	0	∝	∝
Hatha	0	150	15	10	∝	0
Y-Y	∝	∝	∝	∝	∝	∝
Tha	∝	25	∝	0	∝	0

Figure 8. Creating the TM_M_users matrix

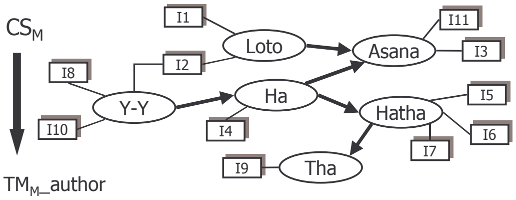
Since the sum of the cells in a column in the TM_M_users shows the number of visits to the concept labelling the column, it can be used to detect the concepts that may be removed (those with a much lower number of visits than the rest).

The ∝ symbol enables the relations ci → cj that have not been followed because they do not match the user's mental schema to be distinguished from the ones that are just impossible. In the first case, the value of TM_M_users[ci,cj] is very low (for example Asana → Loto), while in the second case the value of TM_M_users[ci,cj] is the mark ∝ (for example Asana → Tha). Forgotten concepts can also be detected since the concepts not included in any presentation have the ∝ symbol in all the cells in their rows and columns (for example Y-Y).

3.3 Comparing the Virtual CS_M with the Real CS_M

The transition matrix that represents the real CS_M (TM_M_author) has the value 1 in the cell TM_M_author[ci,cj] if there is a conceptual relation ci → cj in the real CS_M. The other cells have the value 0 (Figure 9).

In order to turn TM_M_users into a matrix comparable with the TM_M_author, the value in each cell TM_M_users[ci,cj] must be transformed into zero or one. For this process, the diagonal is discarded since although the obtained value in the cells TM_M_users[ci,ci] will often be one, this result does not normally correspond with the conception of a cyclic relation in the user's mind, but with the intention of obtaining more information about the current concept by visiting other items associated to it**.**



	Loto	Ha	Asana	Hatha	Y-Y	Tha
Loto	0	0	1	0	0	0
Ha	0	0	1	1	0	0
Asana	0	0	0	0	0	0
Hatha	0	0	0	0	0	1
Y-Y	0	1	0	0	0	0
Tha	0	0	0	0	0	0

Figure 9. The TM_M_author

The transformation is therefore carried out on the cells TM_M_users[ci,cj] that have not been marked as impossible (with a value which is different from ∝), and which are not in the diagonal (i different from j). The transformation process can be divided into two phases: firstly, the relative frequency with which the relation ci → cj has been followed by the users is obtained, and secondly, the value of the cell TM_M_users[ci,cj] is set to 0 or 1 depending on whether the estimated frequency is low or high.

The frequency is calculated in accordance with Equation 2 by dividing the cell's value by the theoretical average obtained when dividing the total number of visits (discarding the ones in the diagonal) by the number of existing relations (discarding the ones in the diagonal and the impossible relations). For example, Figure 10 shows how the frequency obtained for the relation Ha → Asana reflects a usage which is 5 times higher than the average.

$$\mathrm{TM_M\_users}\left[c_i, c_j\right] = \frac{\mathrm{TM}_M\_users\left[c_i, c_j\right]}{\left[\dfrac{\sum_{i=1}^{n}\sum_{j=1}^{n}\mathrm{TM}_M\_users\left[c_i, c_j\right] - \sum_{i=1}^{n}\mathrm{TM}_M\_users\left[c_i, c_i\right]}{(n\times n)-n-x}\right]}$$

with *n* being the number of concepts in CS_M and *x* the number of cells marked with ∝ in TM_M_users **(2)**

Theoretical average = (300-15) / (36-6-14) ≈ 17,8

TM_M_users′

	Loto	Ha	Asana	Hatha	Y-Y	Tha
Loto	♦	0	0	0	∝	∝
Ha	0	♦	5	0,28	∝	0
Asana	0	0	♦	0	∝	∝
Hatha	0	8,4	0,84	♦	∝	0
Y-Y	∝	∝	∝	∝	♦	∝
Tha	∝	1,4	∝	0	∝	♦

Figure 10. Creating the TM_M_users' matrix

Finally, the value of the cell TM_M_users''[ci,cj] is set to 1 if the obtained frequency exceeds a given threshold set by the author, or to 0 if not. The default threshold is again 1, which means that the users are assumed to conceive a relation ci → cj if they have followed it more times than the theoretical average (Figure 11).

Threshold = 1

TM_M_users″

	Loto	Ha	Asana	Hatha	Y-Y	Tha
Loto	♦	0	0	0	∝	∝
Ha	0	♦	1	0	∝	0
Asana	0	0	♦	0	∝	∝
Hatha	0	1	0	♦	∝	0
Y-Y	∝	∝	∝	∝	♦	∝
Tha	∝	1	∝	0	∝	♦

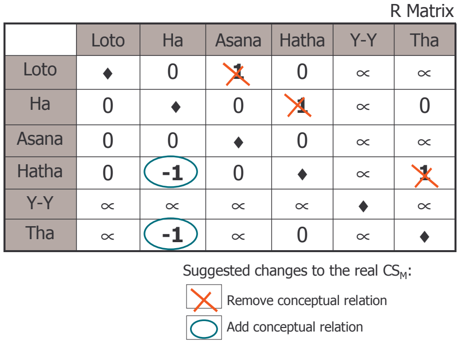
Figure 11. Creating the TM_M_users'' matrix

3.4 Suggestions Given to the Author

In order to compare both matrices, the system calculates the difference matrix: R = TM_M_author – TM_M_users''. The analysis of differences between the real and virtual CS_M is performed using the R matrix, ignoring the diagonal cells and the cells marked with ∝. When these cells are excluded, all the values in the R matrix will be 0, 1 or -1.

- A cell R[ci,cj] with value 0 indicates that the real CS_M and the virtual CS_M agree in the existence or non-existence of the relation ci → cj.
- A cell R[ci,cj] with value 1 will be notified to the author, denoting that most users do not use the conceptual relation ci → cj, which appears in the real CS_M.
- A cell R[ci,cj] with value -1 will be notified to the author, indicating that most users follow the conceptual relation ci → cj, which is not in the real CS_M.

Figure 12 shows the analysis performed with the R matrix obtained from the matrices in Figures 9 and 11. It should be observed, for example, that the system suggests to the author the creation of a relation from the concept Hatha to the concept Ha, and the removal of the existing relation between Loto and Asana.



R Matrix

	Loto	Ha	Asana	Hatha	Y-Y	Tha
Loto	♦	0	~~1~~	0	∝	∝
Ha	0	♦	0	~~1~~	∝	0
Asana	0	0	♦	0	∝	∝
Hatha	0	-1	0	♦	∝	~~1~~
Y-Y	∝	∝	∝	∝	♦	∝
Tha	∝	-1	∝	0	∝	♦

Suggested changes to the real CS_M:
✗ Remove conceptual relation
○ Add conceptual relation

Figure 12. R matrix for the matrices in Figures 8 and 11

3.5 Evolution of the CS_M

The suggestions provide the author with useful information for matching the author's representation models with the mental models of most users browsing the hypermedia system.

Nevertheless, as we commented before, the author will ultimately decide whether to follow all, some or any of the suggestions obtained at run-time. If the author decides to carry out a change in the structural design, the author tool provides a set of evolutionary actions for this.

Each evolutionary action takes care of a specific modification and has an associated set of preconditions that must hold in order to be performed. In addition, its execution can trigger an automatic **change propagation** process that is responsible for leaving the modified subsystem and the affected subsystems in a consistent state.

For example, the following events happen when the author decides to remove the conceptual relation Loto → Asana:

1. The conceptual relation between Loto and Asana in the CS_M Yoga (Figure 2) is removed.
2. The concept Loto becomes disconnected so it is removed from the CS_M, along with its functional relations.
3. When removing the functional relation between Loto and item I1, the second becomes disconnected and so it is also removed from the CS_M (although it is stored for future use).
4. Because of the changes in the CS_M, in the CS_P Yoga-1 (Figure 3), the concept Loto, the functional relation from it to the concept Asana, their functional relations, and items I1 and I2 are hidden.
5. The changes in the CS_P Yoga-1 affect the structures created from it, more specifically they affect the order, update and knowledge rules related to items I1 and I2.

4. CONCLUSIONS AND RELATED WORK

In this paper, we have briefly described the background to adaptive hypermedia systems, outlining some of the problems that currently arise and consequently proposing the SEM-HP model. This model has been presented as a systemic, semantic and evolutionary model for the development of adaptive hypermedia systems.

- Systemic because it conceives the hypermedia system as structured on four interrelated subsystems: the Memorization Subsystem, which is in charge of storing, structuring and maintaining the knowledge domain of the hypermedia system; the Presentation Subsystem, used by the author to define partial views of the knowledge domain; the Navigation Subsystem, that allows the order in which the information can be browsed to be established; and the Learning Subsystem, which takes care of the user adaptation.
- Semantic because it makes explicit (by means of navigable conceptual structures) the meaning of the relations between the concepts in the knowledge domain and the function of the information contained in the items associated to them.
- Evolutionary because it supports specific mechanisms (meta-system, evolutionary actions, change propagation, etc.) that allow the author to perform changes in the hypermedia system’s structure easily and so that consistency is maintained.

We have also outlined the adaptive capabilities of the system, referring to the management of an individual user model that is updated while the user navigates, and which captures (among other features) the user's knowledge state. This knowledge state is employed to distinguish the information users are ready to understand from the information they are not, and to consequently restrict their navigation, or to generate a guided route that allows them to achieve a given knowledge goal.

Finally, we have described the Adaptation by Feedback method in greater detail. The main objective of this method is to identify the differences between the representation of the knowledge domain developed by the author (that is, the CS_M) and the mental conception held by the group of users browsing the system. It also allows the author to be informed of how the users use the structures they are provided with for navigation (for example, to distinguish superfluous presentations or concepts from those that are widely used), and also of some features of these structures (the existence of concepts not included in any presentation, the subdomains of the most visited presentations, the experience interval that covers an under-used presentation, etc).

Focusing on the Adaptation by Feedback method, there are different systems in the literature that use a model of the group of users to adjust the application to a particular user, which include recommender systems such as the one described in [17]. In addition, other authors provide automatic or semiautomatic mechanisms to match the navigation structures and the users’ mental models. Among these, we can highlight the approaches proposed by Bollen [11] and Casteleyn et al [18].

Bollen’s approach also uses the knowledge from the group of users in order to benefit individual users. This approach rests on the belief, which we share, that people generally have stable and predefined ideas about the associations between concepts and that these ideas overlap among the group of individuals. Designers use their ideas of association to link the hypermedia pages, and the users use theirs in order to determine how to navigate the pages. Therefore, if there is not much overlap between the user’s and the designer’s models, the interaction between the user and the system will be inefficient.

Even if the objective of Bollen’s proposal and ours is the same, the way in which it is developed varies significantly. Regarding the way in which the conceptual relations accepted by the users are identified, SEM-HP uses transition matrices, while Bollen’s approach applies frequency, transitivity and symmetry rules. Concerning the usefulness of the identified relations, SEM-HP aims to create or delete associations between the concepts in the CS_M, while Bollen’s approach attempts to order the links provided so that the most “relevant” are placed first. In relation to who restructures the hypermedia network so that it reflects the identified relations, SEM-HP gives the final responsibility to the author, while in Bollen’s approach the hypermedia network is automatically restructured.

Casteleyn et al’s approach analyzes the users’ navigation in order to identify missing or superfluous information. Our approach focuses on identifying missing or superfluous relations between concepts, as well as superfluous items and concepts.

Casteleyn et al’s approach analyzes the navigation for each class of users (audience class). In our approach, it can be understood that the analysis is also carried out for each user class if we define the user class as the group of users that use the same CS_P (this set of users will have similar features since the CS_L is chosen according to the user's features). In addition, we combine the information retrieved about each class of users in order to perform a global analysis, which improves the system's structure from a common perspective.

Similarly, there are common points in the way in which we analyze the information gathered about the users' navigation. In both cases, we use matrices to represent this information, but while Casteleyn et al's approach stores the number of visits that each class of users has performed to each information requirement, we work with a transition matrix in order to analyze the number of visits made to any information associated to a concept, taking into account the previously visited concept. In both cases, we use a threshold to define the meaningful values in the analysis, and in both cases it is based on statistical values. Nevertheless, in Casteleyn et al's approach they adjust the average using the median absolute derivation, and in our case we allow the author to adjust it.

Regarding user adaptation, Casteleyn et al's approach performs an automatic adaptation by copying or linking the missing information in a navigation track, and by placing the links to superfluous information in the final places. We present the analysis results to the author who can consequently then reorganize the structure.

5. FURTHER WORK

Our work currently focuses on finishing and refining the implementation of the JSEM-HP author tool, which is still a prototype, in order to practically validate the SEM-HP model in several domains where we have already conceptually verified its usefulness: e-learning (which is its main application) and other applications such as the integration of children with autism problems [19]. Regarding the described adaptation method, the tool will allow us to check its usefulness in large scale cases.

6. ACKNOWLEDGMENTS

This work has been partially funded by the Spanish MCYT's R&D project ADACO TIN2004-08000-C03-02.

7. REFERENCES

[1] Bush, V. As We May Think. *The Atlantic Month.* Vol.: 176. Pp: 101-108. July, 1945.

[2] Engelbart, D. A Conceptual Framework for the Augmentation of Man's Intellect. *Information Handling, Spartan Books*. 1963.

[3] Nelson, T. H. Getting in Out of Our System. *Information Retrieval: A Critical Review, Thompson Books*. Washington D.C. 1967.

[4] Shneiderman, B.; Kearsley, G. Hypertext Hands-On!: An Introduction to a New Way of Organizing and Accessing Information. *Addison Wesley*. 1989.

[5] Campbell, B.; Goodman, J.M. HAM: A General Purpose Hypertext Abstract Machine. *CACM.* Vol.: 31(7). Pp: 856-861. July, 1988.

[6] Stotts, J.B.; Furuta, R. Programmable Browsing Semantics in Trellis. *Hypertext'89 Conference*. Pp: 27-42. 1989.

[7] Halasz, F.; Schwartz, M. The Dexter Hypertext Reference Model. *NIST Hypertext Standardization Workshop*. Pp: 95-133. February, 1990.

[8] De Bra, P.; Houben, G.J.; Wu, H. AHAM: A Reference Model to Support Adaptive Hypermedia Authoring. *InfWet'98 Conference*. 1998.

[9] De Bra, P.; Calvi, L. AHA: a Generic Adaptive Hypermedia System. *2nd Workshop on Adaptive Hypertext and Hypermedia*. Pp: 5-12. Pittsburgh. 1998.

[10] Koch, N.; Wirsing, M. The Munich Reference Model for Adaptive Hypermedia Applications. *2nd International Conference on Adaptive Hypermedia and Adaptive Web-based Systems*. LNCS 2347 © Springer Verlag. Pp: 213-222. May, 2002.

[11] Bollen, J.; Heylighen, F. A System to Restructure Hypertext Networks into Valid User Models. *The New Review of Hypermedia and Multimedia*. Vol.: 4. Pp: 189-213. 1998.

[12] García, L.; Rodríguez, MªJ.; Parets, J. Evolving Hypermedia Systems: a Layered Software Architecture. Journal of *Software Maintenance and Evolution: Research and Practice*. John Wiley & Sons, Ltd. Vol.: 14(5). Pp: 389-405. 2002.

[13] Medina, N.; Molina, F.; García, L. Diversity of Structures and Adaptive Methods on an Evolutionary Hypermedia System. Journal *IEE Proc.-Software*. ISSN: 1462-5970. Vol.: 152 (3). Pp: 119-126. 2005.

[14] Medina, N.; Molina, F.; García, L.; Rodríguez, MªJ. Coevolution of Models of an Adaptive Hypermedia System. *IDPT'03. © SDPS*. ISSN: 1090-9389. Vol.: 2. Pp: 18, 38-45. 2003.

[15] Brusilovsky, P. Methods and Techniques of Adaptive Hypermedia. *User Modeling and User-Adapted Interaction.* Vol.: 6. Pp: 87-129. 1996.

[16] Medina, N.; Molina, F.; García, L.; Parets, J. Personalized Guided Routes in an Adaptive Evolutionary Hypermedia. *9th International Conference on Computer Aided Systems Theory*. LNCS 2809 © Springer Verlag. ISSN: 0302-9743. Pp: 196-207. February, 2003.

[17] Ardissono, L.; Goy, A.; Petrone, G.; Segnan, M.; Torasso, P. Tailoring the Recommendation of Tourist Information to Heterogeneous User Groups. *3rd Workshop on Adaptive Hypertext and Hypermedia.* Aarhus, Denmark. August, 2001.

[18] Casteleyn, S., Garrigós, I., De Troyer, O.: Automatic Runtime Validation and Correction of the Navigational Design of Web Sites, In *Web Technologies Research and Development - APWeb 2005*, pp. 453 - 463. Springer-Verlag, ISBN 3-540-25207-X, Shangai, China. 2005.

[19] Gea, M.; Medina, N.; Rodríguez, MªL.; Rodríguez, MªJ. Sc@ut: Platform for Communication in Ubiquitous and Adaptive Environments Applied for Children with Autism. *8th ERCIM Workshop. User Interfaces For All*. LNCS 3196 © Springer. ISSN: 0302-9743. Pp: 50-67. June, 2004.