Proceedings of the Fourth International Conference on User Modeling, Hyannis, MA, 99-105, 1994.

# KN-AHS: An Adaptive Hypertext Client of the User Modeling System BGP-MS

**Alfred Kobsa, Dietmar Müller, Andreas Nill**
Dept. of Information Science, Univ. of Konstanz
P.O. Box 5560-D73, 78434 Konstanz, Germany
{kobsa,mueller,nill}@inf-wiss.uni-konstanz.de

**Abstract[1]**

This paper describes the automatic adaptation of hypertext to the user's presumed domain knowledge in the KN-AHS system, and the support that the user modeling shell system BGP-MS can provide for this adaptation. First, basic hypertext concepts will be introduced and reasons given for why hypertext should adapt to the current user (especially to his/her state of knowledge). A brief overview of those representation and inference components of BGP-MS that are used by KN-AHS will then be provided, followed by a description of its adaptive user interface. The interaction between the adaptive hypertext system and the user modeling system will be investigated in detail based on a possible dialog with a user. Finally, the inter-process communication between KN-AHS and BGP-MS will be described and related work discussed. The aim of this work was to demonstrate the feasibility of user modeling with BGP-MS in a "normal" hardware and software environment that is frequently found in the workplace.

## 1. Hypertext and adaptive hypertext

### 1.1 Hypertext

Hypertext consists of any number of objects[2] that can be linked with one another in a network structure. Therefore, hypertext is not necessarily read in linear (i.e., sequential) order like conventional text, but can be read in a non-linear order by navigating within the hypertext node network. This non-linear linkage of objects represents the basic characteristic of hypertext [cf. Seyer 1991, Kuhlen 1991].

Gaining information in a non-linear form is not new (see e.g. information search in encyclopedias). As opposed to printed media, however, the representation of information in electronic form allows the user to directly and comfortably traverse contextual connections. Hypertext has therefore been able to enjoy an increased importance in the last few years, especially as a basis for on-line help systems and electronic encyclopedias.

The user-friendliness of many hypertext systems is rooted in their usage of intuitively understandable direct-manipulative interfaces [cf. Kuhlen 1991, Shneiderman & Kearsly 1989]. The user has the possibility of directly manipulating graphical objects with a pointer (e.g. a mouse) without having to use complicated commands. The effect of these actions can be seen immediately on the screen. Direct manipulation can be used, for example, to reach a different node from the node currently shown on the screen. The usual graphical objects used in such navigating operations include mouse-sensitive text passages (hotwords) or buttons (more on this in Section 3). Other important hypertext components are glossaries, indices, and graphical representations of the hypertext structure, which all offer important meta-information about the basic text objects. Recent research supplements associative navigation by controlled navigation and by search techniques from the field of Information Retrieval, in order to increase the search efficiency [cf. Kuhlen 1991].

### 1.2 Adaptive Hypertext

Two major problems arise when working with hypertext systems:

• *Orientation and navigation problem*s: When navigating in hypertext, users are frequently uncertain as to how to reach their goals. Since users can choose any course within hypertext, they run the risk of losing their orientation. Navigation aids that take users' goals into account may be helpful. Kaplan et al. [1993] showed empirically that navigation suggestions based on knowledge about the objectives and the navigation behavior of previous users as well as the goals of the current user can significantly accelerate the current user's search for information.

• *Comprehension problem*s: Since hypertext is frequently read by users with differing knowledge and experience levels, it may at the same time be too difficult and too detailed for laypersons, and too redundant for experts. Boyle & Encarnacion [1993] showed empirically that an automatic adaptation of hypertext to the user's state of knowledge significantly improves text understanding as well as partially improving search speed.

The system KN-AHS[3] deals with the second problem and adapts hypertext objects to the current user's state of knowledge. In contrast to other adaptive hypertext systems, the realization of KN-AHS took advantage of existing software products. TOOLBOOK [Asymetrix 1989], a widely available hypertext shell system, offered us a powerful tool for the implementation of the hypertext and its user interface. The user modeling shell system BGP-MS offered a wide variety of representation and inference possibilities that ensure flexible adaptation. Both tools run as independent software systems on a PC platform and interact via inter-process communication.

---

[1] This work was supported by the German Science Foundation (Grant Ko-1044) and the Univ. of Konstanz (Grant AFF 17/92).

[2] The objects in a hypertext base are not only text documents but can also include non-textual data (tables, graphs, animation, etc.). If an audiovisual component is involved, then the term 'hypermedia' is used.

[3] KN-AHS stands for KoNstanz Adaptive Hypertext System.

## 2. User modeling using BGP-MS

Over the last few years, researchers have tried to develop so-called 'user modeling shell systems', since programming user modeling components in application systems is very time-consuming. By using these shell systems, integrated mechanisms and methods are made available which are often needed in user modeling components.

One of these shells is BGP-MS[4], which is currently under development. From the perspective of the application system, BGP-MS can be regarded as a "black box" that receives information about the user and answers questions posed by the application system concerning current assumptions about the user. In order to realize its adaptive dialog behavior that is oriented on the user's state of knowledge, KN-AHS utilizes certain services of BGP-MS. The following sections will gradually present the BGP-MS components that are used, and explain their functionality with regard to the adaptation of hypertext by means of examples[5].

### 2.1 Communication between the application and BGP-MS

The user modeling component of KN-AHS is not integrated into the application (as is the case for user modeling components in virtually all other user-adaptive systems), but is rather an independent process that communicates with the application. Observations based on user actions will be reported to BGP-MS by the hypertext system (cf. Fig. 1, part a). The application can ask BGP-MS questions about the user (b) and BGP-MS can in return report its current assumptions concerning the user (c).

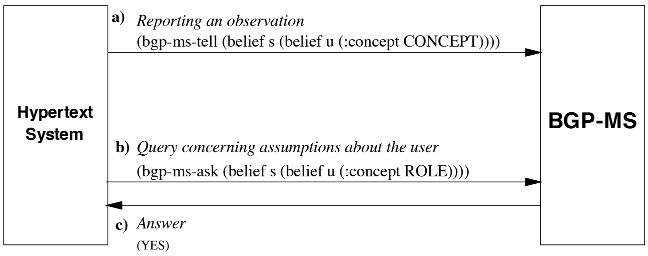


*Fig. 1: Communication between the application and BGP-MS*

### 2.2 Partitions for collecting different types of assumptions

BGP-MS utilizes the partition mechanism SB-PART [Scherer 1990], which allows different types of assumptions about the user to be represented simultaneously, but still separately. These assumptions include, for example, assumptions concerning the user's knowledge or goals, assumptions concerning application-relevant characteristics of user subgroups (so-called 'stereotypes'), or the domain knowledge of the user modeling component.

Partitions can be ordered in an inheritance hierarchy, where subordinate partitions inherit the contents of superordinate ones. Fig. 2 shows a simple partition hierarchy, as is currently used in our adaptive hypertext. The depicted partitions can be divided into three groups:

- The *individual user model* consists of the partitions SBUB (System Believes User Believes), which contains BGP-MS's assumptions about the user's knowledge, and SB¬UB, which contains BGP-MS's assumptions about what the user does not know.

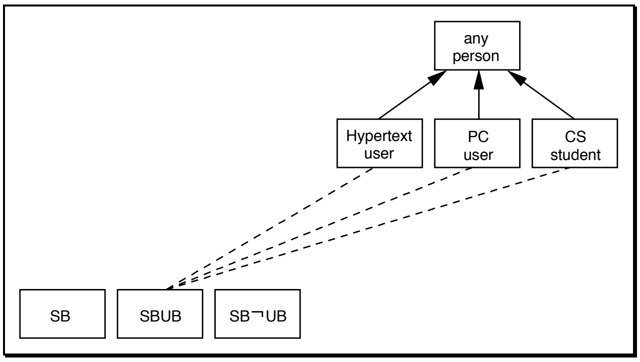


*Fig. 2: Partition hierarchy in SB-PART*

- The *stereotypes* for user subgroups are ordered hierarchically. We assume for our application that the stereotype "any person" is available**,** which includes only general information, i.e. knowledge available to any user. All other stereotypes include typical characteristics of users with various fields of specialization, namely hypertext users, PC users, and computer science students[6]. They inherit the contents from the general stereotype.
- The *domain knowledge* in BGP-MS, which is included in the partition SB (= System Believes).

### 2.3 Stereotype mechanisms

BGP-MS allows the user model developer to define so-called 'stereotypes' that contain application-relevant characteristics of user subgroups. The programmer can also define the conditions under which a user will be assigned to these subgroups, and those under which an existing assignment should be withdrawn. BGP-MS contains a stereotype managing mechanism that analyzes observations received from the application and checks the activation and retraction conditions of all stereotypes. It will then enter

---

[4] BGP-MS stands for Belief, Goal and Plan Maintenance System [Kobsa 1990].

[5] A more comprehensive description of BGP-MS can be found in [Kobsa & Pohl, 1994].

[6] This type of stereotype hierarchy -- a kernel with several specializations -- has been investigated several times in connection with the use of UNIX commands [Hanson et al. 1984, Sutcliffe & Old 1987]. The model is also referred to as 'lettuce model' because of its graphical form in Venn diagrams [Kobsa 1990].

inheritance links between the individual user model and those stereotypes that become active, and delete links to stereotypes that become deactivated. More than one stereotype can be active at the same time, if allowed by the user model programmer. He/she can also define the frequency of stereotype revision.

The broken lines in Fig. 2 represent the possible inheritance relationships with the stereotypes hypertext user, PC user and CS student. Since only “positive knowledge” is contained in the stereotypes of KN-AHS at the time being, a connection to SB¬UB is not possible. Several stereotypes can be active simultaneously, since the readers (such as all authors of this paper) can be both hypertext and PC users. On the other hand, an existing connection can be withdrawn if observations are made that meet the retraction conditions of the stereotype.

## 2.4 Representing domain knowledge in BGP-MS

So far we have described the organization of system knowledge, user assumptions and pre-defined stereotypes using separate partitions and inheritance links. Now we take a closer look at the contents of individual partitions and their representation. One of the knowledge representation languages used within partitions is SB-ONE [Profitlich 1989, Kobsa 1991], which belongs to the family of KL-ONE type languages. The following simplified description of the language elements is sufficient for the purposes of this paper.

The two most important representational elements in KL-ONE type languages are concepts (depicted as ovals in Fig. 3) and role relationships between concepts (depicted as small circles). For the purposes of KN-AHS, two types of concepts are distinguished: field concepts, which represent small fields of knowledge, and terminological concepts, which represent technical terms. In Fig. 3, ‘UM-Shell’, ‘GUMS’, ‘BGP-MS’, ‘KL-ONE’, ‘SB-ONE’, ‘CLASSIC’ and ‘BACK’ are field concepts. ‘ROLE’ and ‘CONCEPT’ are terminological concepts. The role ‘associated-concept’ (the only role used in KN-AHS) defines the relationship between fields and their associated terminology.

Super- and subordinations are also found for concepts, where the subordinated concept inherits all role relationships of the superordinated concept. The field concepts ‘SB-ONE’ and ‘BGP-MS’ in Fig. 3 inherit all roles of the field concept ‘KL-ONE’, and thereby all associated terminological concepts.

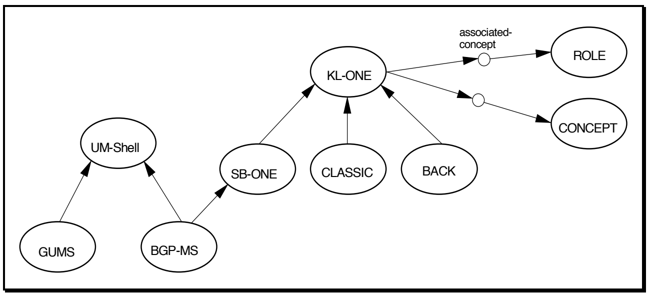


*Fig. 3: Detail of the concept hierarchy in the partition ‘System Believes’ (SB)*

## 2.5 User model acquisition and inferences

Messages that an application communicates to BGP-MS may express various types of information about the user’s knowledge. KN-AHS only takes assumptions about the user’s conceptual knowledge into account when adapting hypertext documents. Its messages to BGP-MS are therefore restricted to information on whether the user is familiar or unfamiliar with certain concepts. These “primary assumptions” about the user become entered into the partitions SBUB and SB¬UB, respectively. Primary assumptions will be compared with all stereotype activation and deactivation conditions in regular pre-set intervals. As an effect, inheritance links between stereotype partitions and the partition SBUB may be entered or erased.

The user model developer may also define inference rules that become executed after each new entry into the individual user model. The inferences used in KN-AHS are based on domain knowledge that is represented in SB. They take sub- and superfield relationships and the “associated-concept” relationship between fields and their respective terminology into account, and comprise the following rules:

a) Sub- and superfield relationships
   - a1) If a minimum percentage P1 of direct subfields of a field were reported to be known/unknown, then all its subfields are known/unknown.
   - a2) If a minimum percentage P2 of direct subfields of a field were reported to be known/unknown (where P1 can be different from P2), then the superfield is also known/unknown.

b) Relationships between fields and their respective terminology
   - b1) If a minimum percentage P3 of the terminological concepts of a field were reported to be known/unknown, then all terminological concepts of the field are known/unknown.
   - b2) If a minimum percentage P4 of the terminological concepts of a field were reported to be known/unknown (where P3 can be different from P4), then the field is also known/unknown.

Conflicts can arise between the observations made by the application and assumptions inferred in BGP-MS. If this is the case, then the inferred assumptions will have a lower priority and will be discarded from the partition. Dependency management between premises and consequences will not be able to be considered until sometime in the future.

# 3. The adaptive hypertext system KN-AHS

In this section, we will first describe the functionality of the user interface of KN-AHS. We will focus especially on reviewing the direct-manipulative actions that are available to the user. Then the assumptions will be described that KN-AHS draws about the user and reports to BGP-MS, as well as the kind of hypertext adaptation that it performs

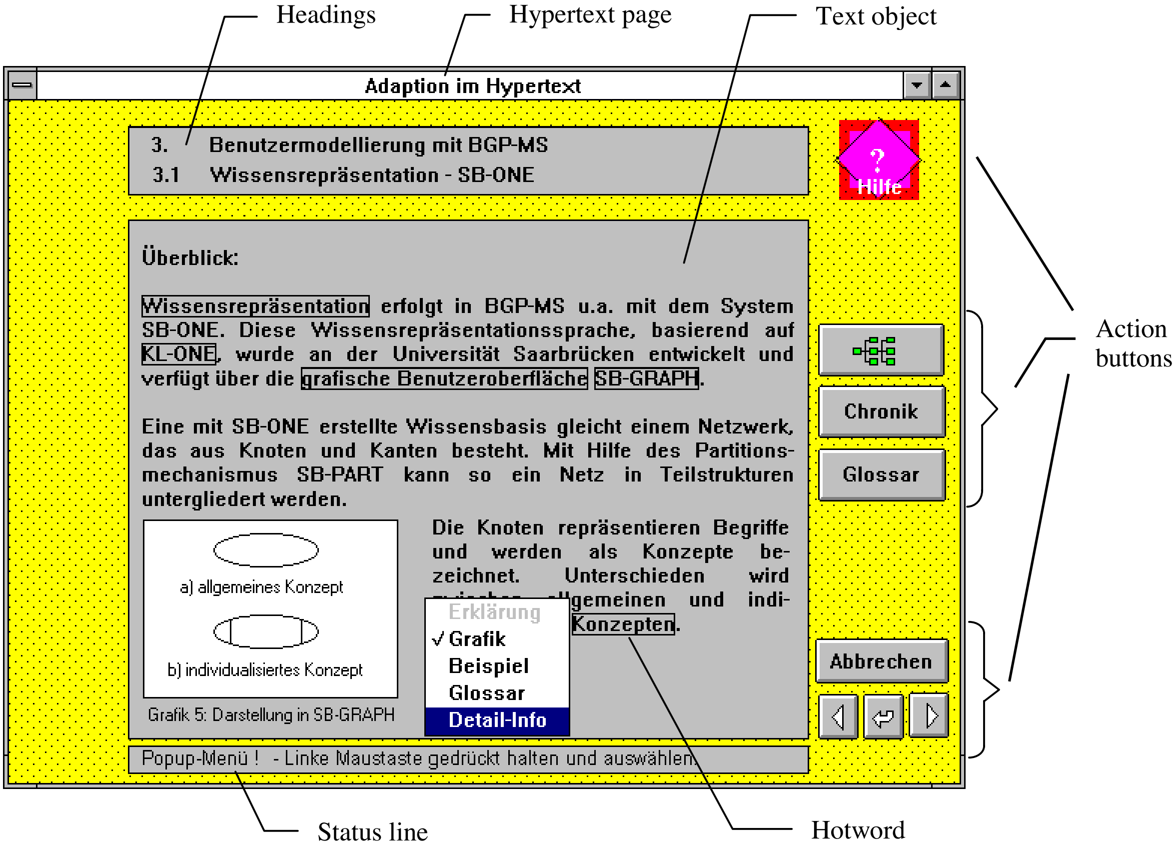


*Fig. 4: Selecting additional information on a hotword*

after consulting BGP-MS on the presumable conceptual knowledge of the user. Finally, the adaptive behavior of KN-AHS as well as its interaction with BGP-MS will be illustrated by an example of a possible user dialog.

## 3.1 The user interface

Fig. 4 shows an example of the user interface of KN-AHS. Special attention was paid to awakening interest and curiosity, and to stimulate users to navigate through the hypertext. It was also important that the interface should be largely self-explanatory. Users should be able to correctly predict the outcome of each possible action. This is especially important if assumptions about their knowledge are formed on the basis of their actions.

In a simplified form, the hyperdocument can be divided into four areas:

• *Headings*: This box informs the users of their whereabouts (i.e., the current chapter).

• *Text object*: This area is the core of the hypertext page. The reader can receive information in the form of text and graphics (if the text object is longer than a screen page, then a vertical scroll bar will be automatically inserted). So-called ‘hotwords’ can be found within this area, i.e. mouse-sensitive text passages such as the boxed text elements ‘Wissensrepräsentation’ (knowledge representation), ‘KL-ONE’, ‘Konzepten’ (concepts) etc. in Fig. 4. The actions that the user can perform on these hotwords will be described further below.

• *Status line*: The status line is found on the lower screen edge below the text object. It offers the user additional information about possible actions for each hotword.

• *Action buttons*: These are found on the right side of the screen. Starting here, the user can “jump” to other areas within the hypertext, for example (from top to bottom)

- to a context-sensitive help text,
- to a graphically represented table of contents,
- to the dialog history, which includes a list of the already-viewed text objects to which the user can return on demand,
- to the glossary, which is context-independent.

Other buttons are also available that enable the reader to jump to a previous or following page, as well as to the front-runner page of the one depicted on the screen.

When the user moves the mouse cursor across a hotword, the form of the mouse cursor indicates the available actions for the hotword. The user may (a) jump to another text object that provides detailed information on the hotword; or (b) request additional information with a pop-up menu, namely an explanation, a graphic, an example, a glossary definition, or additional details. Table 1 lists in greater detail the additional information that is available in KN-AHS via this pop-up menu. An example can be found in Fig. 4, where the user clicked on the hotword ‘Konzepten’.

Some kinds of additional information may not be available for a hotword; the respective menu entries then turn to grey in the pop-up menu (like 'Erklärung' in Fig. 4).

| Menu entry | Effect |
|---|---|
| *Erklärung* (Explanation) | Additional information which explains the hotword *in context* will be inserted near the hotword to ensure a terminologically supportive effect. In some cases the hotword may also become replaced by a simpler term or expression. |
| *Grafik* (Graphic) | A graphic appears which should illustrate the hotword. |
| *Beispiel* (Example) | An example will be displayed that should clarify the hotword. |
| *Glossar* (Glossary) | The page of the glossary that contains the hotword will be displayed. A context-independent definition or description of the hotword can be found there. |
| *Detail-Info* (Detailed information) | Additional details related to the hotword will be inserted near the hotword. |

*Table 1: Effects of the options in the hotword pop-up menu.*

## 3.2 Drawing assumptions about the user

KN-AHS draws assumptions about the user's knowledge based on two information sources: namely an initial interview, and some of the hypertext actions which the user may perform.

In the initial interview, questions are posed to the user that refer to his membership in clearly separable user subgroups (like 'computer science student'), and his exposure to PCs, hypertexts, etc. The user's replies become communicated to BGP-MS, which can activate initial stereotypes for the user (see Section 2.3). If the user decides to skip this interview, BGP-MS will only activate the 'any person' stereotype.

Certain actions that the user may perform afterwards in the hypertext give rise to assumptions about his familiarity with individual concepts:

- If the user requests an explanation, a graphic, an example or a glossary definition for a hotword, then he is assumed to be unfamiliar with this hotword.
- If the user unselects an explanation, a graphic, an example or a glossary definition for a hotword, then he is assumed to be familiar with this hotword.
- If the user requests additional details for a hotword, then he is assumed to be familiar with this hotword.

With each hotword for which more information can be requested, the corresponding SB-ONE concept that represents this technical term in BGP-MS is associated. When KN-AHS draws an assumption about the user's familiarity with a hotword, KN-AHS notifies BGP-MS that the corresponding concept is known or unknown to the user. An example can be found in Fig. 1, in which KN-AHS informs BGP-MS that the user is familiar with the concept 'CONCEPT' after it made the assumption that the user is familiar with the hotword 'Konzepten'.

BGP-MS is also equipped with a component that draws assumptions about the user based on the user's actions [Pohl et al. 1994]. Instead of drawing assumptions itself and communicating them to BGP-MS, KN-AHS could therefore also inform BGP-MS about the actions that the user performed, and let BGP-MS draw the assumptions. However, since the concept names in BGP-MS must be anyway known to KN-AHS in order that it can ask BGP-MS about the user's familiarity with them (see Section 3.3), this option was not chosen in order to avoid redundancy.

## 3.3 Adapting the document based on the user's conceptual knowledge

When the user switches to a new text object, KN-AHS aims at adapting it to the user's presumed conceptual knowledge. For each hotword in the new text object, it asks BGP-MS about the user's familiarity with the corresponding SB-ONE concept. The hotword is then treated in the following way:

- If the user is unfamiliar with the associated concept, an explanation gets automatically added to the hotword. (The very same adaptation would take place if the user had selected the 'explanation' entry in the pop-up menu for this hotword). Also, an icon that symbolizes an available graphic for the hotword is placed near the hotword.
- If the user is familiar with the hotword, more details are automatically added after the hotword.
- If no information is available from BGP-MS concerning the user's familiarity with the hotword, then the hotword is not changed.

Possibly icons that signal the availability of examples and glossary information may be added in the future for hotwords that are unknown to the user. However, this may raise the danger of the hypertext becoming visually overloaded on the terminal screen.

## 3.4 An example of an adaptation step

In this section we present an example that shows how adaptation in hypertext can be performed based on reader actions. We will specifically concentrate on the interplay between hypertext components and the user modeling system.

The user in Fig. 4 would like to learn more about the hotword 'Konzept' and asks for 'Detail-Info' (detailed information). The displayed screen page will be expanded and the desired information will be shown. Because of this user action, the hypertext application reports to BGP-MS that the user is familiar with the associated concept 'CONCEPT' (see Section 2.4). BGP-MS enters the term in the partition SBUB, carries out inferences in the knowledge base and starts the stereotype mechanism in pre-set intervals (see section 2.3).

Fig. 5 shows another text object that the reader could possibly encounter later in the hypertext session. It is interesting to note that detailed information for the hotword

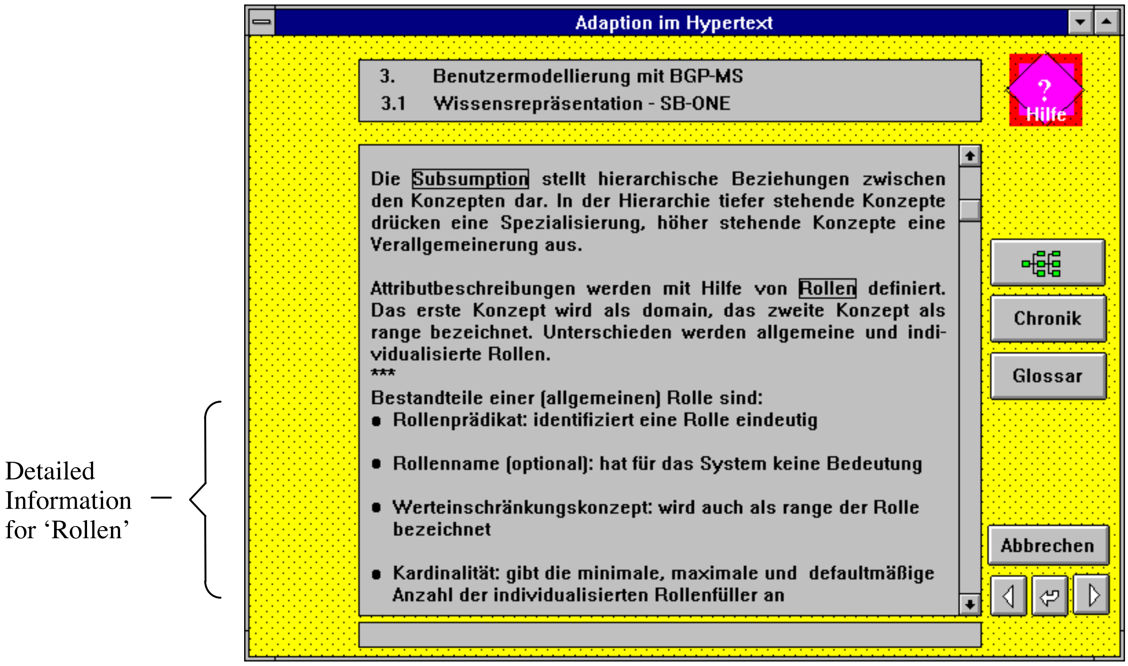


*Fig. 5: Automatic addition of detailed information for the hotword 'Rolle'*

'Rollen' is already provided, even though the user did not explicitly request more information.

How did that happen? Before switching pages the hypertext system asked BGP-MS whether or not the user was familiar with the expandable hotwords shown on the newly requested page. BGP-MS checked the individual user model and informed the hypertext application[7]. The concept 'ROLE' was reported as known (i.e., included in SBUB), and the hotword 'Rollen' was therefore automatically supplemented by detailed information. One reason why the concept 'ROLE' is contained in SBUB could be that rule b1 in section 2.5 fired after the concept 'CONCEPT' was entered in this partition.

## 4. Discussion and related work

The aim of this work was to demonstrate the feasibility of user modeling with BGP-MS in a "normal" hardware and software environment that is frequently used in the workplace. The basic architecture that we described -- application and user modeling system are independent processes that communicate via inter-process communication -- is quite unique. Only Orwant [1994] proposes a related framework (which is located on the level of a computer network however).

Some work already exists in the area of adaptive hypertext documents. The system that seems most closely related to KN-AHS is MetaDoc [Boyle & Encarnacion 1993]. It is also PC based and uses adaptation techniques that are similar to those in KN-AHS. Both its hypertext and its user-modeling component are "self-made". In comparison to MetaDoc, KN-AHS clearly profits from the greater expressiveness of the BGP-MS user modeling shell, which allows for the representation of hierarchically structured knowledge domains (instead of MetaDoc's flat "concept islands"), for inferences based on this hierarchy of knowledge domains, for more complex stereotype hierarchies than those of MetaDoc, and for more flexible stereotype activation rules. The adaptive range of KN-AHS goes somewhat beyond MetaDoc in that also graphics (and in the future possibly examples) become automatically included in the hypertext. A minor but noteworthy difference is that KN-AHS will only adapt the hypertext when the user switches to a *new* text object; it will never change a hotword in the current text as a result of an action that the user performed on a different hotword in the current text object since this seems to violate the constancy principle of software ergonomics.

Other related systems are ANATOM-TUTOR [Beaumont 1994] and ΥπADAPTερ [Böcker et al. 1990], which present hypertext-based tutorials on anatomy and Common Lisp, respectively. Both systems use self-made hypertext components. ANATOM-TUTOR contains a self-made user modeling component, while ΥπADAPTερ employs the MODUS user modeling shell [Schwab 1989], which is functionally included. Both systems are active tutorial systems and not only hypertext browsers like KN-AHS. They therefore have additional adaptive characteristics (e.g., ΥπADAPTερ exploits the user model for topic selection and presentation) and additional sources of information

[7] Possible answers are 'known' or 'unknown' (if the corresponding concept is included in SBUB or SB¬UB) as well as 'no statement possible' (if neither is the case).

about the user (e.g., ANATOM-TUTOR receives information about the user through a quiz). The adaptation of hypertext contents consists in adding or omitting information based on the assumptions about the user's knowledge (and also the user's preferences and learner type in the case of ϒπADAPTερ).

## 5. Hardware and Software Environment

KN-AHS has been implemented under MS-DOS 6.2 and MS WINDOWS 3.1 on a PC platform. TOOLBOOK 1.5, a popular hypertext shell system from Asymetrix Corporation has been used to construct the hypertext. The BGP-MS user modeling shell was developed (and will be further enhanced) in Common Lisp on SUN workstations. The relevant parts were ported to Golden Common Lisp 4.3, which also runs under MS WINDOWS.

The communication between the hypertext system and BGP-MS was realized using the inter-process communication system KN-IPCMS[8]. The current PC version exploits the DDE functionality[9] that is also supported by TOOLBOOK and Golden Common Lisp. KN-IPCMS is a platform-independent message-oriented communication protocol that allows both for synchronous and for asynchronous communication. In the interaction between KN-AHS and BGP-MS, observations made by KN-AHS will be transferred asynchronously. This means that KN-AHS and BGP-MS run concurrently, i.e. the user model management will largely be performed while the user is reading the current text object. Questions posed to BGP-MS will however be handled synchronously and will have priority over incoming observations.

## References


Asymetrix Corporation 1989. *Using TOOLBOOK®: A Guide to Building and Working with Books (Version 1.5)*. Washington.

Beaumont, I. 1994. User Modelling in the Interactive Anatomy Tutoring System ANATOM-TUTOR. *User Modeling and User-Adapted Interaction*. In press.

Boyle, C., and Encarnacion, A. O. 1993. An Adaptive Hypertext Reading System. *User Modeling and User-Adapted Interaction*. In press.

Böcker, H.-D.; Hohl, H. and Schwab, Th. 1990. ϒπADAPTερ*: Individualizing Hypertext*. In: D. Diaper et al., eds.: Human-Computer Interaction - INTERACT '90. North-Holland, Amsterdam, The Netherlands.

Hanson, S. J.; Kraut, R.E.; and Farber, J.M. 1984. Interface Design and Multivariate Analysis of UNIX command use. *ACM Transactions on Office Information Systems* 2(1):42-57.

Kaplan, C.; Fenwick, J.; Chen, J. 1993. Adaptive Hypertext Navigation Based on User Goals and Context. *User Modeling and User-Adapted Interaction* 3(3):193-220.

Kobsa, A. 1990. Modeling the User's Conceptual Knowledge in BGP-MS, a User Modeling Shell System. *Computational Intelligence* 6:193-208.

Kobsa, A. 1991. *Utilizing Knowledge: The Components of the SB-ONE Knowledge Representation Workbench*. In: J. Sowa, ed.: Principles of Semantic Networks: Exploration in the Representation of Knowledge. San Mateo, CA: Morgan Kaufmann.

Kobsa, A., and Pohl, W. 1994: The User Modeling Shell BGP-MS. *User Modeling and User-Adapted Interaction*. Submitted.

Kuhlen, R. 1991. *Hypertext. Ein nicht-lineares Medium zwischen Buch und Wissensbank*. Berlin: Springer.

Orwant, J. 1994. Heterogenous Learning in the Doppelgänger User Modeling System. *User Modeling and User-Adapted Interaction*. Submitted.

Pohl, W.; Kobsa, A.; and Kutter, O. 1994. User Model Acquisition Heuristics Based on Dialog Acts. WIS Report 6**,** Dept. of Information Sc., Univ. of Konstanz, Germany.

Profitlich, H. J. 1989. SB-ONE: Ein Wissensrepräsentationssystem basierend auf KL-ONE. Master Thesis, Dept. of Computer Science, Univ. of Saarbrücken, Germany.

Scherer, J. 1990. SB-PART: Ein Partitionsmechanismus für die Wissensrepräsentationssprache SB-ONE. Master Th., Dept. of Comp. Sc., Univ. of Saarbrücken, Germany.

Schwab, T. (1989): Methoden zur Dialog- und Benutzermodellierung in Adaptiven Computersystemen. Ph.D. Thesis, Dept. of Comp. Sc., Univ. of Stuttgart, Germany.

Seyer, P.C. 1991. *Understanding Hypertext: Concepts and Applications*. Blue Ridge Summit, PA: Windcrest Books.

Shneiderman, B.; Kearsley, G. 1989. *Hypertext hands-on! An Introduction to a New Way of Organizing and Accessing Information*. Reading, MA: Addison-Wesley.

Sutcliffe, A. G., and Old, A. C. 1987. Do Users Know They Have User Models? Some Experiences in the Practise of User Modeling. In: H.-J. Bullinger and B. Shackel, ed.: Human-Computer Interaction: INTERACT'87. North-Holland, Amsterdam, The Netherlands.


[8] KN-IPCMS stands for KoNstanz InterProcess Communication Management System.

[9] DDE (Dynamic Data Exchange) is a communication protocol under MS WINDOWS that defines how WINDOWS applications can exchange messages and data, and helps application programs communicate with one another, as long as they can support DDE as well.