In: Journal Proceedings of the IEEE, Special Issue on Natural Language Processing, July 1986, pp. 948 -960.

# Dialog-Based User Models

W. Wahlster, A. Kobsa

Sonderforschungsbereich 314
University of Saarbrücken
Dept. of Computer Science
AI Laboratory
D-6600 Saarbrücken 11
WEST GERMANY

## Abstract

The paper investigates several approaches to user modeling in natural-language dialog systems. First, reasons are pointed out why user modeling has become so important in the last few years, and definitions are proposed for the notions of 'user model' and 'user modeling component'. Then, techniques for constructing user models in the course of a dialog are presented and recent proposals for representing a wide range of assumptions about a user's beliefs and goals in a system's knowledge base are surveyed. Examples for the application of user models in systems developed to date are presented, and some social implications are discussed. Finally, unsolved problems like coping with collective beliefs or resource-limited processes are investigated, and prospects for application-oriented research are outlined.

## 1. Introduction

During the 70's, many special-purpose natural language (NL) *interfaces* have been developed for various domains of discourse, e.g. moon rock samples, airline fares, computer installations, payroll data, aircraft maintenance data, or university courses. These systems had no interests beyond providing the information-seeking user with relevant data by just *responding* in a slavishly cooperative way to the user's questions. The conversational setting for such dialog systems was somewhat unnatural compared to human dialogs. The user of these systems has to find an appropriate question strategy for getting the information s/he thinks might be relevant for solving his/her problem, and can hardly expect any assistance of the system in this respect

This research was partially supported by the German Ministry for Research and Technology (BMFT), the German Science Foundation (DFG), the Austrian Ministry of Science, and DEC's EERP. We would like to thank G. Ferrari, Ch. Habel, M. Hecking, A. Jameson and an anonymous referee for their comments on an earlier version of this paper.

During the last five years a new class of task-oriented dialog systems has emerged, all of which are mainly characterized by the *conversational roles* which the system and the user are expected to take on. Here, more realistic conversational settings are modeled in which the system may play e.g. the part of

- a clerk in an information booth at a train station (cf. Allen 1979)
- a hotel manager who tries to rent all his rooms (HAM-ANS, cf. Jameson & Wahlster 1982)
- a librarian who recommends novels that the user might like to read (GRUNDY, cf. Rich 1979a, b)
- a nosey guest at a cocktail party who wants to find out as much as possible about the user (ULLY, cf. Hayes & Rosner 1976)
- an expert who assists an apprentice in repairing an air compressor (KAMP, cf. Appelt 1982, 85)
- a UNIX and a SINIX consultant for beginners (UC, cf. Wilensky et al. 1984; SG, cf. Kemke 1985)
- a tax advisor who assists the user in filling in his/her income tax form (XTRA, Kobsa etal. 1986)

These systems are designed as *active* dialog partners who engage in a mixed-initiative dialog (Wahlster 1984). In contrast to the NL interfaces mentioned above, the user of these systems needs no prepared question strategy since the system tries to recognize the user's *intention* in respect to the domain of discourse in order to exhibit more cooperative dialog behavior (e.g. in order to provide better advice). Although even in NL interfaces such a characteristic was necessary to a limited extent, it is particularly in the above conversational settings that the construction and use of an *explicit model of the user's beliefs, goals and plans* becomes a central problem.

Thus, one reason for the recent emphasis on user modeling is the fact that such models are necessary prerequisites for a system to be capable of exhibiting a wide range of cooperative dialog behavior (cf. also Carbonell 1983, Rich 1983, Sparck Jones 1984, Boguraev 1985). A cooperative system must certainly take into account the user's goals and plans, his/her prior knowledge about a domain, as well as false conceptions a user may have concerning the domain (empirical evidence for this in the case of human expert consultation can be found in Pollack et al. 1982). Thus it is no longer only the user's task to construct a mental model of the technical functioning of the system. Instead, it should also be up to the system to draw assumptions on what the user believes, wants and plans, i.e. to develop a model of the user. Ideally, neither the user's nor the system's modeling task should differ to that which is required in person-to-person communication.

A simple example for such cooperative dialog behavior might look as follows:

(1) User: Where can I find the nearest gas station?
System: The nearest one which is still open is located

In order to respond in this cooperative way, a system must discover the presumable plans underlying the user's question, represent them in its knowledge base, examine them for hidden obstacles, and provide information to enable the user to overcome these obstacles. A user model is an indispensable prerequisite for these complex inference processes.

A second reason for the recent emphasis on user modeling is that it has become evident in the last few years that a model of the user is also an important base for intelligent dialog behavior in general, independent of whether the dialog is cooperative or not. Such models are required for identifying the objects which the dialog partner is talking about, for analyzing a non-literal meaning and/or indirect speech acts in his/her dialog contributions, and for determining what effects a planned dialog contribution will have on the dialog partner, etc. (cf. Allen & Litman 1986) . Thus a user model does not just sweeten-up a dialog system to render it more cooperative. Instead, user models form an indispensable prerequesite for any flexible dialog in a wider domain. They interact heavily with all other components of the system and often cannot be separated from them.

In this paper we will restrict ourselves to user models in natural-language dialog systems. In this context, we will use the following basic definitions:

> A *user model* is a knowledge source in a natural-language dialog system which contains explicit assumptions on all aspects of the user that may be relevant for the dialog behavior of the system.
>
> A *user modeling component* is that part of a dialog system whose function is to incrementally build up a user model; to store, update and delete entries in it; to maintain the consistency of the model; and to supply other components of the system with assumptions about the user.

Within the area of artificial intelligence, research on user modeling in NL dialog systems is related to several other fields, namely to the work on *intelligent computer-aided instruction (ICAI), multiple-agent planning systems,* and in part *text comprehension and generation.* ICAI systems (Sleeman & Brown 1982, Reiser et al. 1985) typically incorporate a "student model" which represents the student's understanding of the material to be taught. A simple modeling technique in this field is the overlay approach (e.g., Carr & Goldstein 1977, Clancey 1979), in which the student's knowledge is represented as a subset of the system's knowledge about the domain (cf. section 3). Compared to NL dialog systems, however, the possible user's input to ICAI systems is usually very restricted.

Sometimes only expressions of a formal language are accepted, such as mathematical formulas or statements of a programming language.

Research in the field of multiple-agent planning (e.g. Power 1979, Konolige & Nilsson 1980, Rosenschein 1982, Georgeff 1983) investigates the problems that a group of agents have in collectively constructing and/or executing a plan for a given task. To coordinate their actions, each agent has to represent the beliefs and goals of the others. Much of this work is grounded on the belief representations developed in the possible-world approach (see Section 3). Problems of natural language communication between agents are not studied in this field of research. Moreover, for representational purposes, a number of idealizations is typically made with respect to the kinds of beliefs and goals which agents are allowed to hold. Agents may sometimes not possess beliefs which are contradictory to other agent's beliefs, or may not hold various forms of mutual and infinite-reflexive beliefs (see Section 2.3). The former constellation, however, is quite common in man-computer interaction (a lot of work in the field of user modeling is devoted to recognizing user misconceptions, e.g. Schuster & Finin 1983; McCoy 1985a, b), and the latter is practically ubiquitous.

## 2. Building up user models

In this section, a number of knowledge sources will be identified from which information about the user can be obtained, and a number of techniques for building up a user model in the course of a dialog will be discussed.

### 2.1. Default assumptions

In systems with a limited area of application, various beliefs and goals can be attributed to any user of the system, as long as there is no evidence to the contrary. Such default assumptions made by the system may concern, e.g.,

- *The user's general knowledge:* A management information system (MIS) with a user modeling component, for instance, can assume that all of its users possess basic knowledge in business administration.

- *The user's beliefs about individual facts:* An MIS installed in a firm can also assume that the user is familiar with the principal facts concerning the firm.

- *The user's goals:* All users of a hotel reservation system (Hoeppner et al. 1980) can be assumed to be searching for rooms which meet their requirements.

Other possible default assumptions concern the user's beliefs about the system's goals (e.g., that the system wants to rent its rooms), or assumptions about the user's beliefs about the system's beliefs, etc.

## 2.2. Initial individual user models

One criterion which helps to differentiate between the various dialog systems that include a user modeling component is whether

(A) The system can, at the beginning of a dialog, access an already existing model of a particular user which it had built up during previous interactions with him/her.
(B) The system has no access to information about individual users at the beginning of the dialog.

In the former case, the recorded user model will probably provide a valuable source of information for the current dialog as well. An example for a system with such an initial user model is GRUNDY (Rich 1979a, b) which, at the end of a dialog session, writes all information about the user inferred from his/her dialog behavior into a corresponding file. Whenever that user returns to the system and types in his/her name, his file will be retrieved and used as the initial user model.

Today, however, the latter type of systems is developed and implemented much more often than the former one. This is due not only to the legal, social and ethical problems of security and privacy in such systems, but also to the fact that for many prospective applications in the era of distributed systems the probability that a user will consult the same system more than once is quite low.

## 2.3. Assumptions from the user's input

Assumptions gained from the user's input into the system are the most direct and, hence, also the most reliable in problem solving domains.

There are various techniques for transforming a user's input into assumptions about him, which differ in complexity and in the reliability of the inferred assumptions. The most simple type occurs if the user directly expresses a belief or goal, as in

(2) User: I want to buy your new 500 XGL.
Assumption of the system: The user wants to buy one 500 XGL.

Unfortunately, such direct statements of the user about his/her beliefs or goals are usually very rare. Much more frequent are cases in which assumptions about the user have to be indirectly inferred from the user's input. One type of such inferences is purely *structure-based.* In this case, only the form of a user's input is used for drawing inferences about him/her, and not its meaning. The VIE-DPM system, for instance, draws the following assumptions from a user's wh-question, and enters them into its knowledge base (Kobsa 1985c, forthcoming).

(3) User: To whom does Peter give the book?
Assumptions of the system:

(a) The user believes that Peter and the book exist and that Peter gives the book to somebody whose identity is unknown to the user.

(b) The user believes that the system knows whom Peter gives the book to.

(c) The user wants to be in a situation where

(*) both the user and the system have the same opinion about whom Peter gives the book to, and believe that (*)

(d) The user believes that the system believes that (a)-(d).

The infinite loops in the above definition describe specific belief constellations which are fundamental in dialog settings, namely *mutual beliefs* (or more precisely, *infinite-reflexive beliefs* - cf. Kobsa 1984). All of the above assumptions can be drawn simply because the user has entered a wh-question, without regard to the content of his/her question.

Other purely structural rules of inference (which were used, e.g., in the works of Allen & Perrault 1980, Carberry 1983, Sidner 1983 and McKeown et al. 1985) include

(4) If the user wants to know how to achieve an effect, then his/her goal may be to achieve that effect.
(5) If the user wants to know whether p, then his/her goal may be that p, or not p

Other forms of inference depend on the *meaning* of a user's input to the system, and on a large amount of world knowledge about the domain of discourse. In (3), for instance, the system could also infer

(3e) Assumption of the system: The user believes that the book is now in the possession of this person whose identity is unknown to him/her.

This inference is based upon the content of the user's question, and upon world knowledge about the meaning of 'to give' (namely that it denotes an action, the result of which being the possession of an object by the recipient). The systems of Allen & Perrault (1980), Carberry (1983) and McKeown et al. (1985) have a large number of domain-dependent precondition-effect sequences (i.e., operators) stored in their knowledge bases. They can then use inference rules like

(6) If the user's goal is to achieve certain effects, then s/he will probably use one of the operators which yield this effect and whose preconditions are fulfilled.

for inferring probable plans of the user.

Another type of domain-dependent inferences are those by which assumptions are drawn from "objective" facts about the user to probable global user characteristics, in particular to his/her preferences, requirements, or criteria of assessment in a certain domain. Rich's (1979a, b) GRUNDY system, for instance, draws assumptions about the user's selection criteria for books to read from information about personality characteristics offered by him/her. Morik & Rollinger (1985) draw assumptions about the user's assessment criteria for appartments from information about his/her personal situation, such as the number of children, the acceptable price range, etc. If a great number of inferences is drawn from an antecedent (as is in the purest form the case in GRUNDY), then the set of their consequents may be regarded as a *stereotype* (see section 4.2).

Additional assumptions about the user can be drawn from *linguistic particles* in his/her input, as in

(7) User (to a text editor system): Why is this line not centered?
Assumption of the system: The user wants the line to be centered.

(8) User: I do nor have enough disk space.
Assumption of the system: The user wants more disk space.

Up to now, the analysis of linguistic particles in the user's input has mostly been investigated on a theoretical basis (Kobsa 1985a). Such an analysis would offer a simple way of obtaining assumptions about the user. It seems, however, that interesting particles are not very frequently used in goal-oriented dialogs. In contrast to the techniques discussed above this method is also only applicable in natural-language dialog systems.

## 2.4. Assumptions from dialog contributions made by the system

Dialog contributions of the system (i.e. answers, questions and commands) also lead to new entries in the user model. If the system had previously planned the dialog contribution by trying to anticipate its probable effect on the user, it can now hypothetically enter these effects into the user model (e.g., after answering a user's question the system can assume that the user now knows the content of the answer). These entries may give rise to the planning of additional unsolicited comments (Jameson 1983, cf. section 4.3.3) or follow-up information and explanation.

## 2.5. Assumptions from non-linguistic input of the user

In section 2.3, a number of techniques were discussed for drawing assumptions about the user from his/her input into the system. In contrast to a face-to-face conversation between humans, present dialog systems are restricted to deriving such assumptions solely from the strings typed in by the user; they cannot access any visual, acoustic or other external clues.

By expanding Norman's futuristic idea of a personal user interface on a pocket-sized "smart" card, which, when plugged into any terminal, makes it emulate the one at the user's home or office (cf. Joyce & Wingerson 1983), one could imagine a slight degression from that restriction, namely, that a system's user model is initialized by a kind of electronic identity card storing some of its owner's personality characteristics. The information on this card may be regarded as an initial individual user model in the sense of section 2.2. The information thereon would be more general, however, and not restricted to the specific domain of discourse of a particular system.

At the beginning of an interaction with a dialog system, the use of such a device containing a user-provided individual profile seems no more unnatural than responding to a system's request like

(9) I'd like to know what sort of person you think you are, please type in a few single words that you think characterize the most important aspects of yourself

as is required in Rich's (1979a, b) GRUNDY system. Of course, the problem remains as to whether the conveniences offered by such personality cards are not far outweighed by the dangers they conjure up in respect to loss of privacy and easier surveillance.

## 3. Representing user models

In the field of user modeling, a wide range of formal means are employed for representing the assumptions of the dialog system regarding the user. What representation one opts for is usually determined by the application purpose of the system and by the demands imposed on the expressive power of the user model.

Rich's GRUNDY system, for instance, makes extensive use of simple *linear parameters.* For every personality trait in her system, one parameter indicates the assumed degree to which the personality trait characterizes the user, and another indicates the degree to which the system believes that its assumption about this specific personality trait is correct. An example for such personality traits is given in Fig.l.

| Personality trait | Value | Certainty of assumption |
|---|---|---|
| Education | 5 | 900 |
| Seriousness | 5 | 800 |
| Piety | -3 | 423 |
| Toierate-sex | 5 | 700 |
| Tolerate-violence | -5 | 597 |
| Tolerate-suffering | -5 | 597 |

*Fig.l:* Representation in the GRUNDY system

Sleeman's UMFE system (Sleeman 1985) also uses very simple methods for representing its assumptions about the expertise of a user in a given domain. UMFE is a front end system for the explanation component of expert systems. It takes the inference chains yielded by such components and adapts them to the specific state of knowledge of the current user. This is performed by making sure that only such concepts are used in an explanation which the user has explicitly indicated to be familiar with, or which the system has inferred him to know.

For representing the concepts that a user is probably familiar with, an *overlay technique* is employed in Sleeman's model. A simple parameter associated with the representation of each individual concept in the system's knowledge base indicates whether, according to the system's assumptions, the concept is KNOWN or NOT-KNOWN by the user, or whether the system has NO-INFORMATION in that respect. Another factor associated with each concept indicates the strength of belief with which the system holds its assumptions about the user's familiarity with these concepts. Currently, however, no use is yet made of these factors. An example for a simple overlay model (which is inspired by the UMFE system) is given in Fig.2.

| Concept hierarchy of the system | User model | |
|---|---|---|
| | user's knowledge state | certainty of assumption |
| INFECTQUS-PRQCESS | KNOWN | 100 |
| HEAD-INFECTION | KNOWN | 100 |
| SUBDURAL-INFECTION | NOT-KNOWN | 100 |
| OTITIS-MEDIA | NO-INFORMATION | 100 |
| SINUSITIS | NO-INFORMATION | 100 |
| MENINGITIS | KNOWN | 100 |
| BACTERIAL-MENINGITIS | KNOWN | 90 |
| MYCOBACTERIUM-TB | NOT-KNOWN | 70 |
| VIRUS | NOT-KNOWN | 90 |
| FUNGAL-MENINGITIS | NOT-KNOWN | 100 |
| MYCOTIC-INFECTION | NOT-KNOWN | 100 |
| ENCEPHALITIS | NOT-KNOWN | 90 |
| SKIN-INFECTION | KNOWN | 100 |

*Fig. 2:* An example for an overlay model

As soon as the user model is to give a slightly more detailed account of what the user knows or does not know, of his/her wrong beliefs, or of his/her goals and plans, etc., the linear scale values must be replaced by representational structures with greater expressive power. In natural-language dialog systems, for representing knowledge about the *world,*

representational schemes developed in the field of knowledge representation are employed. Such schemes are, for instance, formulas of first-order predicate calculus, network-like representational schemes, frames, etc. It thus seems most natural to extend these schemes so that the system's assumptions about the beliefs, goals and plans of the user become representable.

Unfortunately, a simple extension is not possible. When using predicate calculus, one has to decide by what epistemological primitives notions like 'believe', 'want', etc. should be represented. Opting to regard them as sentential operators (as are, e.g.,$\vee$, $\wedge$, $\sim$ or $\supset$) raises severe problems, since these operators would not be extensional (an operator is extensional if the truth value of a complex formula constructed by it solely depends on the truth value, and not on the "meaning", of its constituent formulas). The lack of this property raises serious problems since fundamental principles of predicate logic (as, e.g., substitution of terms with same extension, existential generalization) are no longer applicable.

Another possibility is to regard these notions as meta-language predicates which are applied to meta-language constants and variables that denote object-language formulas. These formulas, in return, express the content of the beliefs and goals. The system's assumption that the user believes that Peter loves Mary, thus, would be represented in this approach by something like 'BEL(USER,P)', where 'BEL' is a meta-language predicate, 'USER' a meta-language constant that denotes the user, and 'P' a meta-language constant that denotes an object-language formula like 'loves(peter,mary)'. Konolige (1981) developed a logic for belief representation that is based on this idea. For more complex belief constellations, however, such representations become computationally intractable. Moreover, as has been shown in Kobsa (1985a), a number of beliefs that are fundamental for dialog planning based on a user model cannot be represented in Konolige's scheme.

Another approach is to regard notions like 'believe' and 'want' as modal operators, and to use one of the many modal calculi for their axiomatization. This idea was first developed in Hintikka (1962). However, there is no proof procedure yet for modal logics. And, as was pointed out by Moore (1980), it will probably not be possible to develop such procedures by simply using standard axiomatizations of modal logic.

To overcome these difficulties, Moore (1980) and Appelt (1982) investigated a belief and goal representation which formalizes the standard semantics bf modal logic, namely *possible-world semantics.* In their representation, modal formulas are immediately translated into possible-world sentences, to which the standard deduction techniques can be applied. The modal operators 'believe' and 'want' are thereby translated into accessability relations between possible worlds. An agent (e.g., the user) knows some fact p if p is true in all worlds which are compatible with what s/he knows.

The system's assumption that Peter loves Mary, thus, would be represented in this approach by a formula like

(10) $\forall W$ [K(USER,W0,$W$) $\supset$ T($W$,P)]

*'W'* thereby is a meta-language variable ranging over possible worlds, 'W0' a meta-language constant denoting the world in which the user is currently in, and 'P' a meta-language constant denoting some object-language formula like 'loves (peter, mary)'.

'K (USER, WO, *W)*' is true iff *W* is accessible from WO according to what the user knows. 'T( *W,* P)' is true iff the formula denoted by P is true in *W*. Such a situation is depicted in Fig.3. Here the user knows the fact expressed by 'loves(peter,mary)' but is uncertain in respect to the fact expressed by 'loves(mary,peter)', since there is a world which is accessible to him according to what he knows in which the negation of the latter formula is true.

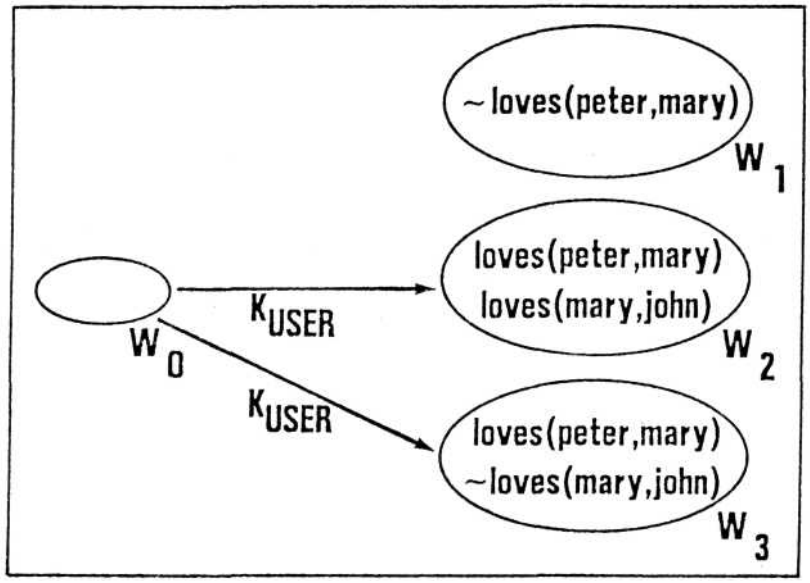


*Fig.3:* A possible-world representation of
'The user knows that Peter loves Mary and is uncertain whether Mary loves John'

Possible-world semantics has recently been intensively investigated as a means for belief representation (Halpern & Moses 1985, Fagin & Halpern 1985, Vardi 1985). However, as was pointed out in Kobsa (1985a), as far as the representational needs in the field of user modeling are concerned, the expressive power of this approach is somewhat limited.

Another approach to belief and goal representation is the *"partition approach"*, which is applicable to all forms of concept-based knowledge representations. It was first used by Cohen (1978) for his special network-oriented representation. The idea behind this approach is to maintain a number of separate partitions to store the system's beliefs about the domain, the system's goals, the system's assumptions about the user's beliefs about

the domain, the system's assumptions about the user's goals, the system's assumptions about the user's assumptions about the system's beliefs about the domain, etc. For representing the contents of these beliefs and goals, the standard concept-based knowledge representation schemes (e.g., network-like representations, predicate calculus) may then be employed. An example for such a representation with a partition for the system's beliefs and another for the system's assumptions about the user's goals is given in Fig.4.

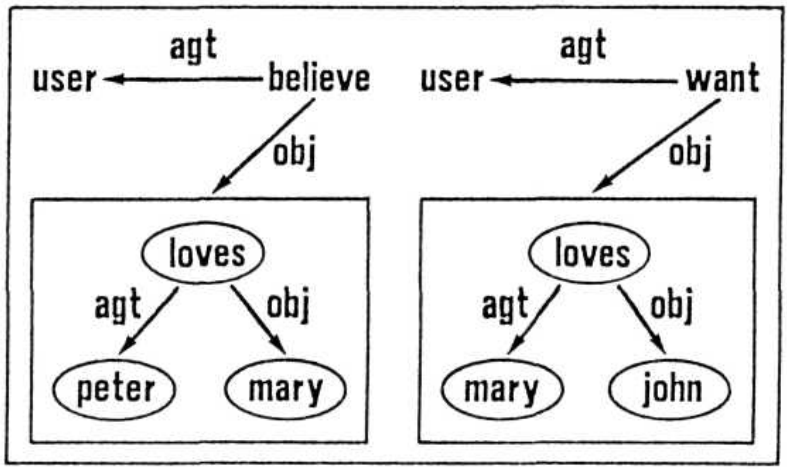


*Fig. 4:* A partition representation of 'The user believes that Peter loves Mary and wants that Mary loves John'

Unfortunately, in the form described here, this approach raises some problems which were spelled out, e.g., in Moore (1980), Soulhi (1984) and Kobsa (1985a). In Kobsa (1985b), a new interpretation is therefore given for this sort of belief and want representation, and new epistemological primitives (the so-called *'acceptance attitudes')* are added to increase its expressive power for more subtle belief and want types.

# 4. Exploiting user models

## 4.1. Exploitation outside of the system

In most systems, user models are applied as aids which help the system in pursuing a particular set of dialog goals or to behave cooperatively in performing a task together with the user. However, there are conversational settings where the building up of a user model is itself the *central* dialog goal (as is the case for, e.g., exams or personnel interviews) and not just a means to an end (in these cases, the exploitation of the user model is left to human interpreters).

An early proposal for such a distinguished function of a user model in an AI system was the NL dialog system ULLY (cf. Hayes & Rosner 1976). The conversational setting for ULLY was a fictitious cocktail party at which both the user and the program were "guests"

The main goal of ULLY was to find out as much as possible about the user. ULLY was designed "to be a friendly and reasonably co-operative conversational partner, who is politely nosey" (cf. Hayes & Rosner 1976, p. 138).

The authors assumed that at the beginning of the conversation, ULLY has a certain amount of knowledge about the other party guests, such as their names, what they do, how they are related to each other (e.g., X detests Y), but does not know anything about the user. At the end of the session, ULLY should have a more or less refined idea of what the user is like.

ULLY has never been implemented, since - we think - some of die project's goals were overambitious with regard to the primitive state of the art of user modeling in those years. Although we have seen some advances in user modeling during the last ten years, even today some of ULLY's maximes point to open research questions, as e.g.

- Try to increase the level of intimacy without being impolite.

- Try to keep the conversation at a level which is understandable.

- Treat assertions by the user with "care".

## 4.2. The relevance of user modeling in various applications of natural-language dialog systems

If a system is designed as a cooperative consultant who actively recommends X (e.g., a book in GRUNDY or a hotel room in HAM-ANS) or advises the user against X, a model of the user is a necessary prerequisite for performing the corresponding speech acts (cf. Morik 1984). A dialog system which generates only fixed and biased recommendations, on the other hand (e.g. a version of GRUNDY which always suggests the Bible first) has hardly any need for a user model. The following list of some types of conversational behavior relevant for AI applications is ordered according to an increasing need for a user model:

(A) question-answering and biased consultation

(B) cooperative question-answering

(C) cooperative consultation

(D) biased consultation pretending objectivity.

A system that responds to the literal meaning of simple questions about a flight schedule or a hotel reservation system which always offers the most expensive room first are examples of category (A). Such systems must possess limited user models only, e.g. for handling

discourse phenomena like anaphoric reference and ellipsis appropriately. Though all systems of this category implemented to date are simple question-answering systems, it is highly probable that biased systems will be installed as soon as the technology of NL processing gets further commercialized. The focus of current research is shifting from systems of category (B) (e.g. CO-OP, cf. Kaplan 1979, 83) to category (G) (e.g. ARGOT, cf. Allen 1982; UC, cf. Wilensky et al. 1984). Systems of both of these categories include user modeling techniques for the generation of various forms of over-answering (cf. Wahlster et al. 1983) and for the generation and recognition of (indirect) speech acts (cf. Allen 1979; Appelt 1982, 85). Systems of category (D), which mimic e.g. professional salesmanship, require a very sophisticated form of user modeling. The experimental dialog system IMP, which will be discussed below, includes some aspects of that kind of dialog behavior.

As was pointed out by Rich (1983), user modeling is especially important if the dialog system is used by a heterogeneous group of users and if the system shows some flexibility in what it tells the user. A typical example for this is her GRUNDY system which, in playing the part of a librarian, can suggest from a large number of books one that it assumes the user might like to read. The selection is made on the base of a model of the particular user which the system constructs in the course of the dialog.

Another example of a dialog system for a heterogeneous user group is HAM-ANS which, in its hotel reservation application, plays the part of a hotel manager who tries to rent all rooms available in his/her hotel. On the basis of an explicit model of the individual user, the system can select one of four room categories and suggest a vacant room in the selected category. In this conversational setting the user is assumed to have the overall goal of getting a hotel room which meets his/her requirements.

From a greeting like that in Fig.5 (which includes the name and optionally the title, the affiliation and the home or office address of the user) and from further booking information like that provided by the user in Fig. 5, the system can derive some assumptions about the prospective hotel guest, e.g. his profession, the purpose of his trip (business trip, vacation, etc.) and his solvency. Several stereotypes are associated with each of these dimensions.

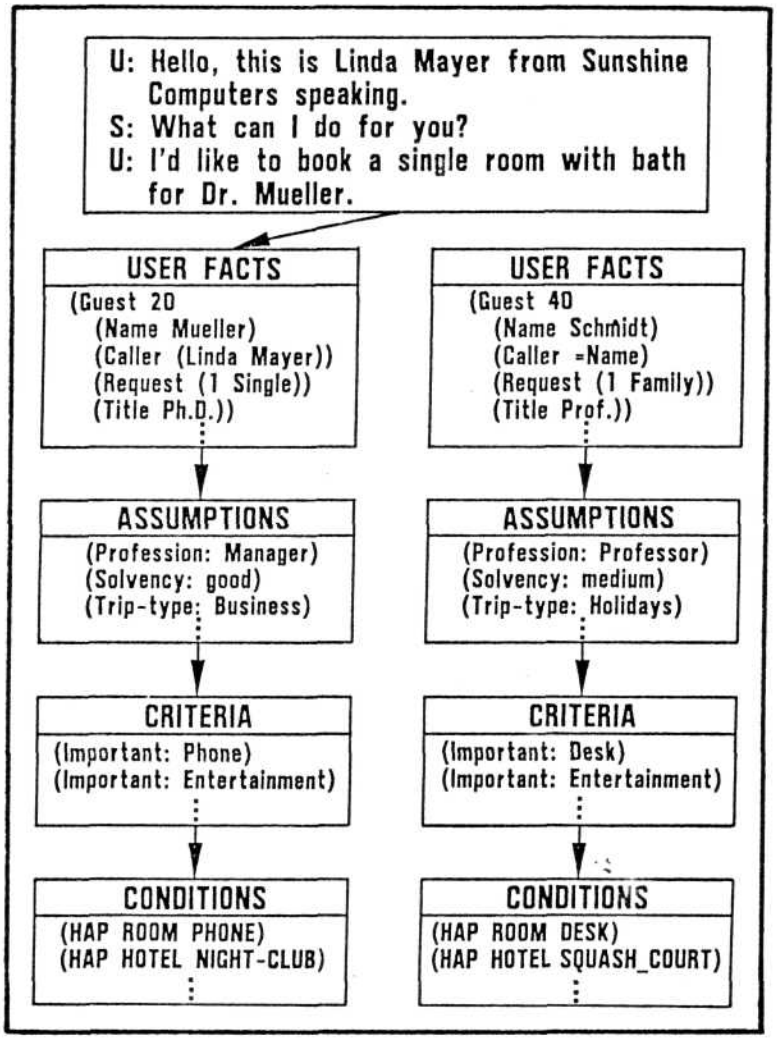


*Fig. 5:* The use of stereotypes in HAM-ANS

As in GRUNDY, stereotypes are used as collections of frequently recuring characteristics of users. In the example above, HAM-ANS may assume that Mayer is the secretary of Mueller and that Mueller is a manager in a high tech company ("Sunshine Computers"), since he holds a Ph.D. and has a secretary. Furthermore, the system assumes that Mueller is on a business trip and, because his expenses are reimbursed, he is able to spend a lot of money for a room. The user-provided information relevant for the consulting procedure is stored as factual knowledge in the user model, whereas the stereotypes triggered by these facts are stored as hypotheses, which can be withdrawn if they are disconfirmed by the subsequent dialog. Note incidentally that if the caller is booking a room for someone else, the part of the "user" model discussed here does not contain information about the actual user of the system, but about the prospective hotel guest instead. So, to use Sparck Jones' (1984) terms, the agent and the patient (the person who is talked about in the discourse) are not identical in this example. One can imagine future dialog systems which also model the intermediate persons (e.g., for also taking into account private interests of secretaries), but this is beyond the scope of our discussion.

HAM-ANS can characterize a particular hotel guest by several disjoint stereotypes. E.g., its assumptions about a user named Schmidt may be that he is a university professor who wants to spend his vacation together with his family, and that he is not able to spend much because he has five children. As soon as the system disposes of all necessary booking facts, it checks whether there is a vacant room which meets the requirements explicidy stated by the user (e.g., period of stay, number of beds). If several rooms belonging to different categories meet these explicit requirements, the system uses the activated stereotypes to predict further demands of the user. For example, HAM-ANS may assume that for Mueller, as a manager on a business trip, a room telephone is very important. Prof. Schmidt on the other hand, being a scientist, actually needs a large desk even on his vacation, but in this particular situation a room telephone is not so important for him. These stereotypical inferences generate a list of criteria rated by different importance values. The set of derived criteria is matched against the available room categories in order to generate a strong or weak recommendation and to indicate which criteria are fulfilled and which are not. However, since individuals differ in the conditions they impose for the fulfillment of a criterion, this matching process again depends on the user profile (cf. Morik & Rollinger 1985). For example, if the guest plans to stay for a longer period in the hotel, attractive entertainment facilities become very important. This criterion, of course, is evaluated differently by a businessman who travels by himself compared with a professor who spends his vacation together with his family in a hotel.

## 4.3. Anticipation feedback

A special case of exploitation of a user model (which is realized in dialog systems like HAM-ANS, NAOS and IMP) is its use in various forms of *anticipation feedback loops.* Anticipation feedback loops involve the use of the system's language analysis and interpretation components to simulate the user's interpretation of an utterance which the system plans to verbalize. The application of anticipation feedback loops is based on the implicit assumption that the system's procedures for language analysis and interpretation (but not necessarily the contents of the knowledge sources used by these procedures) are similar to those of the user. Such a similarity assumption seems to be quite common among participants in everyday conversations. Since most people (except, of course, some computational linguists and AI researchers) cannot be assumed to have an exact model of the understanding capabilities of their dialog partners (except for some naive assumptions concerning small children and non-native speakers) it is highly plausible that they use the similarity assumption as a default rule. Although the current state of the art in NL dialog systems in no way provides sufficient evidence for assuming a far-reaching similarity, user modeling based on anticipation feedback loops can considerably improve the quality of present dialog systems.

### 4.3.1. Local anticipation feedback for ellipsis generation

The ellipsis generation component of the dialog system HAM-ANS incorporates a local anticipation feedback loop which helps to ensure that the system's utterances are not so brief as to be ambiguous or misleading (cf. Jameson & Wahlster 1982). Consider for example the user's question (11).

(11)User: Do five colleagues have more than three weeks vacation?

If five colleagues have in fact not more than three, but more than two weeks vacation or if only two colleagues have more than three weeks vacation, a cooperative system might respond with (12), instead of simply answering 'No'.

(12)System: No, two.

By using its ellipsis resolution component in an anticipation feedback loop, the system can recognize that (12) would be a confusing response since it is so elliptical that, without further context, no unambiguous interpretation can be assigned to it. Acceptable responses like (13) and (14) include more overlap with question (11).

(13)System: No, two weeks.

(14)System: No, two colleagues.

Having rejected (12), the system internally generates the less elliptical responses (13) or (14), depending on the intended meaning. This time, the anticipation of the user's understanding causes a positive feedback, so that the elliptical structure can now be mapped onto a well-formed NL utterance by the surface transformation component of the system (for a more complete account of HAM-ANS's user modeling in ellipsis generation cf. Jameson & Wahlster 1982).

Fig.6 shows an extremely simplified version of the general structure of a dialog system with an anticipation feedback loop for user modeling. The parsing of an input results in the semantic representation construction SR-A1, whose evaluation yields SR-G1 as the semantic representation for the intended response. The result of a preliminary verbalization process for SR-Gl (which includes the above-mentioned ellipsis generation) is fed back into the system's analysis component, which is equipped with an ellipsis resolution component. The semantic representation construction SR-A2, which forms the result of the ellipsis resolution process (and which may consist of a disjunction if the input is ambiguous) is then compared with the intended meaning of the response, namely SR-Gl. If SR-A2 matches SR-Gl, the structure fed into the analysis component is transformed and finally outputted, otherwise an alternative verbalization of SR-Gl is checked in the next iteration of the feedback process.

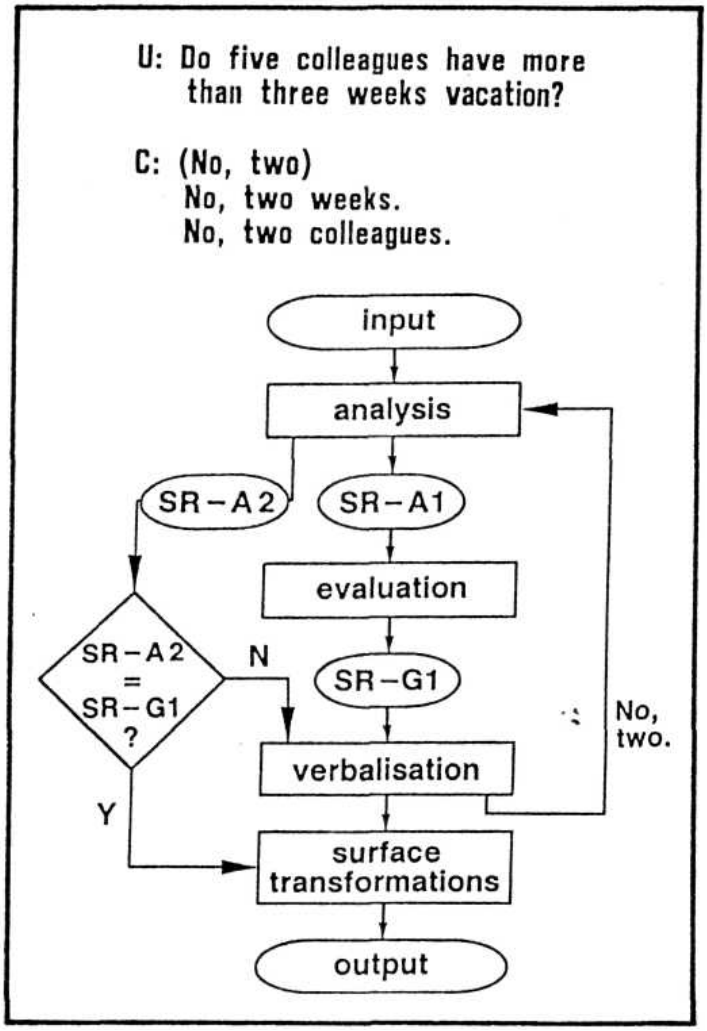


*Fig. 6:* A simple anticipation feedback loop

The anticipation process discussed above is based on a local feedback loop, i.e. one in which the generation procedures of one particular aspect of an utterance are linked with the recognition procedures for the same aspect.

A variety of phenomena, such as the generation of definite descriptions, pronouns, deictic expressions and noun-noun modifications can be handled in a dialog system either by a corresponding multiplicity of local feedback loops (cf. Wahlster et al. 1983, Marburger & Wahlster 1983) or by a larger one within which the whole generated utterance would be tested once, just before applying the final surface transformations (see Fig.3). Such global anticipation feedback loops require dependency-directed backtracking, so that the system returns to the adequate decision point in the generation process whenever the result of the feedback process is negative.

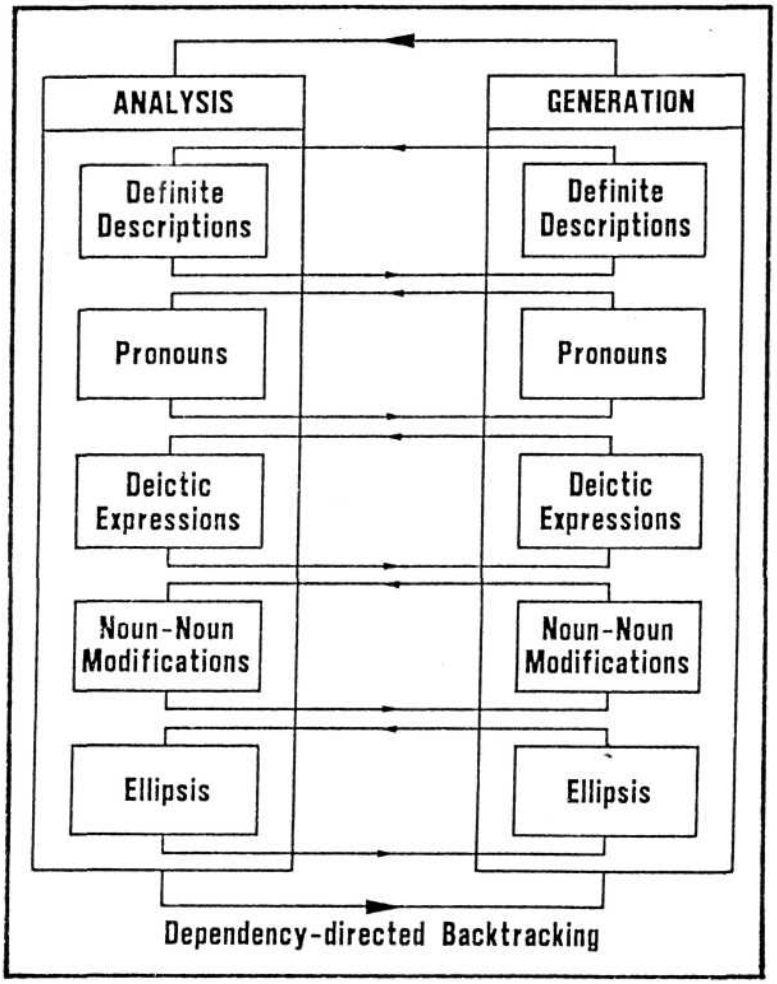


*Fig. 7:* Local and global anticipation feedback loops

### 4.3.2. A global anticipation feedback loop

An example of a more global anticipation feedback loop is the imagination component of the system NAOS, which generates NL descriptions of real-world image sequences. Novak & Neumann (1984) describe the design of a user model for NAOS, which contains a representation of the imagined scene as a result of an anticipation of the user's understanding of a planned utterance. Such a system would take into account the present state of the dialog and the assumed knowledge and dialog goals of the user. On this base, the representation of the imagined scene (which always contains some uncertainty) and the actual scene observed by the system must be matched in order to check whether the planned utterance will be interpreted by the user in a consistent manner and whether it will generate the intended perceptual impression on an adequate level of detail. For instance, it is quite clear that the image sequence of a car accident in which one was involved is described differently to the police, the lawyer of the adverse party, or a good friend.

This application shows that a user model cannot completely rely on representation schemes only suited for the representation of discrete domains. Instead, they must also

provide means for the representation of continuous domains, e.g. of the fuzzy regions of an object's trajectory which might form part of the imagination anticipated for a planned utterance like (15).

(15) A red BMW passed a small VW.

### 4.3.3. Anticipation feedback in evaluation-oriented discourse

Jameson (1983) has studied a very advanced form of anticipation feedback for user modeling in his IMP system. IMP was designed to answer the questions of a user whose sole dialog goal is assumed to be the assessment of some object. On the basis of explicit assumptions about the user's standards and prior expectations, IMP goes beyond the literal answering of questions by adding unsolicited comments that have consequences for the evaluation of the object under discussion. In evaluation-oriented dialogs (e.g., personnel selection interviews, dialogs with travel agents, hotel managers or other salespeople) the selection of such comments is influenced by the nature of the speaker's bias (e.g., a desire to present something in a favorable light). Since listeners take these facts into account when interpreting such comments, IMP anticipates the interpretation of the user while taking into account the bias which the user possibly ascribes to it. IMP's actual bias can be positive, negative or objective, and it can attempt to maintain a positive, negative or objective image. The actual and projected bias may diverge (e.g., usually a salesman's actual bias is positive though he attempts to maintain a neutral image).

On the representational level, IMP's user model includes an explicit record of the user's expectations concerning a particular room offered for rent. This record is updated according to the user's anticipated interpretation of the system's answer. The most interesting procedural aspect of IMP's user modeling is an anticipation feedback loop (cf. Jameson & Wahlster 1982) which takes into account the effect of a planned utterance on the user's impression of the room and helps to filter out comments which would betray the system's bias.

IMP's user model is also used to select expressions like 'but', 'by the way' and 'unfortunately', which precede the additional comments generated by the system. An example is given in (16):

(16) User: What about facilities outside the room?

System: There's a kitchen and there's a bathtub and by the way, the room is very large.

User: Is the room on a quiet street?

System: No, in fact unfortunately there's a lot of noise from the street.

(Jameson 1983, p.618)

For selecting such connectives and sentential adverbs the system compares the shift in

the expected value produced by the present comment with that produced by the preceding statement. IMP shows that for an explanation of the complex semantics and pragmatics of such expressions a user model is necessary.

As we have shown, the use of an anticipation feedback loop is an important part of the process of designing a successful utterance. In essence, anticipation on the part of the speaker means answering question (17) (cf. Rosner 1981).

(17) If I heard this utterance relative to a set of beliefs B and goals G, where B and G are what /assume the hearer believes and wants, Respectively, then what would be the effect on me?

It is obvious that a dual question like (18) is useful for the hearer's analysis of utterances.

(18) If 7 had said this utterance, on the basis of what beliefs and goals would I have said it?

## 4.4. Some social problems in exploiting user models

If the user is *aware* that the system with which s/he is interacting is equipped with a user modeling component, then s/he has the chance to apply all the techniques for hiding his/her beliefs, goals and plans which s/he constantly uses in every-day person-to-person interaction. This includes, e.g., pretending that one knows less, or more about a domain, hiding one's goals and plans, beating about the bush, etc. In some applications (e.g., tutoring or unsolicited advice) the user is necessarily aware of the existence of a modeling component, in other applications its existence might not be so apparent to him.

*Control* of the user over the user modeling component means, e.g., that the user can inspect the model which the system has made of him/her in the course of a dialog. However, since the number of assumptions made by the system in a normal interaction can be expected to be enormous (cf. section 2.3.), the user has no chance to inspect them all. Another problem is rendering these assumptions comprehensible to him/her. The representational categories employed for user modeling often do not correspond to the usual "folk-psychological" belief and goal categories. Thus the contents of user models are often very difficult to translate into ordinary English.

Control of the user over the user modeling component may also mean that the user is entitled to alter the assumptions that s/he has inspected. In this case, as well, the problem arises of how to explain the consequences of such alterations to the user. One might argue that such an option might seduce him/her to change actually correct assumptions in order to make them correspond to his/her (more positive) self concept. In the case of cooperative systems and computer-based tutoring, however, it is the user's problem if *the* system's cooperativity declines as a result of these manipulations in the user model.

Another interpretation of a user's control over a system's user modeling component is that the user has the possibility to "switch off" this component and to communicate with the "unintelligent" remaining system alone, if s/he does not consent to being modeled by the system. In some cases (e.g., tutoring), this is not possible due to the nature of the application. In other cases (e.g., cooperative information), it is certainly not advisable to do so. However, there are also applications (e.g., unsolicited advice) where such a demand of the user may be justified. The technical question remains, however, whether it is possible to separate the user modeling component from the remaining system in such a way that it is possible to "switch it off".

User modeling therefore involves the risk of *misunderstandings* (as is also the case in normal human communication). It can be expected, however, that the gain in cooperativity will greatly exceed that risk. Often it is only this cooperativity which enables the casual user to gain access to the system at all. In traditional dialog systems, by the way, there is also the problem of misconception, however only on the side of the user (e.g. about the meaning of some command, which, when undetected, may lead to a wrong interpretation of the results.) In any case, a necessary characteristic of a user model must be that the assumptions stored in it be revisable if counterevidence is observed. Recent work on belief revision (Doyle & London 1980) seems to be an interesting onset in this direction.

# 5. Open questions and future research in user modeling

## 5.1. Coping with collective beliefs

In the research discussed so far, user modeling was restricted to models of individuals. However, as Wilks and Bien (1983) have pointed out, it is necessary in some dialog situations to model the beliefs of organizations (e.g., NSF's view of AI research), of states (e.g., Germany's view of the European Community), or of classes of individuals (e.g., the sales division's view of AI). Sometimes in a conversation, an individual (e.g., a sales representative, a district attorney, a diplomat) has to advocate the beliefs and goals of such a collective third party without refering to his/her personal beliefs or goals. Thus it may be reasonable for a cooperative NL interface to a commercial DBMS to restrict the dialog-independent user-model e.g. to the point of view of the sales division, the research division, or the production division of a company (the view mechanism in DB technology may be seen as a primitive precursor of such a group-oriented user modeling scheme). Based on the assumption that an individual user inherits all beliefs and goals from the division s/he is working for, a simple user model may in such a professional environment ignore the "private" beliefs of the dialog partner, and model only the collective ones.

However, for most conversational settings, this is an oversimplified approach since often in the course of the dialog "private" beliefs and "inherited" beliefs are mixed (see Fig.8). For

example, a sales representative may begin a conversation with a client by referring to a "private" belief like (19), but later on express the beliefs and goals of his employer, as in (20).

(19) Last time we met we had an excellent dinner together.

(20) This product is going to be a big seller.

Here the open question is how the system can decide which beliefs are to be regarded as "inherited", and therefore to be propagated to that part of the knowledge base which is reserved for the representation of collective beliefs, and which ones are to be stored in the individual user model. Sometimes even a human listener has difficulty coming to a decision and therefore must ask a meta-question like (21).

(21) Is this your personal opinion or an official statement of your company?

Although our current techniques for plan recognition and speech act interpretation have to become much more refined before we can cope with this problem, a future solution may have great practical importance, e.g. in consultative and argumentative dialogs and in knowledge acquisition dialogs with expert systems

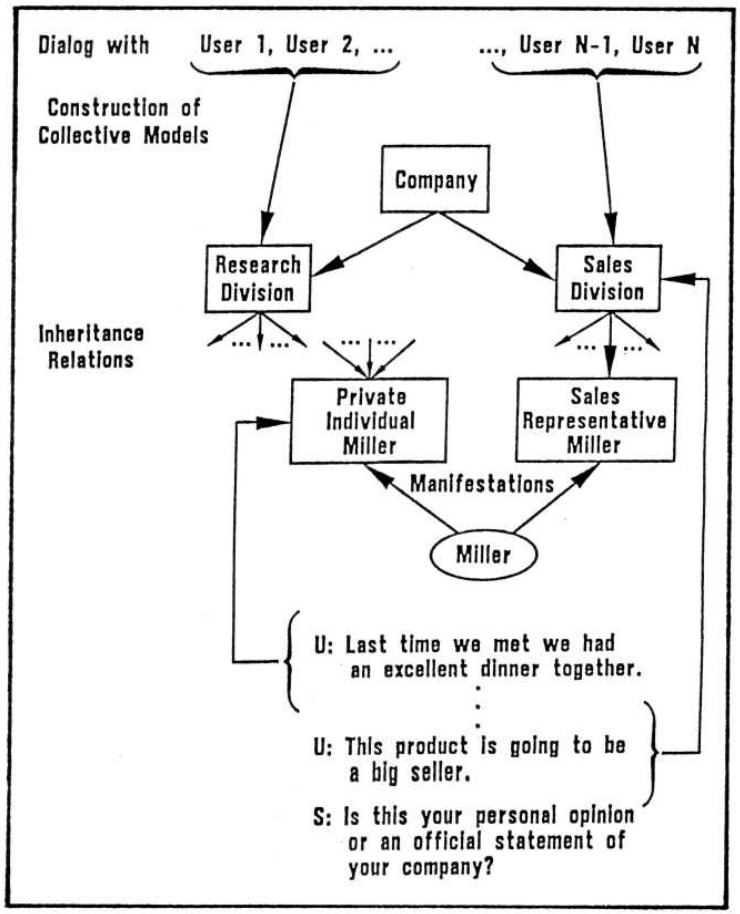


*Fig.8:* Coping with individual and collective beliefs in a user model

Another open question related to the previous one is: How can the system construct, maintain, and update consistent permanent models of collective beliefs on the basis of dialog sessions with individual users? At least for the sake of storage economy, it is necessary for a dialog system with heterogeneous user groups to identify consistent sets of beliefs and construct inheritance networks of beliefs. A solution to this problem must rely on techniques for machine learning, especially inductive inference and stereotype formation.

## 5.2. Adaptability to diverse conversational settings

Today, an important trend in the design of NL dialog systems is the development of transportable systems. HAM-ANS (cf. Hoeppner et al. 1983), TEAM (cf. Martin et al. 1983), LDC (Ballard et al. 1983) are examples of transportable systems that can be adopted to different domains by the experts in these fields (rather than by AI or linguistics specialists).

In Hoeppner et al. (1983) we proposed to go beyond domain-independence by creating dialog systems adaptable to applications that differ not only with respect to the domain of discourse, but also to dialog type, user type and intended system behavior: *transmutable systems* as we call them (see Fig. 9).

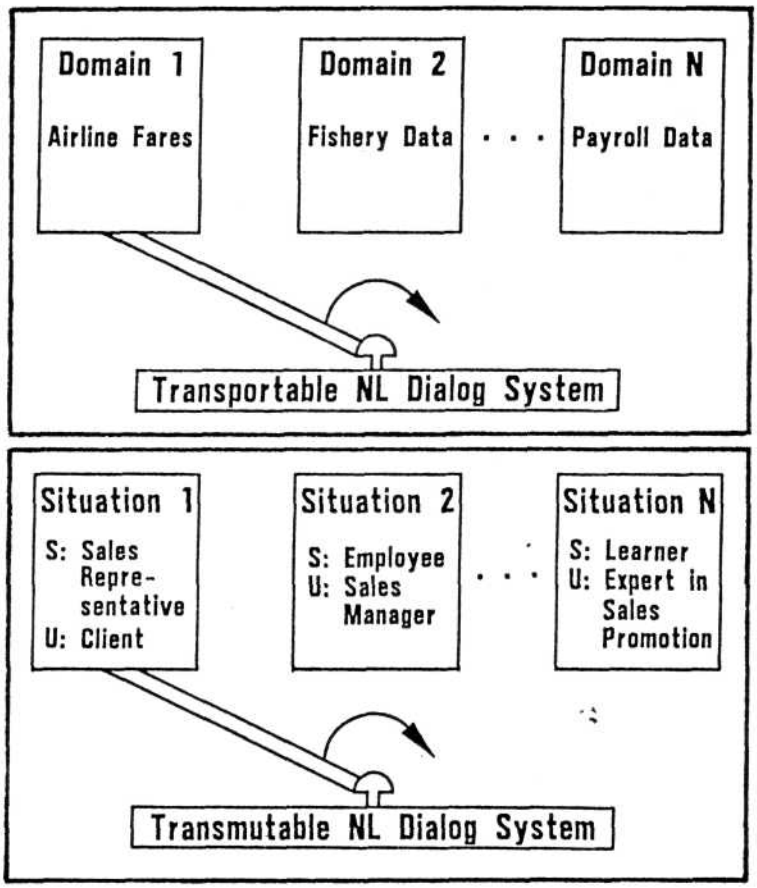


*Fig.9:* Transportable vs. transmutable systems

Such transmutable systems are transportable and adaptable to diverse conversational settings and dialog games (cf. Levin & Moore 1977). As a fictitious scenario for the application of a transmutable system let's suppose that the system's main task is to play the part of a sales representative for a particular company. Sometimes the sales manager interviews the system concerning its job performance. If s/he is dissatisfied with the system's performance, s/he asks an expert in sales promotion to instruct the system.

If you recall the DEC system family XCALIBUR, XSEL and XCON, you will notice that such a scenario is not too unrealistic: XCALIBUR (Carbonell et al. 1983) is a natural-language interface to XSEL, an automated salesman assistant which advises a customer on the selection of appropriate VAX components. XSEL then produces a sales order and passes it on to the XCON expert system for an automatic configuration of the ordered components.

Nilsson (1983) proposed the development of a class of computer systems that are never to be turned off. A dialog system with continuing existence seems to provide a good framework for basic research on transmutable systems. Such a system should have a constantly changing model of its various users and should be able to adapt to diverse conversational settings and to take on various communicative roles.

The major problem that developers of transmutable systems are confronted with is the lack of a representational vocabulary for the declarative description of the relationship between the system and the user, the system's intended dialog behavior, and the associated conversational tactics. Although some of the research discussed in this paper tackles aspects of this problem, a general solution to this question requires lengthy future research on user modeling.

## 5.3. User modeling as a resource-limited process

Before asking one's supervisor for a raise in salary, one usually prepares oneself for this dialog by analyzing mutual beliefs and by anticipating the various possible plans and goals of the dialog partner. But in most everyday conversations, the evaluation of complicated embedded, reflexive, or mutual beliefs and goals as well as a detailed plan analysis is a luxury which one may often not be able to afford. Thus, a computationally plausible outline of a user modeling component must provide some built-in processing restrictions that result from resource-limited real-time understanding of spontaneous discourse. Rosner (1981) has pointed out that if one carries the user modeling approach to its extreme, there is a very good chance that nothing will get said at all (or if said, not understood) because the speaker and hearer will spend all of their time simulating each other's roles without actually doing anything else. Obviously this process must stop somewhere, but the question is where. Allen and Perrault (1980), for instance, have recognized that their plan-based approach to user modeling may have to be complemented by a rule-based one. We believe, a promising direction is the compilation

of the results of a detailed plan-based analysis of speech act sequences into more efficient "shallow" conversational rules.

A related question which so far has not been settled is at what time nested beliefs should be constructed. In the approach proposed by Perrault, Cohen and Allen, most of the possible perspectives on beliefs are already considered as computed (e.g., what one believes one's supervisor believes about one's beliefs about the supervisor's beliefs about a salary increase). Wilks & Bien (1982) adopt the opposite point of view and propose in their multiple environment approach as a least-effort hypothesis that nested beliefs should only be constructed when needed.

## 5.4 Prospects for application-oriented research

As can be seen, research on user modeling has been very active in the last few years. A number of interesting proposals have been made concerning representation systems for user models, methods for generating assumptions about the user from his/her natural language input, and strategies for the planning of the system's dialog contributions on the base of its model of the user. Also, several prototype systems have been implemented in order to exemplifiy and further investigate these proposals.

However, a great number of fundamental problems still remain open, some of which have been mentioned in the last few sections. As far as possible applications are concerned, an additional problem arises in that the development of flexible user modeling components is rather costly. Therefore, application-oriented research will probably lead in two directions:

- *Development of user modeling components in help systems for widely used existing systems*

  Such a component would observe the user's interaction with the host system, draw assumptions about the user's plans and experience, and provide user-specific help on request. First onsets in this direction are the help systems for the UNIX (Wilensky et al. 1984) and SINIX (Kemke 1985) operating systems, a.help system for the SCRIBE text editor (Rich 1982), and a system which provides unsolicited advice for users of the VMS operating system (Finin 1983).

- *Development of general user modeling components*

  Another possible direction seems to be the development of general user modeling components, comparable to the development of expert system shells in the last few years. Such components might provide, e.g., a representation scheme for assumptions about a user's beliefs and goals, an inference engine for this representation, a mechanism for detecting inconsistencies in the assumptions and for belief revision, etc. Such a shell could then be filled with domain-specific inference rules as well as domain-specific

default assumptions about any user, and then be integrated into a dialog system. Beginning research in this direction is Finin & Drager's (1985) GUMS system.

## References


Allen, J. F. (1979): A Plan-Based Approach to Speech Act Recognition. TR 131/79, Dept. of Computer Science, Univ. of Toronto, Cda.

Allen, J. F. and C. R. Perrault (1980): Analyzing Intention in Utterances. Artificial Intelligence 15, 143-178.

Allen, J. F. (1982): ARGOT: A System Overview. TR101, Computer Science Dept., Univ. of Rochester, Rochester, NY.

Allen, J. F. and D. J. Litman (1986): Plans, Goals, and Natural Language. In this volume.

Appelt, D. E. (1982): Planning Natural-Language Utterances to Satisfy Multiple Goals. Technical Note 259, SRI International, Menlo Park, CA.

Appelt, D. E. (1985): Planning English Sentences. Cambridge etc.: Cambridge Univ. Press.

Ballard, B., J. Lusth and N. Tinkham (1984): LDC-1: a Transportable, Knowledge-Based Natural Language Processor for Office Environments. ACM Trans, on Office Information Systems 2, 1-25.

Boguraev, B. K. (1985): User Modelling in Cooperative Natural Language Front Ends. In: G. Nigel Gilbert and C. Heath, eds.: Social Action and Artificial Intelligence. London: Gower Press.

Carbonell, J. G. (1983): The Role of User Modelling in Natural Language Interface Design. Report CMU-CS-83-115, Dept. of Computer Science, Carnegie-Mellon University, Pittsburgh, PA.

Carbonell, H. G., W. M. Boggs, M. L. Mouldin and P. G. Anick (1983): XCALIBUR Project Report 1. First Steps Towards an Integrated Natural-Language Interface. Technical Report CMU-CS-83-143, Dept. of Computer Science, Carnegie-Mellon University, Pittsburgh, PA.

Carberry, S. (1983): Tracking User Goals in an Information-Seeking Environment. Proceedings of the AAAI-83, Washington, DC, 59-63.

Carr, B. and I. Goldstein (1977): Overlays: A Theory of Modeling for Computer Aided Instruction. AI Memo 406, MIT, Cambridge, MA.

Clancey, W. J. (1979): Transfer of Rule-Based Expertise through a Tutorial Dialogue. Report STAN-CS-769, Computer Science Dept., Stanford University, Stanford, CA.

Cohen, P. R. (1978): On Knowing What to Say: Planning Speech Acts. TR-118, Dept. of Computer Science, Univ. of Toronto, Cda.

Doyle, J. and P. London (1980): A Selected Descriptor-Indexed Bibliography to the Literature on Belief Revision. SIGART Newsletter 71, 7-23.

Fagin, R. andj. Y. Halpern (1985): Belief, Awareness and Limited Reasoning: Preliminary Report. Proceedings of the 9th International Joint Conference on Artificial Intelligence, Los Angeles, CA, 491-501.

Finin, T. W. (1983): Providing Help and Advice in Task Oriented Systems. Proceedings of the 8th International Joint Conference on Artificial Intelligence, Karlsruhe, West Germany, 176-178.

Finin, T. W. and D. Drager (1985): GUMS: A General User Modeling System. MS-CIS-85-43, Dept. of Computer and Information Science, University of Pennsylvania, Philadelphia, PA.

Georgeff, M. (1983): Communication and Interaction in Multi-Agent Planning Systems. Proceedings of the AAAI-83, Washington, DC, 125-129.

Halpern, J. Y. and Y. O. Moses (1985): A Guide to the Modal Logics of Knowledge and Belief: Preliminary Draft. Proceedings of the 9th International Joint Conference on Artificial Intelligence, Los Angeles, CA, 480-490.

Hayes, P. J. and M. A. Rosner (1976): ULLY: A Program for Handling Conversations. Proceedings of the 1976 AISB Summer Conference, Edinburgh, 137-147.

Hintikka, J. (1962): Knowledge and Belief. Ithica, NY: Cornell Univ. Press.

Hoeppner, W., T. Christaller, H. Marburger, K. Morik, B. Nebel, M. O'Leary and W. Wahlster (1983): Beyond Domain-Independence: Experience with the Development of a German Language Access System to Highly Diverse Background Systems. In: Proceedings of the 8th International Joint Conference on Artificial Intelligence, Karlsruhe, West Germany, 588-594.

Jameson, A., W. Hoeppner and W. Wahlster (1980): The Natural Language System HAM-RPM as a Hotel Manager: Some Representational Prerequisites. In: R. Wilhelm (ed.): GI - 10. Jahrestagung Saarbrücken. Berlin: Springer.

Jameson, A. and W. Wahlster (1982): User Modelling in Anaphora Generation: Ellipsis and Definite Description. Proceedings of the 1982 European Conference on Artificial Intelligence, Orsay, France, 222-227.

Jameson, A. (1983): Impression Monitoring in Evaluation-Oriented Dialog: The Role of the Listener's Assumed Expectations and Values in the Generation of Informative Statements. Proceedings of the 8th International Joint Conference on Artificial Intelligence, Karlsruhe, West Germany, 616-620.

Joyce, C. and L. Wingerson (1983): Can We Adjust to Computer Culture? New Scientist, April 1983, 72-73.

Kaplan, S. J. (1979): Cooperative Responses from a Portable Natural Language Data Base Query System. Ph.D. Thesis, Univ. of Pennsylvania, Dept. of Computer and Information Science. Also published as: Report HPP-79-19, Computer Science Dept., Stanford Univ., Stanford, CA.

Kaplan, S. J. (1983): Cooperative Responses from a Portable Natural Language Database Query System. In: M. Brady and R. C. Berwick, eds.: Computational Models of Discourse. Cambridge, MA: MIT Press.

Kemke, C. (1985): Entwurf eines aktiven, wissensbasierten Hilfesystems fur SINIX. Bericht Nr. 5, AI Laboratory, Dept. of Computer Science, Univ. of Saarbrücken, West Germany.

Kobsa, A. (1984): Three Steps in Constructing Mutual Belief Models from User Assertions. Proceedings of the 6th European Conference on Artificial Intelligence, Pisa, Italy, 423-427.

Kobsa, A. (1985a): Benutzermodellierung in Dialogsystemen. Berlin: Springer.

Kobsa, A. (1985b): Using Situation Descriptions and Russellian Attitudes for Representing Beliefs and Wants. Proceedings of the 9th International Joint Conference on Artificial Intelligence, Los Angeles, CA, 513-515.

Kobsa, A. (1985c): VIE-DPM: A User Model in a Natural-Language Dialogue System. In: J. Laubsch, ed.: GWAI-84, 8th German Workshop on Artificial Intelligence. Berlin: Springer.

Kobsa, A., J. Allgayer, C. Reddig, N. Reithinger, D. Schmauks, K. Harbusch and W. Wahlster (1986): Combining Deictic Gestures and Natural Language for Referent Identification. Technical Report, SFB 314, Dept. of Computer Science, Univ. of Saarbrücken, West Germany (forthcoming).

Kobsa, A. (forthcoming): Generating a User Model from, Wh-Questions in the VIE-LANG system. To appear in the Proceedings of the 1984 GLDV Conference, Heidelberg, West Germany.

Konolige, K. and N. J. Nilsson (1980): Multiple Agent Planning Systems. Proceedings of the AAAI-80, Stanford, CA, 138-141.

Konolige, K. (1981): A First-Order Formalisation of Knowledge and Action for a Multi-Agent Planning System. In: J. E. Hayes, D. Michie and Y-H Pao, eds.: Machine Intelligence 10. Chichester: Ellis Horwood.

Levin, J. A. and. A. Moore (1977): Dialogue-Games: Metacommunication Structures for Natural Language Interaction. Cognitive Science 1, 395-420.

Macmillan, S. A. (1983): User Models to Personalize an Intelligent Agent. Ph.D. Thesis, School of Education, Stanford University, Stanford, CA.

Marburger, H. and W. Wahlster (1983): Case Role Filling as a Side Effect of Visual Search. Proceedings of the First European ACL Meeting, Pisa, Italy, 188-195. Martin, P., D. Appelt and F. Pereira (1983): Transportability and Generality in a Natural-Language Interface System. Proceedings of the 8th International Joint Conference on Artificial Intelligence, Karlsruhe, West Germany, 573-581.

McCoy, K. F. (1985a): The Role of Perspective in Responding to Property Misconceptions. Proceedings of the 9th International Joint Conference on Artificial Intelligence, Los Angeles, CA, 791-793.

McCoy, K. F. (1985b): Correcting Object-Related Misconceptions. Ph.D. Thesis, MS-CIS-85-57, Dept. of Computer and Information Science, University of Pennsylvania, Philadelphia, PA.

McKeown, K. R., M. Wish and K. Matthews (1985): Tailoring Explanations for the User. Proceedings of the 9th International Joint Conference on Artificial Intelligence, Los Angeles, CA, 794-798.

Moore, R. C. (1980): Reasoning about Knowledge and Action. Technical Note 191, AI Center, SRI International, Menlo Park, CA.

Morik, K. (1984): Partnermodellierung und Interessenprofile bei Dialogsystemen der Künstlichen Intelligenz. In: C.-R. Rollinger, ed.: Probleme des (Text-) Verste-hens. Tubingen: Niemayer.

Morik, K. and C.-R. Rollinger (1985): The Real Estate Agent - Modeling Users By Uncertain Reasoning. The AI Magazine 6, Summer 1985, 44-52.

Neumann, B. and H.-J. Novak (1983): Event Models for Recognition and Natural Language Description of Events in Real-World Image Sequences. Report IFI-HH-M-107/83, Dept. of Computer Science, Univ. of Hamburg.

Nilsson, N. (1983): Artificial Intelligence Prepares for 2001. The AI Magazine, Winter 1983, 7-14.

Novak, H.-J. and B. Neumann (1984): Szenenbeschreibung und Imagination in NAOS. In: C.-R. Rollinger, ed.: Probleme des (Text-)Verstehens. Tübingen: Niemayer.

Perrault, C. R. and J. F. Allen (1980): A Plan-Based Analysis of Indirect Speech Acts. American Journal of Computational Linguistics 6, 167-182.

Pollack, M. E., J. Hirschberg and B. Webber (1982): User Participation in the Reasoning Process of Expert Systems. MS CIS-82-9, Dept. of Computer and Information Science, Univ. of Pennsylvania.

Power, R. (1979): The Organisation of Purposeful Dialogues. Linguistics 17, 107-152.

Reiser, B. J., J. R. Anderson and R. G. Farrell (1985): Dynamic Student Modelling in an Intelligent Tutor for LISP Programming. Proceedings of the 9th International Joint Conference on Artificial Intelligence, Los Angeles, CA, 8-14.

Rich, E. (1979a): Building and Exploiting User Models. Ph.D. Dissertation, Computer Science Dept., Carnegie-Mellon University, Pittsburgh, PA.

Rich, E. (1979b): User Modelling via Stereotypes. Cognitive Science 3, 329-345.

Rich, E. (1982): Programs as Data for Their Help Systems. Proceedings of the National Computer Conference, Houston, TX, 481-485.

Rich, E. (1983): Users are Individuals: Individualizing User Models. Int. J. Man-Machine Studies 18, 199-214.

Rosenschein, J. S. (1982): Synchronization of Multi-Agent Plans. Proceedings of the AAAI-82, Pittsburgh, PA, 115-119.

Rosner, M. (1981): Some Thoughts about Actions in Conversation. Proceedings of the Workshop on Models of Dialogue: Theory and Application. Linkoping, Sweden, 105-116.

Soulhi, S. (1984): Representing Knowledge about Knowledge and Mutual Knowledge. 10th International Conference on Computational Linguistics, Stanford, CA, 194-199.

Schuster, E. and T. W. Finin (1983): Understanding Misunderstandings. MS-CIS-83-12, Dept. of Computer and Information Science, Univ. of Pennsylvania.

Sidner, C. L. (1983): What the Speaker Means: The Recognition of Speaker's Plans in Discourse. Computers and Mathematics with Applications 9, 71-82.

Sleeman, D. H. and J. S. Brown, eds. (1982): Intelligent Tutoring Systems. London: Academic Press.

Sleeman, D. H. (1985): UMFE: A User Modelling Front End Subsystem. Int. J. Man-Mach. Comm. (in press).

Sparck Jones, K. (1984): User Models and Expert Systems. Technical Report No. 61, Computer Laboratory, University of Cambridge, Cambridge, England.

Vardi, M. Y. (1985): A Model-Theoretic Analysis of Monotonic Knowledge. Proceedings of the 9th International Joint Conference on Artificial Intelligence, Los Angeles, CA, 509-512.

Wahlster, W., H. Marburger, A. Jameson and S. Busemann (1983): Over-Answering Yes-No-Questions: Extended Responses in a NL Interface to a Vision System. Proceedings of the 8th International Joint Conference on Artificial Intelligence, Karlsruhe, West Germany, 643-646.

Wahlster, W. (1984): Cooperative Access Systems. Future Generation Computer Systems 1, 103-111.

Wilensky, R., Y. Arens and D. Chin (1984): Talking to UNIX in English: An Overview of UC. In: CACM 27, 574-593.

Wilks, Y. and J. Bien (1983): Beliefs, Points of View and Multiple Environments. Cognitive Science 7, 95-119.