In U. J. Timm and M. Rössel (Eds.), *ABIS-98, Adaptivität und Benutzermodellierung in interaktiven Softwaresystemen*. Erlangen, Germany: FORWISS, October 1998. (This paper is also available from http://www.cs.uni-sb.de/users/jameson/.)

Adapting to the User's Time and Working Memory Limitations: New Directions of Research

Anthony Jameson*
Department of Computer Science, University of Saarbrücken
P.O. Box 151150, D-66041 Saarbrücken, Federal Republic of Germany
`jameson@cs.uni-sb.de`
Fax: +49 681 302-4136, Phone: +49 681 302-2474

Abstract

As interactive computing devices become increasingly mobile, the user's time limitations and situational distractions are becoming important determinants of the quality of interaction with these devices. Automatic adaptation to such situational constraints is a challenging goal for user modeling research. This paper first briefly summarizes research in the first three-year phase of the READY project, in which a prototype of such a system was developed. It then discusses four new lines of research that will be pursued in this project during its second phase: extension to multimodal interaction; allowing for different degrees of granularity in the modeling of the user; learning the system's Bayesian networks automatically from empirical data; and dealing with more complex discourse structures.

Computing power is moving off of the desktop and into the hustle and bustle of everyday life. This trend generates new challenges and opportunities for those who develop adaptive systems. Users of a desktop computer may have to cope with deadlines and distractions, but on the whole they can measure their time in minutes and hours, rather than in seconds; and they can typically concentrate fully on whatever task they are using the computer for at the moment. By contrast, some of the more recent additions to the family of computing devices—such as hand-held computers and smart mobile phones—often constitute just one element in a complex environment that the user has to deal with. An interaction may be have to be completed within seconds, while the user is simultaneously conducting a conversation and/or navigating through unfamiliar territory. In such a situation, two critical factors that influence the success of interaction with the system are:

1. the amount of time that the user ($\mathcal{U}$) has available; and
2. the extent to which $\mathcal{U}$ can concentrate on the interaction, as opposed to dealing with situational distractions.

Somewhat more abstractly, we can speak of bottlenecks caused by temporary limitations of the user's *resources* of time and working memory.[1]

It is not obvious that the user's temporary resource limitations need to be recognized and adapted to automatically. First of all, systems of this sort are typically designed from the start in such a way that they can be used quickly and without much mental effort. But no single type

*The research outlined in this paper will be funded from 1999 though 2001 by the German Science Foundation (DFG) in its Collaborative Research Center on Resource-Adaptive Cognitive Processes, SFB 378, Project B2, READY. The principal investigators are Wolfgang Wahlster and Anthony Jameson. The complete text of the research proposal (Wahlster & Jameson, 1998a) is available on request. The project Web page is `http://w5.cs.uni-sb.de/~ready/`.

[1]The Collaborative Research Center on Resource-Adaptive Cognitive Processes, where the research discussed here is being conducted, investigates ways in which the performance of cognitive systems adapts to limitations in resources of various types (see `http://www.coli.uni-sb.de/sfb378/`).

of interaction is optimal for all configurations of resource availability. For example, if a system is designed for the worst case of a rushed and distracted user, it will probably make too little use of the user's resources when more of these are available.

What about the strategy of making these systems *adaptable* rather than *adaptive*? For example, the designers could give users the opportunity to choose, in each situation, how they would like to enter their input to the system and what type of output they desire. Unfortunately, a user who is rushed and/or distracted is in a poor position to adapt his system use accordingly—or even to signal the problem to the system.

In sum, it seems to be a reasonable goal to enable some types of interactive computing devices to do what people do when interacting with others: The device should assess, without being told, roughly what resources the person has available, and it should adapt its behavior accordingly.

To be sure, realizing such automatic adaptation is a challenging task, for the following reasons, among others:

1. The *evidence* that can permit a system ($\mathcal{S}$) to recognize resource limitations tends to be scant and not very reliable. A usefully accurate assessment is often only possible when several pieces of evidence can be combined.
2. The *temporal variability* of these resource limitations means that $\mathcal{S}$ is tracking a poorly visible moving target.
3. Even if the user's resource availability is perfectly known, it is often hard for a system to decide *what type of adaptation is appropriate*. For example, an especially concise instruction may initially save a user some time, but if it is too dense do be understood completely, it may in the end cause the user to waste time or make errors.

The READY project, which began in 1996, represents an initial effort to meet these challenges. This paper will first briefly summarize the research of the first three-year phase.[2] It will then discuss the new directions of research that will be pursued during the second phase.

The Current Prototype

The example scenario that was used during the first phase of research is illustrated in Figure 1: Users are drivers whose cars need minor repairs; they request assistance from the system in natural language via mobile phone. Exploratory empirical studies in this scenario (and with a different scenario involving emergency calls to a fire station) provided some background information concerning the following points:

- *Causes* of resource bottlenecks: factors (such as the difficulty of the current task, or the need to perform another task simulteneously) that create a situation in which the user's resources may not be adequate for successful interaction.
- *Symptoms* of resource bottlenecks: aspects of the user's behavior (e.g., speech disfluencies) that indirectly indicate the extent to which $\mathcal{U}$'s resources are adequate to meet the current demands.
- *Strategies* for dealing with resource bottlenecks: general ways in which systems (or persons) can adapt their behavior when a resource bottleneck is perceived.

[2] Jameson, Schäfer, Weis, Berthold, and Weyrath (in press) give a recent English-language overview of the research of this phase. Schäfer, Weis, Weyrath, and Jameson (1997) presented an earlier overview in German. The report by Wahlster and Jameson (1998b) is available on request. Papers presented at previous ABIS workshops include those by Schäfer and Weyrath (1996) and by Berthold (1997).

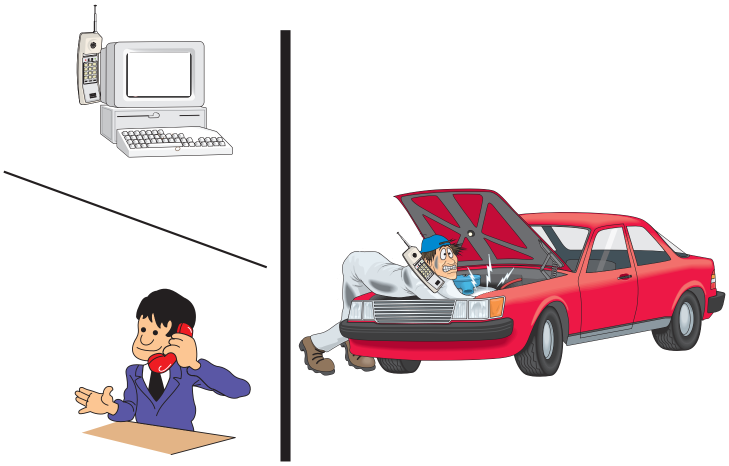

Figure 1. *Illustration of* READY*'s first example scenario, showing the similarity of* READY*'s role to that of an auto repairman who offers advice by phone.*

These studies guided the development of the initial READY prototype. The basic workings of the prototype can be explained with reference to the sketch of the architecture in the left-hand side of Figure 2.

The User Interface is designed so as to make it unnecessary for READY to deal with the challenging problems of speech processing raised by the scenario. Input is done via a natural language menu interface with which the "user" can compose utterances and specify a number of aspects of their form, such as the position and length of pauses.

What is sent to the Dialog Management component is a representation of $\mathcal{U}$'s utterance that contains the information that READY needs in order to update its user model and determine an appropriate response, i.e.: the meaning of $\mathcal{U}$'s utterance in the current dialog context plus any evidence that can be extracted from the input (including noises from the environment) that bears on $\mathcal{U}$'s current resource limitations.

The Dialog Management component uses its knowledge about the current dialog state and about possible diagnosis and repair actions to determine a set of possible dialog contributions (e.g., instructions) that might make sense in the current situation—not yet taking into account $\mathcal{U}$'s resource limitations. These contributions may differ in their basic content (e.g., prescribing a simple or a complex action) and/or in their form (e.g., using simple, redundant formulations or concise, technical ones). The Dialog Management component then sends the possible contributions to the User Modeling component, which decides which one seems best in the light of $\mathcal{U}$'s current resource limitations.

The User Modeling component employs dynamic Bayesian networks as a representation and processing formalism, because they are well suited to dealing with (a) the large amount of uncertainty involved in inferences about a user's resource limitations and (b) the dynamic nature of these resource limitations, which often change quite rapidly. Network Schemata represent assumptions about the causal relationships between resource limitations and their causes and symptoms.

Figure 3 gives an idea of the way in which this prototype adapts to a "user's" perceived resource limitations. The numbers on the right are the expected values of the system's current assessments of key variables, including available WM capacity (vAGK) and time pressure (Zeitdruck). In this case, the user (P) has just input the problem description Kühlwasser-Warnlampe leuchtet ("Cooling water warning lamp on"). Because of its unusual terseness and the absence

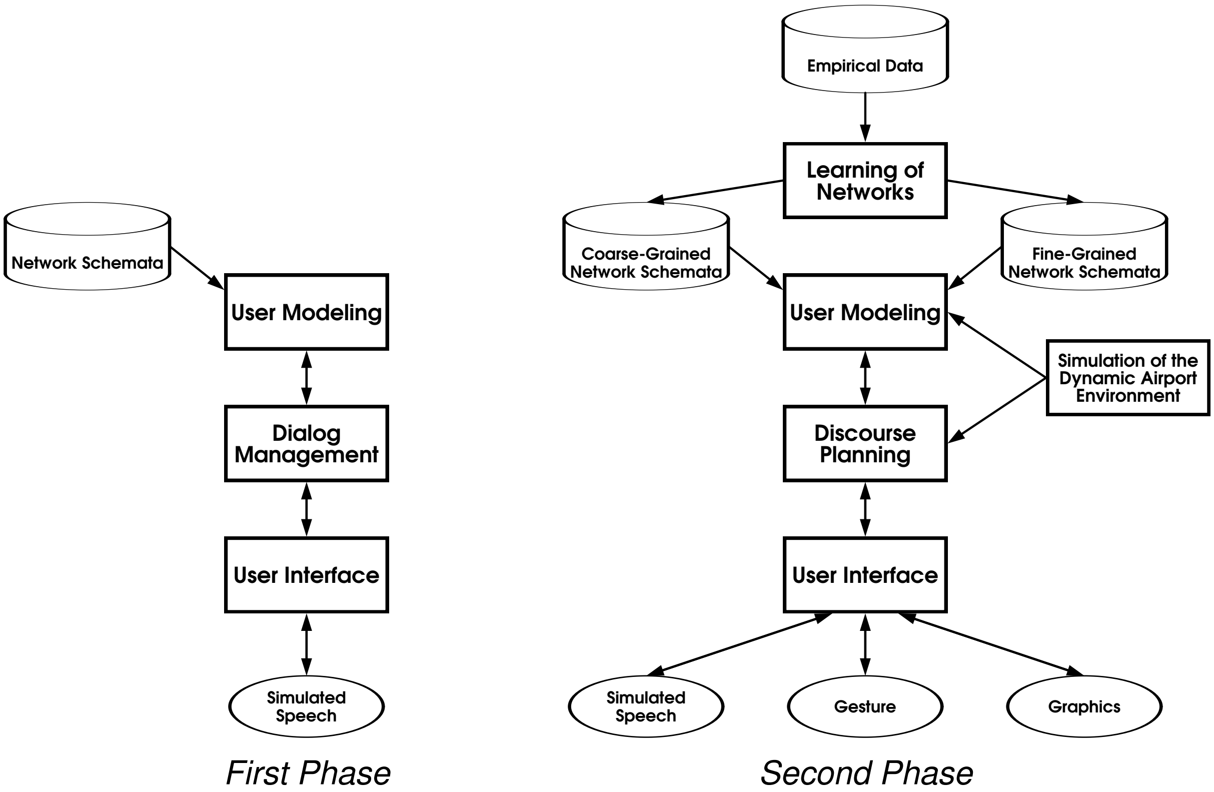

Figure 2. *Coarse-grained comparison of the architectures of the existing and the proposed* READY *prototypes.*

(Boxes denote processing components, cylinders knowledge bases, and ovals input and output. Arrows show the flow of information.)

of any symptoms of high WM load, READY's estimate of $\mathcal{U}$'s time pressure increases quite drastically, and the estimate of his available WM capacity increases somewhat as well. The reasons for these and other changes can be examined in the Bayesian network, shown here in the background, that is built up dynamically to model the user's changing resource limitations.

Overview of Plans for the Second Phase

In the second phase of research on READY, several new issues of resource-adapted interaction will be investigated.

1. How can the techniques developed in the first phase be generalized so that they can be applied to input and output media other than spoken language?
2. In what cases is it feasible and worthwhile to use a fine-grained model of the user's working memory and other resources?
3. How can the probabilities in the Bayesian networks be learned automatically on the basis of empirical data?
4. How can the techniques used so far for the selection of system utterances be generalized to the planning of larger dialog segments?

Especially the first two issues can be investigated more effectively within a new scenario, the *airport scenario*.

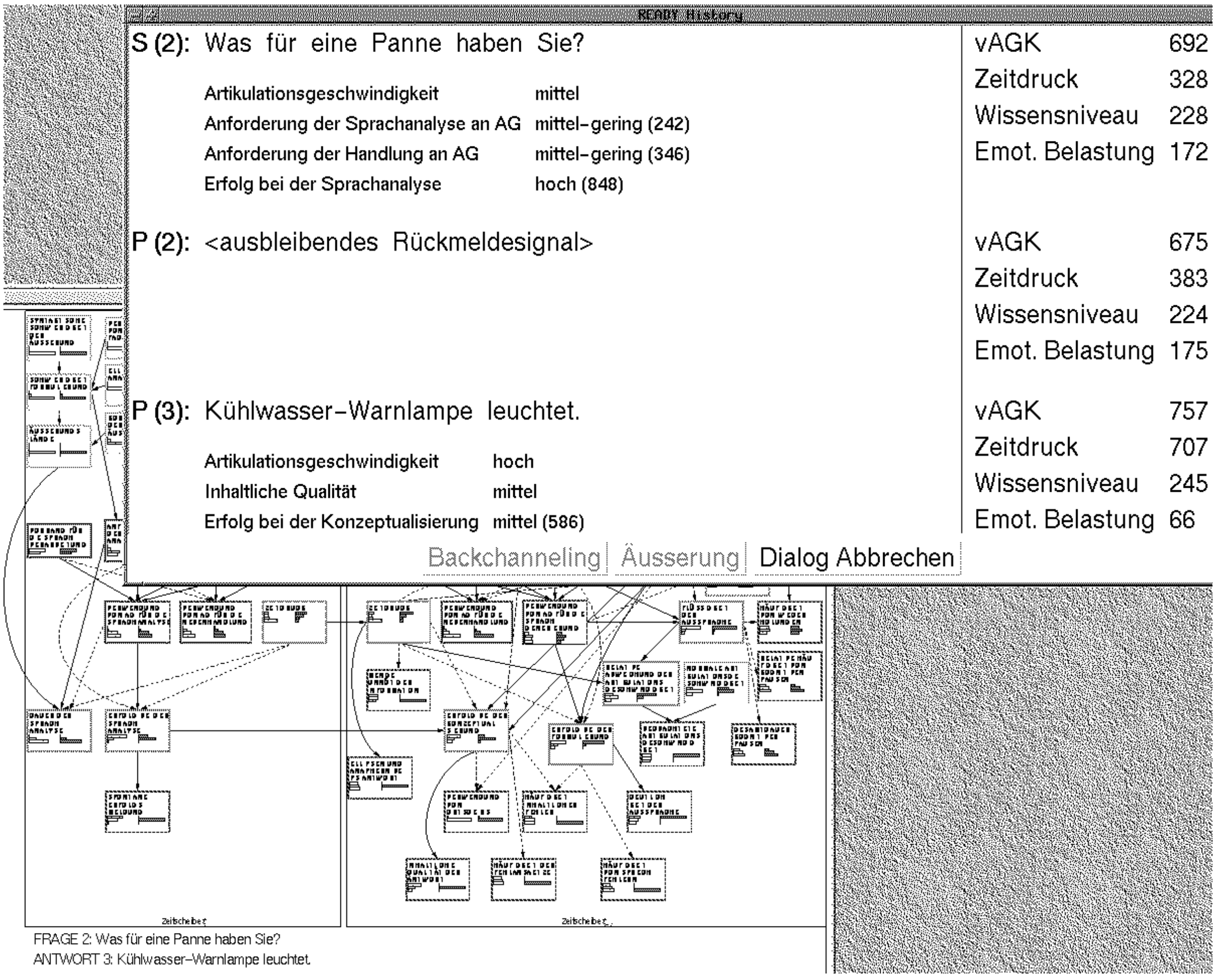

Figure 3. *The main interaction screen of the current* READY *prototype.*

The Airport Scenario

In the airport scenario, READY serves as a hand-held assistance system that offers a traveler various types of help, such as: describing the location of airport facilities and giving route descriptions; helping with the planning of travel to or from the airport; giving instructions about standard procedures or the use of airport facilities, and explaining access restrictions.[3]

In the airport scenario, the traveler's time restrictions can be even more severe than in the car-repair scenario. Likewise, a high WM load can arise—for example, when the traveler tries to remember a route description while dealing with obstacles in the airport. As was the case in the car-repair scenario, the system has to take into account not only the long-term limitations of the user's WM but also the user's current situation (competing tasks, distractions, and emotional pressure)—as well as static variables, such as the user's familiarity with the airport situation, that likewise influence WM load.

Figure 4 shows an example of the type of adaptation that READY is supposed to achieve in this scenario. Each of the two cases depicted involves a user who has just arrived on a flight. In both cases, READY is supposed to give the user information about the next train connection to

[3]Such a system would have to get much of its information and computing power via a wireless connection to a server. The actual development of the READY prototype will be done on a normal workstation that includes a simulation of the airport context.

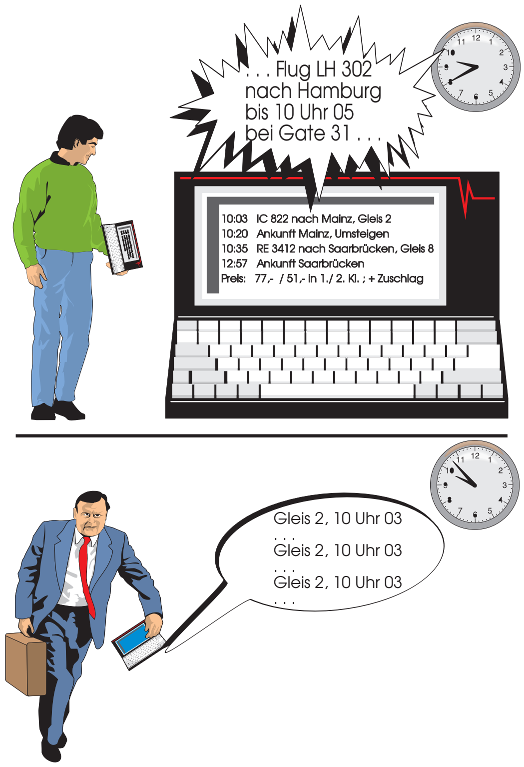

Figure 4. *Example of the type of adaptation aimed at in the airport scenario.*

Saarbrücken. The vacationer shown in the first case (above in the figure) has a good deal of time left to catch the train. Because of the announcements that are being made over the loudspeakers, he might have some difficulty in remembering information that is presented through speech output; on the other hand, since he doesn't have to get moving immediately, he can easily read off data from the system's display. Accordingly, the system displays in tabular form all of the data that might be of some use.

The businessman in the second hypothetical case has only 10 minutes to catch his train. Because he is also under emotional pressure for other reasons—as the system has noticed on the basis of other cues—his effective WM capacity is lower that usual. In this situation, the only data that are important enough to be worth presenting are the ones that the user absolutely requires in order to catch the train. Moreover, because of his need to make his way through the airport, the user cannot consult the display repeatedly. Accordingly, the system outputs the essential information via speech. It also repeats the information at regular intervals, in case the user should lose it from working memory because of some distraction.

Proposed Architecture

The right-hand side of Figure 2 shows how READY's architecture is to be extended during the second phase. The new module Simulation of the Dynamic Airport Environment will enable the system to take into account the fact that relevant aspects of the environment can change during an interaction (e.g., when a flight is delayed). The JAVA-based User Interface must be extended to be able to handle the additional forms of input and output that are appropriate in the airport scenario (see Issue 1 below). The essentially reactive Dialog Management that

READY currently incorporates will be replaced by a module for Discourse Planning that looks farther ahead in the dialog (Issue 4). The User Modeling module, instead of employing a single set of network schemata to construct its models of individual users, will choose from among schemata of different degrees of granularity (Issue 2). And the network schemata will have a stronger empirical basis that the ones that have been used so far, having been derived at least in part from empirical data through the use of machine learning techniques (Issue 3).

The rest of this paper will consider, in turn, each of the four main new directions of research.

Issue 1: Generalization of Methods to Other Interaction Techniques

In the airport scenario, various interaction techniques (besides speech) are potentially usable by READY: For user inputs, these include pointing (and simple drawing) with a stylus on a small touch screen, handwriting, and the use of a (physical or virtual) keyboard. As has been the case so far with speech input, for practical reasons there are no plans to include a real recognizer for handwriting. Instead, handwriting input will be emulated with a simpler interaction technique. For system outputs, the applicable new techniques include printed text and tables, maps and drawings.

When the selected interaction techniques are implemented in the User Interface, the emphasis will be less on achieving state-of-the-art quality than on exploring possibilities for adaptation. Where input techniques are involved, the goal is to ensure that the interface transmits to the Discourse Planning module useful clues concerning the user's resource limitations (e.g., errors, hesitations, or other irregularities). On the output side, the interface must have enough functionality to be able to present each of the types of message that the system generates in various ways, depending on the state of the user's resources.

The two central modules (Discourse Planning and User Modeling) will also have to be generalized so that they can deal with the new techniques. In particular, the following points have to be taken into account:

1. The observable *symptoms* of resource limitations are largely different when different input techniques are used (e.g., illegible handwriting instead of poor articulation; recognizably incorrect menu inputs instead of inappropriately formulated utterances).

2. Although some of the *causes* of resource bottlenecks are the same (e.g., emotional arousal, distractions from the environment), we also find some new, media-specific causes (e.g., poorly legible system output, inaccurate handwriting recognition).

3. The system will now have, for each message it has to present, a much larger set of possible specific ways of presenting the message: It has to decide on an output technique (or a combination thereof) before choosing the particular form of its message. Also, the specific considerations that have to be taken into account can differ from one input technique to the next. For example, with speech output READY tries to avoid giving long, detailed instructions if it suspects that the user is experiencing heavy WM load. But when an instruction of this sort is presented as text on a display screen, the considerations become different, since in this case $\mathcal{U}$ doesn't have to remember the entire instruction.

4. $\mathcal{S}$ can also influence the input techniques used by $\mathcal{U}$. This possiblity offers the system new ways of taking $\mathcal{U}$'s resource limitations into account. For example, if $\mathcal{U}$ is suffering from distraction, $\mathcal{S}$ might present menus rather than allowing $\mathcal{U}$ to make inputs freely with handwriting.

5. Finally, the use of the interaction techniques listed above may interfere with $\mathcal{U}$'s per-

formance of actions in the environment to a greater extent than the use of speech does. This complication is discussed in connection with Issue 2.

The investigation of these problems can draw from results of research on *intelligent multimedia interfaces* (see, e.g., Maybury & Wahlster, 1998, for an overview). A system in this class employs different interaction techniques, the choice depending on characteristics of the current situation. In this area, little explicit attention has been paid to working memory constraints (though the criteria for the selection of interaction techniques often seem to be based on the implicit goal of avoiding unnecessary WM load). Among the systems that take into account time constraints, some reason mainly about the temporal requirements of the system's own processing (see, e.g., Butz, 1997; Han & Zukerman, 1997). Others, like READY, base decisions on predictions of the user's processing time. For example, MOSES (Maaß, 1996) is a cognitive agent that estimates the time that an automobile driver will need to decode instructions and derive the appropriate navigation behavior. PEDRO (Van Mulken, 1998) includes a Bayesian model for predicting and interpreting both (a) the time that the user will need to decode a multimodal presentation and (b) the likelihood that the decoding will be successful.

Another useful source of ideas is research in the area of human-computer interaction on the properties of various interaction techniques. Many studies have yielded results concerning the causes and consequences of excessive WM load (see, e.g., Conklin, 1987) or of time limitations (see, e.g., Muller et al., 1995) on the part of users.

Issue 2: More Fine-Grained Modeling of the User's Resources

One question that the research in the first phase of READY leaves open concerns the degree of granularity with which the user's working memory should be modeled. On the one hand, basic psychological research on working memory has produced a great deal of evidence that WM comprises several interrelated subsystems, such as a *phonological loop* and a *visuo-spatial scratchpad* (see, e.g., Baddeley, 1986; Logie, 1995). On the other hand, when natural, complex tasks are investigated, it becomes more difficult to identify specific roles for such subsystems. And a dialog system that tries to model a user's working memory on-line will typically have access to less extensive specific data about the user than psychologists have about experimental subjects (cf. Jameson, 1997).

Viewed in this light, it is not remarkable that READY has so far gotten by with a way of modeling working memory—as a homogeneous store with a limited capacity—that is crude in comparison with the models developed in basic working memory research. On the other hand, it is by no means clear that the degree of granularity that READY currently uses is optimal. In fact, it is quite possible that models of different degrees of granularity are desirable in different situations. If this is the case, READY should ultimately be equipped with:

- a set of several schemata for modeling WM use that differ in their degrees of granularity; and
- criteria for choosing the most appropriate schema in any given situation.

The approach taken here is to look for situations in which a fine-grained modeling of working memory has the greatest chance of being worthwhile. A likely example would be the sort of situation depicted in the top half of Figure 4: READY's current schema for modeling WM would underestimate the extent to which the loudspeaker announcements would interfere with the traveler's attempts to remember his track number and departure time, since it would not take into account the specific properties of the phonological loop. If it turns out that a finer-grained modeling is both feasible and useful in situations like this, the multiple-schema approach sketched above (and reflected in the proposed architecture in Figure 2) will be realized.

In the airport scenario there are other resources of the user which, though in some ways simpler than working memory, raise some novel modeling problems. In order to perform tasks in the airport and navigate through it, the user often has to employ the same sensors (mainly eyes and ears) that are also required for interaction with the system. If READY can predict the ways in which $\mathcal{U}$'s sensors are used for other tasks, it can adapt the way in which it presents its output accordingly. Similarly, if READY can predict which of $\mathcal{U}$'s effectors (e.g., hands, mouth) are being used at the moment, it can offer methods of input that don't require their use.

Some previous adaptive systems have taken into account long-term individual differences in perceptual and motor abilities (see, e.g., Fink, Kobsa, & Nill, 1997). Modeling the situation-dependent availability of the user's sensors and effectors is a more novel challenge that becomes especially relevant in connection with mobile computing devices.

Issue 3: Learning of Bayesian Networks on the Basis of Empirical Data

The empirical studies of the first phase yielded mainly qualitative results relevant to the structure of the Bayesian networks that constitute the core of READY's user modeling component. The quantitative specification of the causal links in these networks was based on rough estimates. These estimates were adequate for producing system adaptation that appears to be plausible, explainable, and potentially useful. But it is clearly desirable, for both theoretical and practical reasons, to ensure that the system's user modeling is as realistic as possible.

Fortunately, a large number of methods for automatically learning Bayesian networks from data have been developed in recent years. The main challenges for READY are:

1. to identify (and where necessary adapt) the learning techniques that best fulfill the specific requirements of READY;
2. to develop and apply methods for collecting empirical data that will yield the input that is required for the learning techniques.

The data collection methods will necessarily differ in a number of respects from the usual methods of psychological experimentation. For one thing, the goal is not to determine whether particular independent variables reliably affect other variables—or even to estimate the strength of such influences in terms of effect size. Instead, the goal is to arrive at conditional probability tables, each of which describes how a particular variable depends on one or more other variables.

The goal of this line of work is not just to obtain realistic network schemata for the final version of READY. More important will be the empirical and computational methods for arriving at such schemata. After all, the specific network structures and conditional probabilities that emerge from these studies will inevitably depend to a certain extent on the scenario, the system's functionality, the user group, and other context factors. Whenever a system is developed for a significantly different context, it will be necessary to apply the data collection and learning methods once again.

Learning techniques for Bayesian networks address one or both of two basic questions:

1. What *structure* is the network supposed to have—i.e., what variables are to be represented, and what causal links exist among them?
2. How can the individual causal links be quantified?

Overviews of the relevant learning techniques are given by the following authors, among others: Buntine (1996), Heckerman (1995), Krause (in press), and Russell and Norvig (1995, chap.

19). Some of the techniques presuppose that all of the variables in the network are *observable*, so that information about their actual values is available when the network is learned. These techniques are of limited use in the context of reasoning about a user's situational resource constraints. To be sure, some variables that are unobservable in practical contexts can be treated as observable in a special setting in which data are collected.[4] But some variables correspond to theoretical constructs that have no simple relationship to observable variables—e.g., the demands placed by a particular understanding task on $\mathcal{U}$'s WM. Such variables have to be treated as *hidden variables*. Consequently, the techniques that allow for such variables are the ones with the greatest relevance (see, e.g., Bauer, Koller, & Singer, 1997; Binder, Koller, Russell, & Kanazawa, 1997).

When hidden variables are involved, it is essentially necessary to specify the structure of the to-be-learned network in advance. In spite of some efforts to develop techniques for learning the structure of networks with hidden variables (see, e.g., Cooper & Herskovits, 1992), such techniques have not yet reached a stage of development where they could be useful in the present context (cf. Russell & Norvig, 1995, chap. 19).

Issue 4: Planning of Larger Discourse Structures

In the first phase of the research, the adaptation of READY involved the evaluation, selection, and formulation of individual utterances. Specifically, when the existing prototype considers making a particular utterance, it predicts the utterance's effects in two *time slices*:

1. the time slice in which the utterance is comprehended by $\mathcal{U}$; and
2. the time slice in which $\mathcal{U}$ responds to the utterance (e.g., by answering a question or carrying out an instruction).

But often $\mathcal{S}$ has to plan farther ahead. For example, $\mathcal{S}$ may need to present instructions for a sequence of several actions. Which of the following forms of presentation is more appropriate?

1. A sequence of individual instructions, after each of which the system will wait until $\mathcal{U}$ reports completion of the action; or
2. a single, longer instruction in which all of the actions are specified, perhaps more compactly than could be accomplished with a sequence of instructions.

Presumably the choice should depend on (a) the specific demands that the instructions and actions place on $\mathcal{U}$'s resources (e.g., working memory); and (b) the current availability of these resources.[5]

To be able to reason about cases like this, READY requires an appropriate representation of discourse structure. Rhetorical Structure Theory has been used successfully in dialog systems, both those using text (see, e.g., Moore & Paris, 1993) and those using multimodal input and output (see, e.g., Stein & Thiel, 1993). The representation of discourse structure will have to be extended so that it can represent information about the demands that a segment of discourse is likely to place on $\mathcal{U}$'s resources. As a starting point, Weis (1997) has developed a representation of action plans that includes such information.

[4] For example, in a normal dialog READY will in general not be able to observe directly whether $\mathcal{U}$ is engaged in a concurrent task, or how much time pressure $\mathcal{U}$ is experiencing. In an experimental situation, these variables can be manipulated more or less directly, and therefore their values are known.

[5] At the time of this writing, this hypothesis is being tested in an experiment.

Concluding Remarks

The problems that are to be dealt with during the next phase of the READY project are challenging and partly open-ended. It will be some years before thoroughly worked out solutions to them can be incorporated in practically useful, deployed systems. On the other hand, given the generality and long-term importance of the issues involved, basic research on them does seem to be warranted at this point in the development of research on adaptive systems.

References

Baddeley, A. D. (1986). *Working memory.* Oxford: Oxford University Press.

Bauer, E., Koller, D., & Singer, Y. (1997). Update rules for parameter estimation in Bayesian networks. In D. Geiger & P. P. Shenoy (Eds.), *Proceedings of the Thirteenth Conference on Uncertainty in Artificial Intelligence* (pp. 3-13). San Francisco: Morgan Kaufmann.

Berthold, A. (1997). „Ich – ähm – denke, daß ...": Zu sprachlichen Indikatoren für Ressourcenbeschränkungen und ihrer Verarbeitung in natürlichsprachlichen Dialogsystemen. In R. Schäfer & M. Bauer (Eds.), *ABIS-97, Adaptivität und Benutzermodellierung in interaktiven Softwaresystemen [ABIS-97, Adaptivity and User Modeling in Interactive Software Systems]* (pp. 139-147). Saarbrücken, Germany: SFB 378, University of Saarbrücken.

Binder, J., Koller, D., Russell, S., & Kanazawa, K. (1997). Adaptive probabilistic networks with hidden variables. *Machine Learning*, *29*, 213-244.

Buntine, W. (1996). A guide to the literature on learning probabilistic networks from data. *IEEE Transactions on Knowledge and Data Engineering*, *8*, 195–210.

Butz, A. (1997). *Ein inkrementeller Ansatz zur Generierung informativer 3D-Animationen [An incremental approach to the generation of informative 3D animations].* Unpublished doctoral dissertation, University of Saarbrücken, Germany.

Conklin, J. (1987). Hypertext: An introduction and survey. *IEEE Computer*, *20*, 17-41.

Cooper, G. F., & Herskovits, E. (1992). A Bayesian method for the induction of probabilistic networks from data. *Machine Learning*, *9*, 309-347.

Fink, J., Kobsa, A., & Nill, A. (1997). Adaptable and adaptive information access for all users, including the disabled and the elderly. In A. Jameson, C. Paris, & C. Tasso (Eds.), *User modeling: Proceedings of the Sixth International Conference, UM97* (pp. 171-173). Vienna, New York: Springer Wien New York.

Han, Y., & Zukerman, I. (1997). A mechanism for multimodal presentation planning based on agent cooperation and negotiation. *Human-Computer Interaction*, *12*, 187-226.

Heckerman, D. (1995). *A tutorial on learning with Bayesian networks* (Tech. Rep. No. MSR-TR-95-06). Microsoft Research. (Revised November 1996)

Jameson, A. (1997). Modeling the user's processing resources: Pragmatic simplicity meets psychological complexity. In R. Schäfer & M. Bauer (Eds.), *ABIS-97, Adaptivität und Benutzermodellierung in interaktiven Softwaresystemen [ABIS-97, Adaptivity and User Modeling in Interactive Software Systems]* (pp. 149-160). Saarbrücken, Germany: SFB 378, University of Saarbrücken.

Jameson, A., Schäfer, R., Weis, T., Berthold, A., & Weyrath, T. (in press). Making systems sensitive to the user's time and working memory constraints. In M. T. Maybury (Ed.), *IUI '99: International Conference on Intelligent User Interfaces.* New York: ACM.

Krause, P. J. (in press). Learning probabilistic networks. *Knowledge Engineering Review*.

Logie, R. H. (1995). *Visuo-spatial working memory.* Hove, UK: Erlbaum.

Maaß, W. (1996). *Von visuellen Daten zu inkrementellen Wegbeschreibungen in dreidimensionalen Umgebungen: Das Modell eines kognitiven Agenten [From visual data to incremental route descrip-*

tions in three-dimensional environments: The model of a cognitive agent]. Unpublished doctoral dissertation, University of Saarbrücken, Germany.

Maybury, M. T., & Wahlster, W. (1998). Intelligent user interfaces: An introduction. In M. T. Maybury & W. Wahlster (Eds.), *Readings in intelligent interfaces* (pp. 1–13). San Francisco, CA: Morgan Kaufmann.

Moore, J. D., & Paris, C. L. (1993). Planning text for advisory dialogues: Capturing intentional and rhetorical information. *Computational Linguistics*, *19*, 651–694.

Muller, M. J., Carr, R., Ashworth, C., Diekmann, B., Wharton, C., Eickstaedt, C., & Clonts, J. (1995). Telephone operators as knowledge workers: Consultants who meet customer needs. In I. R. Katz, R. Mack, L. Marks, M. B. Rosson, & J. Nielsen (Eds.), *Human factors in computing systems: CHI '95 conference proceedings* (pp. 130-137). New York: ACM.

Russell, S. J., & Norvig, P. (1995). *Artificial intelligence: A modern approach.* Englewood Cliffs, NJ: Prentice-Hall.

Schäfer, R., Weis, T., Weyrath, T., & Jameson, A. (1997). Wie können Ressourcenbeschränkungen eines Dialogpartners erkannt und berücksichtigt werden? [How can resource limitations of a dialog partner be recognized and taken into account?]. *Kognitionswissenschaft*, *6*, 151-164.

Schäfer, R., & Weyrath, T. (1996). Einschätzung von verfügbarer Arbeitsgedächtniskapazität mit temporalen Bayesschen Netzen. In H.-G. Lindner (Ed.), *ABIS-96, Adaptivität und Benutzermodellierung in interaktiven Systemen: Workshop-Beiträge [ABIS-96, Adaptivity and User Modeling in Interactive Software Systems: Workshop Papers]* (pp. 5.1-5.9). Dortmund: VEW AG.

Stein, A., & Thiel, U. (1993). A conversational model of multimedia interaction in information systems. In *Proceedings of the Eleventh National Conference on Artificial Intelligence* (pp. 283–288).

van Mulken, S. (1998). *Modeling the user's text and graphics understanding in intelligent multimedia interfaces: A probabilistic approach.* Unpublished doctoral dissertation, University of Saarbrücken, Germany.

Wahlster, W., & Jameson, A. (1998a). READY: Benutzerorientierte Ressourcenadaptation [READY: User-oriented resource adaptation]. In W. H. Tack (Ed.), *Ressourcenadaptive kognitive Prozesse: Fortsetzungsantrag, Januar 1999 – Dezember 2001 [Resource-adaptive cognitive processes: Proposal for the second funding period, January 1999 – December 2001].* Saarbrücken, Germany: SFB 378, University of Saarbrücken.

Wahlster, W., & Jameson, A. (1998b). READY: Ressourcenadaptive Dialogführung [READY: Resource-adaptive dialog]. In W. H. Tack (Ed.), *Ressourcenadaptive kognitive Prozesse: Arbeits- und Ergebnisbericht, Januar 1996 – Januar 1998 [Resource-adaptive cognitive processes: Report on activities and results, January 1996 – January 1998].* Saarbrücken, Germany: SFB 378, University of Saarbrücken.

Weis, T. (1997). Resource-adaptive action planning in a dialogue system for repair support. In G. Brewka, C. Habel, & B. Nebel (Eds.), *KI-97: Advances in artificial intelligence* (pp. 361-372). Berlin: Springer.