The Knowledge Engineering Review

Dialogue management for co-operative knowledge based systems

S. P. STENTON
Hewlett-Packard Ltd, Bristol Research Centre

Abstract

This review presents a brief survey of approaches to describing dialogues with computers which are relevant to improving communication between users and knowledge based systems. One approach to achieving improvements is to exploit the conventions of human communication. The review draws together some of the work on co-operative behavour which reflects these conventions. An analysis is presented which goes some way towards identifying the knowledge required and the issues to be tackled.

1 Introduction

This review will look at research which addresses the problems of supporting co-operative dialogue with software systems. Current software systems typically adopt simple dialogue management techniques which centre around the architecture of the underlying system and a limited model of a generic user. Dialogue context is usually limited to the existence of modes ("hpdesk", "Intray", "edit", etc.). Within these modes utterances are dealt with independently or as part of rigidly defined dialogue structures (e.g. "Set Tabs"). Co-operative dialogue is usually generated via help facilities, browsers, and error messages.

Most of these systems are adequate for the population of users for which they are designed: computer professionals (operating systems, etc.); or frequent users (dedicated systems such as spread sheets, word processors and databases). Such people are willing to invest the time to become familiar with the limitations and unnatural conventions of a dialogue management system.

For infrequent users and users of a particular class of emerging systems (i.e. knowledge based advice giving systems) the limited communication channels offered by current dialogue management systems are prohibitive. To allow more people access to the benefits of both traditional and knowledge based software systems, it will be necessary to accommodate a wider range of responses which exploit the conventions of human communication.

The development of AI technology and its potential for recreating human capabilities has expanded not only the functionality of computer systems, but also the communication channel between system and user. It is now possible to consider the nature of co-operative interaction in a wider sense, i.e. the knowledge sources, assumptions and strategies important for communicating with people.

To exploit the conventions of human dialogue involves extending the analysis of input and output beyond the semantics of a single sentence. Identifying the rules and assumptions which guide general discourse seems, at first glance, fraught with pragmatic difficulties. A closer look suggests it IS fraught with pragmatic difficulties. The promise of tractability lies in the task specific nature of systems and the boundaries they impose. Just as a word processing package doesn't "expect" to be asked about the weather so we should be able to predict useful boundaries to discourse capabilities, whilst remaining within the conventions of dialogue for the task. Not surprisingly, this is the underlying assumption of most of the work in the field, though the extent of the boundaries differ.

2 Approaches to describing dialogue

Before focusing in on task dependent dialogue research I should like to consider some general dialogue research issues. What follows is a brief survey of theoretical approaches to describing dialogue. Some of these are not intended as frameworks for computer algorithms. They are included here because they present different ways of looking at dialogue. This survey will be followed by the main body of the review which will look at research towards supporting co-operative dialogue with software systems.

2.1 Discourse genres

This approach attempts to identify types or genres of dialogue. The basic assumption is that genre dictates structure (Reichman, 1986). The resulting analysis centres around the components of particular discourse genres and the order in which those components appear. That is, the classification of a given dialogue constrains the components that can legally appear and their order. One classification of dialogue types describes the different structures of narratives, descriptions and jokes (Linde, 1980). A narrative for example, typically comprises: an abstract of the story to be told; some orientation information (i.e. where, when etc.); the story; a coda to mark the finish; and finally an evaluation indicating the point of the story (Rumelhart, 1975).

Characterizing dialogue in this way emphasizes the rules within discourse forms but the resulting schemata are usually insensitive to the rules which apply across forms (e.g. the development of themes, or task related nesting of forms). For task related dialogue of the kind required for co-operative systems it is important to understand not only the composition of discourse structures, but also how and when they are used.

2.2 Ethnomethodology

This approach attempts to discover rules of discourse as a social interaction between dialogue participants. A key concept is that of adjacency pairs which describe the relations between utterances within the dialogue (Sacks, Schegloff, and Jefferson, 1974). Examples include a pairing of greetings, a question-answer pair, or an offer and its acceptance/rejection. The focus of analysis is on turn taking and the legal points of transfer between dialogue participants. These govern the introduction, nesting and sequencing of adjacency pairs which describe the development of the dialogue.

Techniques of ethnomethodological analysis may prove useful when addressing the issues of initiative control during expert system consultations. One of the drawbacks of 1st generation expert systems has been the lack of dialogue control offered to the user. The user provides answers to questions on demand with a limited facility to ask questions back. If more control of the dialogue is to be given to users (to input relevant constraints or suggest possible solutions) it may be useful to identify suitable points of transfer and the way it is achieved in naturally occurring consultations.

2.3 Linguistics

Linguistic approaches divide into two camps: Deep structure analysis; and Cohesive devices. The former identifies the underlying structure between objects described in the discourse and uses this information to interpret surface linguistic phenomena such as pronominal reference. Grosz (1977) demonstrated that the resolution of pronominal reference could be explained in terms of the nested structure of the dialogue. The dialogue concerned was an expert telling a novice how to assemble a water pump. The dialogue reflected the nested nature of the task and so comprised a series of nested sub-dialogues. The influence of this deep structure was demonstrated by the fact that ambiguities of reference did not occur between objects in different sub-dialogues. To account for this Grosz proposed a push down stack model of global focus (i.e. objects which are the subject of the current sub-dialogue and therefore legal referents).

The latter approach ignores deep structure altogether in search of surface linguistic devices which maintain the cohesion of the dialogue. Halliday and Hasan (1976) identify five surface devices which are used to achieve cohesive dialogue. These are: reference; substitution; conjunction; ellipsis; and lexical cohesion. Reference includes the use of pronouns and the definite article. Conjunction refers to the conjoining of sentences with words like: "Furthermore" or "In addition". Ellipsis is the omission of parts of a sentence based on its predecessor or earlier parts of the sentence:

d.1 *"Some like it hot. Some don't"*

Lexical cohesion uses the relations between the meanings of words (e.g. between door and key in the following example:

d.2 *"The door will be locked. I'll leave the key with Rico."*

2.4 Analysis of intentions

For linguistic approaches the focus of analysis is the structure of the discourse ("deep" or "surface"). This approach focuses on the *intentions* which generate the discourse. Most famous amongst this tradition is the *Speech act* theory of Austin (1962) and Searle (1969). Communication is viewed as communicating intention. Identification of a speaker's intention is the prime factor in understanding the meaning of an utterance.

A major consequence of this approach has been a shift of emphasis from the analysis of *propositional content* to *illocutionary force* (i.e. knowing *what* has been said is not sufficient, for effective communication it is necessary to know *why* it was said.).

To convey his/her intention the speaker must select an utterance with the correct illocutionary force. This may or may not be reflected in the surface structure of the utterance. The illocutionary force of a particular utterance is the speech act it is intended to convey (e.g. making a promise, requesting something etc.). The asignment of illocutionary force can be obvious from the surface structure of an utterance:

d.3 *"Give me your money"*

or may depend upon the situational context and/or social convention:

d.4 *"I'm sure you would like to hand over your wallet to my handsome friend here with the knuckle dusters".*

More familiar examples of the latter are: "Can you pass the salt?" and "Do you have the time?"

The type of analysis generated by this approach is characterized by Clark and Carlson's analysis of the following line from Shakespeare's Othello:

d.5 *"Come, Desdemona"*

"Othello intends to get Desdemona to understand that he wants her to go with him. He gets her to understand this by getting her to recognize his intention to get her to understand this."

(Clark and Carlson, 1982)

The recognition of intention is clearly important for effective communication but it is not the whole story. Further questions need to be answered to present a complete account of successful dialogue (e.g. How are speech acts selected to convey intention? How do they constrain the interpretation of surface utterance? Where do these conventions come from (social, situational, personal) ?). The answers to these questions may be found within the concepts of other approaches (e.g. mutual knowledge or linguistic convention).

2.5 Mutual knowledge

The failure of Speech act theories to adequately explain how the intention of the speaker is recognized by the hearer has led to the types of circular analyses described above. Mutual knowledge has been proposed as the key to recognition of intent (Clark and Marshall, 1981).

Consider the same situation where Desdemona is a polyglot from the planet Fnic where "come" means "stay". She would be in real trouble without mutual knowledge. Consider the complications if Othello knows Fnician but Desdemona only knows he *might* know Fnician, or Desdemona knows Othello knows Fnician but doesn't know whether he knows she knows he knows ! The same paradox arises concerning knowledge of Desdemona's multi-linguistic talents. Recognition of Othello's intent is no longer as obvious as Clark and Carlson's original analysis would suggest.

The above example not only shows the need for mutual knowledge but also the need for sufficient mutual knowledge to allow the reasoning mechanism which utilizes it, to reach a confident prediction of the speaker's intent. Clearly life isn't as complicated as we've made it for Othello and Desdemona. For every-day communication we share many more assumptions about each other's personal, situational, social and linguistic knowledge. The latter three are usually referred to as conventions (Searle, 1971).

Searle's conventions are rather vague and all embracing. They refer to the knowledge that is required to interpret intention from the surface characteristics of an utterance. "Context" is a similarly vague concept often used to refer to such knowledge. Failure to provide detail is not an oversight by Searle. Such detail is simply not available. Research in this area is only just beginning to make significant progress.

Grice provides one of the most influential attempts to specify pragmatic conventions (Grice, 1975). Conversational implicature is proposed as the key to understanding the intention of an utterance. Unlike Conventional implicature which refers to the conventional meaning of the words used, Conversational implicature refers to what the speaker can imply, suggest or mean. Such implication is recognized by the hearer because of the mutual knowledge that the participants in the conversation will adhere to the Co-operative Principle and obey its related maxims ("Quality; Quantity; Relation; and Manner"). For example:

d.6 *P1* "*Should I hire Jones as a professor?*"
P2 "*He's a great guy everyone likes him*"

It is the belief by *P1* that *P2* is adhering to the co-operativity principle that allows her to make sense of *P2's* answer.

The *Co-operative Principle* states:

"Make your conversation contribute such as is required, at the stage at which it occurs, by the accepted purpose or direction of the talk exchange in which you are engaged."

The related maxims are: Quality; Quantity; Relation; and Manner. Quality and Quantity are given below as they will be referred to later.

Quality
Make your contribution one that is true, specifically: do not say what you believe to be false; and do not say that for which you lack adequate evidence";

Quantity
Make your contribution as informative as is required for the current purposes of the exchange. Do not make your contribution more informative than is required for the current purposes of the exchange.

Put less formally, the four maxims are: Be helpful; don't waffle; stick to the point; and be clear. Though still vague Grice's maxims have succeeded in provoking specific questions to be addressed by any system attempting pragmatic interpretation (e.g. How do you determine what is "relevant"? How is the "direction" of the conversation identified? What are the principles which govern what is helpful?).

2.6 Programming metaphor

This approach to discourse theory is not so much a metaphor as a borrowed technique. The basis of this approach is the view that discourse is a means by which we change each other's mental states using cultural and linguistic conventions. Mental states comprise our own beliefs and goals and our beliefs about the beliefs and goals of other people. These are specified as a set of sentences in a formal language which change over time in response to events. The term belief is used instead of knowledge to distinguish the knowledge an agent has from a God's eye view of the world where knowledge is absolute truth.

Discourse theories are described in terms of axioms which interrelate the beliefs, goals and actions of the participants. Dialogues are described as sequences of "mental states" in the same way that computer programmes have been described as sequences of "computational states". Abstract specification of computer programs in this way, have been shown to be useful for generating reliable software (Dijkstra, 1976). It is hoped that specifying discourse in terms of changes in mental states will have similar benefits (Rosenchein, 1981).

The idea of discourse analysis as a specification of inputs and outputs is an appealing one. It is also clear that if pragmatic inference requires mutual knowledge a representation of mental states is unavoidable. However, there are at least two difficult problems still to be solved: What is an adequate theory of discourse (what are the axioms); and which beliefs are relevant (and so, should be represented)? The work of the ex-Toronto team of Ray Perrault, Phil Cohen, James Allen, and Hector Levesque has been most influential in addressing these problems. The aim of this work is to develop a formal theory of communication by axiomatizing the notion of a rational agent. The term rational is used because it allows a number of simplifying assumptions to be made (e.g. an agent's actions will be consistent with his or her goals and beliefs). This approach attempts to account for discourse level phenomena (e.g. indirect speech acts) within a general theory of the participants rather than developing separate phenomena driven theories.

The description of a rational agent comprises: a theory of action which relates the agent's intentions to his or her actions (actions can be physical or linguistic actions); a theory of intentions which includes commitment and choice (i.e. how are intentions adopted and when are they dropped); and a theory of rational interaction between agents (how do agents communicate to affect each other's beliefs and intentions?). A lot of work has been done towards the development of these theories which are expressed as a weak S5 modal logic. The best account of the research to date can be found in Cohen & Levesque (1987).

A similar approach based on psychological data rather than formal programming techniques describes discourse as the invocation of a set of "Dialogue Games" which are procedures which change the "mental states" of the participants (Levin & Moore, 1977). Procedures are called when their preconditions are satisfied. This can be the direct result of actions or their consequent mental state changes. Another approach describes 'Discourse Moves' (Reichman, 1985) which are also implemented as procedures but which change situational states called "Context Spaces". Once called, both "Discourse Moves" and "Dialogue games" dictate the structure and analysis of the ensuing discourse.

3 Two goals of co-operative dialogue

The previous section presented a brief survey of theoretical approaches to dialogue. Section 4 will describe work which has focused on problem solving dialogues. That is, dialogues between databases and their users or experts and their clients. The aim of this work is to improve the communication capabilities of expert and database systems in order to facilitate the problem solving tasks for which they are used. To achieve this within AI, attempts have been made to exploit the conventions of human dialogue. These have not always been principled. That is, it is unusual for the underlying principles of co-operative communication to be fully specified in the analysis. A common approach seems to comprise: the identification of certain dialogue structures as "co-operative"; and the subsequent development of programming techniques which generate them. This will not be sufficient for the general utility of dialogue management systems. To avoid a dialogue which iteratively chases to ground every conceivable ambiguity, we need to understand the *use* of co-operative responses and not just their generation.

Before describing the different types of co-operative response that have been identified and generated within existing software systems (most of which remain experimental systems) I will identify two goals of co-operative communication. The following review of co-operative responses will then be cast in the light of these goals to provide a clearer insight into their underlying principles and the extent of the knowledge required to support them.

The two goals I should like to suggest as underlying discourse during co-operative problem solving are: Task Directed Co-operativity (TDC) and Belief Maintenance Co-operativity (BMC). The aim of the former is to facilitate an efficient performance of the problem solving task. The aim of the latter is to foster the mutual representations between dialogue participants essential for effective communication.

TDC relies on an identification of the user's goals and a knowledge of the domain in which those goals are achieved. Co-operative responses are usually based on: "essential requirements for doing things that way"; or "a better way of doing things"; or "possible consequences of doing things that way". Work in this area includes Pollack's (1985) on "enablement" and "generation", Miller's (1985) on "Critiquing", Allen and Perrault's on plan recognition (1980).

Pollack's work focuses on the relations between the goals stated by a client and those addressed by the expert. Enablement and generation are two such relations. The assumption being that the client does not always pose the query appropriate to his or her problem. Miller's Attending system allows the user (an anaesthetist) to generate a plan through a series of decision points and then provides a critique of the plan based on the consequences of the choices made and known constraints. Allen and Perrault's system uses knowledge about the plans users may have. Once the user's plan has been identified the system can go beyond literal responses to provide more co-operative ones. For example:

d.7 *P1.* "What time does the train to Toronto leave?
P2. "3.30 from platform 2"

In this example the system has recognized that the questioner's most likely plan is to catch the train to Toronto. It is therefore able to give information extra to that requested, to aid in the execution of this plan.

Belief Maintenance Co-operativity (BMC) requires a more general knowledge of the rules of co-operative communication. Co-operative responses are usually a reflection of the desire to: achieve common understanding; and avoid giving misleading information. This typically involves keeping a check on the implications and assumptions which affect and reflect the state of the hearers beliefs. This can be done explicitly (e.g. by including clarifications *such as*: "by that I mean . . ."; ". . . for example . . .") or implicitly (e.g. by the inclusion or omission of words which imply generality ("Today the fastest way to London is by car", "The fastest way to London is by car"). It is easy to identify these co-operative techniques, but knowing when to use them is not so easy.

4 Co-operative dialogue with software systems

What follows is a survey of co-operative responses which have been identified in naturally occurring dialogues and suggested as useful for communicating with software systems. They will not fall cleanly into belief maintenance and task directed responses. Some will contribute to both goals. However, the order in which they are described will reflect a change in contributory bias from belief maintenance to task directed responses. The following categories are based on those described by Finin et al. (1986) with the addition of "Co-operative search" (4.6) and a new perspective.

4.1 Paraphrasing

It is easy to see the benefit of paraphrasing in artificially induced situations such as database query specification in SQL, claiming Supplementary Benefit, or making a statement to the police. In such cases it is clearly important that the details are correctly represented by the hearer. What is not so obvious is that paraphrasing is a co-operative response commonly used in every-day situations (even more so in advice giving situations). As a tool for co-operative communication paraphrasing can serve three purposes:

1 to demonstrate the ambiguity of an input, in an attempt to elicit the intended interpretation;
2 to demonstrate *how* an input has been interpreted; and
3 to demonstrate that an input *has* been understood.

Most of the work to date has been in response to the requirements of building a natural language front-end for database query (McKeown, 1983; Muekstein, 1983; Boguraev and Sparck Jones, 1984). Paraphrases are constructed to demonstrate how a database query has been interpreted from the natural language input. Incorrect interpretations may also serve to demonstrate ambiguity in the original request. The user is thus provided with a check before the database is searched.

There are two approaches. In the first, paraphrases are based on syntactic analysis of the natural language form of the db query. From this it is possible to derive: the assumptions underlying the query; a statement of the "core" of the query; and finally its modifiers. For example, the paraphrase below might be generated in response to the question:

> d.8 "*Which managers in the knowledge based programming department manage projects with an interest in dialogue?*"

paraphrase:

> "*Assuming there are projects with an interest in dialogue; which managers manage those projects; searching only those managers in the knowledge based programming department.*"

The second approach uses information about the interpretation of the query. Paraphrases are constructed on the basis of the syntactic structure of the derived query language form (e.g. take the SQL form of the query, generate its SQL parse tree and map this back into an English expression).

It is not clear how useful the ability to paraphrase user input would be beyond the requirement for database query checking. The effect of not providing this type of feedback is as yet undocumented. As a window onto a system's interpretation of the user's situation it would be useful to provide a paraphrase during advice giving consultations. The problem of when to provide control is an important one as overuse can cause users to be irritated by its intrusion or to ignore it completely.

Analysis of naturally occurring consultations may identify points in the dialogue where a paraphrase is essential to efficient communication. If specific dialogue contexts cannot be formally identified it may be useful to paraphrase at the request of the user (e.g. users familiar with database query may on occasion, want the benefits of the natural language interface without waiting for a paraphrase to be constructed). The trade-offs of computational expense vs benefit to the user would have to be applied.

4.2 Clarifying terminology

In naturally occurring dialogues clarification of terminology provides a means of steering towards the mutual representations necessary for efficient communication. As a co-operative response for knowledge based systems it serves a similar function by providing information about the view(s) of the domain captured by the systems representation(s).

For current systems the two way negotiation of mutual representations is constrained by the relative inflexibility of the system's representation. However it has been demonstrated in the previous section that useful clarification can be supported within these constraints (McKeown, 1983).

At one level clarifying terminology is trivial and at another it is a black hole. At the trivial level it involves word swapping or synonym generation which allows the system and user to cover any surface cracks which appear in their "mutual understanding" (Hendrix et al., 1978). At the "black hole" level it involves a deeper negotiation of common abstractions which make up that mutual understanding. The problem stems from the probability that we do not hold common conceptions of the world. The abstractions which make up our representation of it are time, culture, occupation, situation, intention, (etc.) specific.

Hopefully it is possible to dance round the edge of this black hole by choosing an appropriately constrained domain and so avoid the need for negotiation of fundamental levels of abstraction. This optimism stems from a framework from the analysis of domains by Kidd and Sharpe (1986). The following is a brief precis of the points relevant to this discusssion. For a comprehensive account the reader is referred to the original paper.

There are subsets of world knowledge which are (relatively speaking) highly formalized at a particular level of abstraction for a particular subsection of the population (e.g. mathematics). Thus, individuals who belong to this subsection of the population are likely to hold representations of this knowledge constructed from common abstractions. The maintenance of similar representations during dialogue is therefore less likely to involve a negotiation of fundamental concepts.

There also exist subsets of knowledge which are barely formalized at any level of abstraction for any subsection of the population (e.g. personnel management). Within these areas it cannot be assumed that individuals hold common abstractions. These areas are sometimes referred to as "woolly" (self referential term!).

Maintenance of mutual representations within these areas can require clarification at more fundamental levels. It is also difficult to predict the level at which clarification is required.

By choosing domains within the former category it will be possible to address the problems of providing co-operative responses without confronting (amongst other difficulties) the difficulties of supporting nested clarification (negotiation of fundamental concepts).

There are two approaches to the maintenance of mutual representations through clarification of terminology: One is to "reform the system" by allowing the user to change the system's representation; the other is to "reform the user" by presenting the system's representation in the hope of bringing that of the user "into agreement". The inflexibility of the system's representation usually limits the former to synonym generation. The latter is not so constrained and so has scope for the development of non-trivial solutions.

The most significant work on clarification as a negotiation of meaning is the "TEXT" system (McKeown, 1985) which generates definitions of the terms used in a database. The domain (US Naval systems) falls within the formal language category (above). The approach is "reform the user" by providing definitional information constrained by the abstractions in the database.

The significance of this system lies in its ability to compose clarifications using dialogue strategies appropriate to the goal of the discourse. The basic units of these dialogue strategies are "Rhetorical predicates" (e.g. "Analogy", "Constituency", "Compare/contrast"). The communicative goals considered are: Requests for definitions; Requests for differences between entities; and requests about the information that is available in the database. The database's dictionary information is augmented by meta-level knowledge in the form of an Entity Relationship Model (ERM) and response construction strategies are implemented as Augmented Transition Networks (ATNs).

To respond to a question TEXT decides what information is required (this defines the "Relevant Knowledge Pool") and which strategy (sequence of predicates) to employ. These are important considerations for the generation of any co-operative response.

4.3 Correcting misconceptions

In order to foster mutual representations between dialogue participants it is important to act upon perceived disparities. That is, it is important to identify who holds the misconception causing the

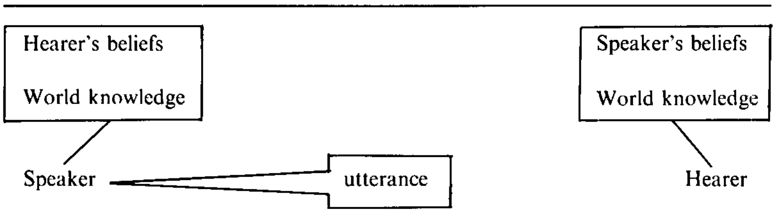

Figure 1 A simple model of the beliefs of two dialogue participants. World knowledge in each case, is equivalent to the holder's beliefs which may include knowledge common to both.

disparity. This decision must depend on the value of the different representations unless one of the participants is a dogmatic know-all (who actually does know-all). The popular view of information giving systems is that they should adopt this dominant role. Thus, approaches to correcting misconceptions have emphasized the detection of the disparity and correction of the user. For this role to be appropriate remedial action is usually limited to giving information about the system itself or passing on "expertise" (where the definition of the roles facilitates the assumption of a dominant representation). Utterances are usually based on two broad categories of facts: those about the receiver; and those about the world. A simple model of the beliefs of two participants in a dialogue (speaker and hearer) is illustrated in figure 1. This simple model is introduced here not because it is

Table 1 Co-operative Speaker Rules

1 IF	you believe an assumption A to be false
THEN	don't use it
2 IF	you believe an assumption A is almost certainly true
THEN	leave A implicit and take no further action;

d.9 "*What courses will the undergraduates be taking this year*"

3 IF	you believe an assumption A is possibly false
AND	inconsequential to the purpose of the communication (i.e. it doesn't matter if it's false)
THEN	leave A implicit and take no further action;

d.10 "*Make sure you retract your aerial before entering the car wash*"

4 IF	you believe an assumption A is possibly false by virtue of not having some knowledge K
AND	you believe the hearer has knowledge K;
THEN	leave A implicit and if the hearer doesn't correct it, infer that A is true

d.11 "*List all the numbers that are both odd and even*"

(when you're not sure whether numbers can be both odd and even)

5 IF	you believe assumption A is possibly false by virtue of some knowledge K
THEN	make A explicit.

d.12 "*See you on the 3.45 tomorrow, if the buses are running*"

Table 1—The rules are intended to be interpreted top-down (i.e. If the antecedent is false then go on to the next rule down, else execute, go back to the top and wait for the next utterance). It is interesting to note that rule "5" above is appropriate whether or not the hearer has the knowledge which puts the assumption in doubt. Making the assumption explicit serves to avoid a misconception (if the hearer has the knowledge) or to inform the hearer (if the hearer doesn't have the knowledge). This is also true of hearer rules "2" and "3" below. Thus these rules are independent of the other participant's knowledge.

insightful (it isn't) but to illuminate the idea of mutual representations and facilitate the explanations to follow.

During natural conversation the collective knowledge of the discourse participants is drawn upon, to amend, confirm, remind and speculate about the assumptions upon which utterances are based. In this respect Grice's maxim of "Quality" may be violated by individual participants. However, its enforcement by the discourse participants allows this violation to be informative. The system's model of the user comprises: what the system believes the user believes about a domain and about the system itself; and what the system believes the user's goals are. One of the communication tasks of the dialogue is to maintain this knowledge source ensuring that what the user believes is accurate for the purposes of the dialogue.

Rules for the maintenance of mutual representations through the assumptions which underlie individual utterances are suggested in Tables 1 and 2.

To generate the corresponding rule set for the hearer it has been assumed that only assumptions which are believed to be false or possibly false will require action. These will be detected by virtue of: the hearer's knowledge; or explicit reference by the speaker respectively. Possible rules may be as follows:

Table 2 Co-operative hearer rules

1 IF you detect an assumption A which you believe to be false by virtue of some knowledge K
AND A is not made explicit by the speaker
AND A is inconsequential to the purpose of the communication (i.e. it doesn't matter that it's false)
THEN ignore it;

("*My car doesn't have an aerial to retract*" is not worth saying in reply to d.10)

2 IF you detect an assumption A which you believe to be false by virtue of some knowledge K
AND A is not made explicit by the speaker
THEN inform the speaker of this knowledge;

d.13 "*numbers cannot be both odd and even*" (in reply to d.11)

3 IF you detect an assumption A which you believe *may* be false by virtue of some knowledge K
AND A is not made explicit by the speaker
THEN inform the speaker of this knowledge;

d.14 "*Yes, I'll see you on the 3.45 if the buses are running*"
(in reply to "*See you on the 3.45*")

4 IF you detect an assumption A which you believe *may* be false by virtue of some knowledge K
AND A is made explicit by the speaker
THEN ignore it;

d.15 "*Yes*" (in reply to d.12)

5 IF you detect an assumption A which you believe may be false by virtue of explicit reference in the utterance
AND you do not have knowledge K implied by the utterance
THEN ask for this knowledge.

d.16 "*What makes you say 'if the buses are running'?*"
(in reply to d.12)

Table 2—Again rules are to be interpreted top-down. The subtle difference between "hearer" rules "2" and "3" is used to distinguish between knowledge that makes an assumption false and knowledge that makes an assumption uncertain. In rule "2" the reply is to inform or remind the speaker of the error. In rule "3" the reply serves to qualify the speaker's utterance.

The two major problems which have to be solved in order to build a system capable of checking the assumptions behind a user's utterances are identification and bounding. That is, how is it possible to identify the assumptions underlying an utterance, and limit the search to those assumptions which are relevant? Systems which have been built so far (McKeown, 1983; Kaplan, 1982; Mercer, 1984; Gal, 1985; Mays, 1980) limited the problem of identification to explicitly mentioned assumptions about the existence of entities and the relations between them (see below). Less explicit assumptions may be available only through the dialogue model (i.e. saying X at this point indicates the user believes Y).

In most situations many of the assumptions/beliefs underlying an utterance are not worthy of exhaustive checking by either participants due to the extremely low probability that they are false (e.g. the world will probably still exist tomorrow; a cheese shop sells cheese). Where such "safe" assumptions are so rarely incorrect it is more efficient to repair errors when they occur rather than check before hand.

To identify the relevant assumptions, from all the possible assumptions upon which an utterance could be based, requires bounding constraints. Without these the propositional content of the last speaker example (above) could have reflected an infinite number of possible worlds, any one of which could conspire to prevent the meeting on the 3.45 bus:

> d.17 "*See you on the 3.45 tomorrow, if the buses are running and I manage to get up and I don't break my leg and I don't get falsely accused of riding a bike with no lights....*"

Clearly, many of these scenarios are unlikely and so fall into the safe category of assumptions which need not be made explicit. Indeed, making any one of them explicit is to imply by the quantity maxim, the increased likelihood of their occurrence. That is, it implies that the speaker knows something which may put the truth of the assumption in doubt (e.g. the speaker intends to ride a bike with dim lights past the police station at midnight). This is an interesting co-operative response which relies on the conventions of discourse involved in the maintenance of mutual representations. More of this later (see 4.4). For now, what of the boundary conditions?

Boundary conditions limit the search for information which highlights the relevant assumptions underlying an utterance. This may define or be defined by, the context of the utterance. For every-day utterances like d.17 the context comprises a large slice of human experience. Fortunately it is possible to isolate tasks where informed and useful decisions can be made without recourse to such an extensive knowledge base. It is important to note here that a successful dialogue management system may depend less on the comprehensive nature of its contextual knowledge, and more on the discourse rules which allow its knowledge to be used more effectively. This is a position adopted, though unstated, by most of the work on dialogue management to date. To summarize it:

(1) boundary conditions can be limited by choice of task; and
(2) success does not depend on recreating the boundary conditions of human-human dialogue.

Research which has addressed the problem of identifying misconceptions (false assumptions) has generated a number of co-operative add-ons to existing systems. The boundary conditions (context) within which assumptions are checked, are provided in two ways: By using information originally in the system (Kaplan, 1982; Mercer, 1984; Gal, 1985); or by building small knowledge bases extra to the original system (Mays, 1980).

Kaplan's (1982) CO-OP checks for the existence of entities mentioned in natural language queries to a database. Existence in this case relates to the database's knowledge. The possibility of disparate user and system representations is triggered by the database query producing a null answer. Traditional database systems leave the user to solve the ambiguity of this response. Such an empty set could mean the question was inappropriate for the system's representation of the domain (i.e. the query could never be satisfied) or the query was appropriate but not satisfied on this occasion. In the former case there is clearly a disparity between the user's representation of the world and the system's. This could be because the user has a misconception about the world or a misconception about the way the system represents the world. Either way communication could be improved if the misconception was detected. For example:

d.18 "*How many students failed CS100 last year?*"

If no such course (CS100) is known to the database a null response will be produced (i.e. the database would reply "0"). This is likely to lead the questioner to believe that everyone passed unless the misconception is pointed out.

Once a null answer is generated Kaplan's co-operative system checks that all the entities mentioned in the query exist in the database. Should one or more turn out to be empty sets the null response is replaced by a co-operative one of the form: "*I don't know of X*". In the case of d.18 the reply would be "*I don't know of a CS100*". The proposed effect of such a response is to repair either the user's beliefs about the system or the user's beliefs about the world.

A system with a similar goal (Gal et al., 1985) uses the integrity constraints used normally to maintain consistency within the database. These are like type constrains which determine the values that can appear in the fields of the database. Using this information it is possible to check not only assumptions about the existence of entities but also the relations between them.

A further source of information against which to check assumptions is the lexical knowledge which may be available as part of the natural language facility. Whereas Kaplan's and Gal's systems check the "extentional" misconceptions (Kaplan), information regarding the use of particular lexical items has been used to correct inferred misconceptions (Mercer, 1984). Finin (1986) gives the following example:

d.19 "*Has John stopped taking CIS153?*".

The use of the word stopped implies having started. Mercer's system is able to add this to its check list and reply with:

d.20 "*No, because he hasn't started CIS153*".

Mays (1980) builds upon the information already present in the original system by providing a rich data model against which to check for misconceptions. The data model comprises three types of information: "Taxonomic"; "Selectional"; and "Mutual exclusion". Selectional constraints are the same as sortal constraints in natural language processing (e.g. "teach" requires a "teacher" and a "course"). Mutual exclusion supplies constraints are similar in effect to Gal's integrity constraints (e.g. Undergraduates can't teach). When combined with taxonomic information the system can check the existence of a broader range of user descriptions and that their relationships conform to selectional restrictions. Thus a reference to a woman teaching will only be perceived as a misconception if she turns out to be an undergraduate.

The above research has focused on communication with database information. The reduced functionality of such systems limit the misconceptions possible to those about reference (i.e. the existence of entities and their relations). If the functionality of the underlying system was extended to expert systems a new class of misconception is possible. These are misconceptions about the utility of plans (Kidd, 1984) and typically arise when clients propose possible solutions for expert evaluation. For example:

d.21 "*Can I get rid of the mould on my bathroom tiles with bleach?*"

To which a literal reply would be "*yes*". A more co-operative answer would be to point out that the result of this plan is only short term and that a more permanent solution is to cure the ventilation problem.

The users of expert systems are typically looking to solve problems beyond those of data retrieval. Such systems are expected to identify problems within a domain and provide useful information towards their solution. The success with which current systems perform these tasks has been shown to be limited (Kidd, 1984; Steels, 1984; Breuker and Wielinga, 1985; Davis, 1982). One of the consequences of this realization has been the suggestion that advice giving systems should allow the user a more active role in the problem solving process of the consultation (Kidd, 1984). Users should be allowed to make their own suggestions towards a solution as in d.21 rather than simply collecting data for the autocratic system.

An extended range of questions that have been proposed to increase user participation (Kidd, 1984) is discussed in more detail in section 4.5. For now, it is sufficient to note that allowing the user to input suggestions towards a solution, would require an evaluation of the user's plans (implicit and explicit). Thus, a new class of misconceptions would have to be dealt with by the system: those concerning the assumptions about the domain upon which the user's plans are based. The knowledge required to support the detection and correction of such disparities concern the prerequisites for, and consequences of, actions within the domain.

Another type of misconception common to current expert systems has been the focus of Martha Pollack's work (1985) which was briefly described in section 3. The misconception built into such systems, is that the user is asking the appropriate question given his/her requirements. The aim of Pollack's work is to develop advice giving systems which do not make "*The appropriate query assumption*". Three problems are identified, which provide the basis of this capability:

(1) When does the expert respond directly to the user's stated goal?
(2) What information constrains the expert's search for another goal?
(3) What sort of relations hold between the stated goal and the goal the expert addresses?

The work reported to date tackles the last of these problems. To enable the system to infer the goals behind user queries three relations between actions are described:

"Generates"—A generates B if the user can be said to be attempting B by attempting A
"Enables"—A enables B if doing A puts the user in a position to do B
"Is-alternative-to"—A is alternative to B if A achieves some but not all the effects of B

An interesting point to note about this work is that correction of the misconception is sometimes implicit. The expert may simply answer the inferred "appropriate" query.

The result of this work may eventually correct a long standing misconception held by (built into) computer systems (advice giver role) but the consequence will be the potential to correct a further class of user (advice seeker role) misconception regarding "the nature of their problem".

4.4 Preventing misconceptions

At the start of the previous section it was suggested that the underlying goal behind the detection of misconceptions was to foster mutual representations between the dialogue participants. It was also suggested that due to the inflexibility of computer representations it is currently necessary for such systems to take the dominant role. This didn't matter so much because of the discourse roles being emulated (i.e. consultation roles).

A similar story is true for co-operative responses which seek to avoid misconceptions. The misconceptions described above concerned assumptions already made by the speaker and used to generate the utterance, this type of misconception refers to assumptions made by the hearer on the basis of what the speaker said. It is the consequence of forward chaining behaviour where valid but false inferences are made on the basis of an utterance. Elsewhere this type of misconception has been called misconstrual (Joshi, 1982).

The necessity for this type of co-operative behaviour stems from the use of default reasoning by the dialogue participants. This arises from the interplay between Grice's maxims of Quality and Quantity.

The default reasoning that ensues concerns the belief that dialogue participants will adhere to these rules. Thus, hearer's will interpret utterances as reflecting the speakers beliefs and containing all information that is relevant. By default, if a speaker fails to mention something then he or she (the speaker) believes it is not relevant. Conversely, whatever is included is relevant (speaker rule "4" takes advantage of this to confirm or deny an uncertain assumption).

This type of default reasoning provides a window onto a speaker's beliefs which is essential for the correction of misconceptions described in the previous section. The window results from a combination of the speaker's utterance and the hearer's default reasoning. The problem for the speaker is that this window is only one-way. That is, the speaker does not have a similar window onto the beliefs of the hearer (until the hearer becomes a speaker). To predict the effect of an utterance on the hearer's beliefs the speaker must reason about the interaction between the utterance and the hearer's default reasoning. But what are the bounds of this reasoning? What determines "the current purpose of the exchange"?

Believing that the broad purpose of maintaining mutual beliefs was not covered by the original maxims, Joshi (1982) has suggested the following addition to the maxim of "Quality":

> "*If you the speaker intend to say anything which may imply for the hearer something you believe to be false, then provide further information to block it*".

Joshi's addition provides a statement of the underlying principle which triggers the generation of the class of co-operative response discussed in this section.

Having expanded the notion of the "current purpose of the exchange" it becomes necessary to identify the types of reasoning which contribute to the maintenance of beliefs. It is the combination of these with the general default reasoning described above that will determine the effect of the speaker's utterance on the hearer's beliefs. The speaker's intention (following Joshi's addition) should be to anticipate those valid reasoning strategies the hearer might employ which could lead to false assumptions (misconstrual).

The following is a discussion of default reasoning strategies which could lead to misconstrual.

4.4.1 Default reasoning about the rules of communication

This relates to a hearer replying to an utterance generated via the third 'speaker rule' described above. That is:

"*IF* *you believe an assumption A might be false by virtue of not having knowledge K*
AND *you believe the hearer has knowledge K*
THEN *leave it implicit and if the hearer doesn't correct it, infer that it's true*".

If the hearer believes A to be false then a reply which does not correct the misconception will only succeed in confirming it. Thus the corresponding hearer rule (rule 2) is required:

"*IF* *you detect an assumption A which you believe to be false by virtue of some knowledge K*
AND *A is not made explicit by the speaker*
THEN *inform the speaker of this knowledge.*"

Whether the hearer is *correcting* a misconception or *preventing* one in this case, depends upon the speaker's belief about the assumption in question. The hearer doesn't need to know what this is, as the action to be taken (inform speaker) and its effect (repair speaker's beliefs) are the same in both instances. For such cases it is therefore possible to prevent misconception as a consequence of invoking the mechanisms for correction.

An example dialogue might be as follows:

d.22 *P1.* "*List all players that are both professional and amateur*"
P2. "*Players cannot be both professional and amateur*"

Had P2 replied with "*There are none*" as a database would. P1 might have deduced that a previous assumption was true. This is the same problem addressed by Kaplan's system described in 4.3 and illustrated by d.18.

4.4.2 Defeault reasoning about the content of communication

The type of default reasoning referred to here concerns the structure we impose upon the world to ease our interaction with it. The use of such structure was first proposed by Bartlett (1932) and later adopted by Minsky (1975) and Schank and Abelson (1977). Representing default information about objects and events is a way of taking advantage of the repetitions within our experiences. Thus, avoiding the problems of perceiving each new experience as novel.

This use of default reasoning has obvious advantages for communication, allowing reference to objects, events and situations without having to specify all the details. It becomes possible to specify relevant detail as those things about the current situation which are different from the default situation. Thus, going some way towards answering the question of quantity posed by Grice's maxim.

For dialogue participants to make use of default representations they must anticipate the consequences of the default reasoning which supports them. Failure to do this may lead to the hearer adopting default values which may generally hold but which are incorrect for the current dialogue:

> d.23 "*Rico's serves excellent mexican food but you'll have to take your own wine because they don't have a licence*"

4.4.3 Default reasoning about the tools of communication

A third source of possible misconception is another class of default reasoning. In this case the default reasoning is about the tools of communication rather than the content (as above). Reasoning about the normal usage of these tools may stem from linguistic or social conventions. The use of indirect speech acts relies on this type of reasoning. For example:

> d.24 "*Are you going to be in the bathroom all morning?*"

This illustrates a social convention where the interrogative form is used as a request. The use of such indirect speech acts does not usually lead to false assumptions beyond the original interpretation of non-literal meaning. More oblique constructions may be more troublesome.

Conventions which are more likely to lead to deeper levels of misconstrual if violated, are those concerning specificity and definite reference. The former refers to the use or omission of words which imply the specific nature of information given and provides one example of this type of default reasoning:

> d.25 *P1.* "*The fastest way to London is by car*";
> *P2.* "*Today, the fastest way to London is by car*"

The absence of Today in the former utterance implies that the fastest way to London is always by car. It is interesting to note that this may not be true for speakers known to be uncooperative. User expectations have to be stimulated carefully.

A second convention of normal usage mentioned above was the use of definite reference. This is normally used to imply an unchanging, uniquely identified entity (e.g. "the name of the company", "the Archbishop of York", "the big number nine"). When it is the uniqueness of the definite referent which the hearer believes to be false, this is a case for *assumption correction* as no correct answer is possible. However, when the permanence of the referent is in doubt it is possible to give a correct answer and so the possibility of misconstrual must be blocked. For example, in replay to

> d.26 *P1.* "*What's the combination of the lab door?*"
> *P2.* "*999*"

P2's answer may be correct at the time of answering but incorrect a day later. To avoid the incorrect assumption resulting from default reasoning about the use of 'the' the response should include a block such as:

P2. "999, but it's changed every Monday".

A third example where false implication is likely to arise when normal usage is violated, occurs during database enquiry. Consider the following request:

d.27 *P1. "Which proteins with structure X have property Y?"*

After completing the search the system may return with the list below:

P2. "Methionine; Tryptophan; Tyrosine"

From this the user may conclude that three members of the set [X] have property Y. Whilst this is true it is misleading if the three proteins listed exhaust the set [X]. That is, the conventions of normal usage would naturally produce the following utterance:

P2. "All the proteins with structure X have property Y"

"*All but one . . .*" would be a similarly co-operative response (provided the exception is named). For database responses it may be necessary to include a list of the proteins satisfying the query.

4.5 Adaptive responses

Adaptive responses have for a long time been a major goal within the AI subfield of Intelligent Computer Aided Learning. However, the potential use of such interfaces extends far beyond this application to most interactive tasks which require the system to impart information to the user.

User modelling has been seen as the enabling technology from which adaptive responses will emerge. These models can be "Static" or "Dynamic". The latter are constructed on-line by the system during its interaction with the user. The system's responses are then tailored to the requirements of the user on the basis of the information collected in the user model. This is the type of modelling referred to in the previous discussion of misconceptions (4.3 and 4.4).

Static user modelling is a common feature of any system which requires user input. It is the generic model of the user built into the design of the system's interface. It constrains the nature of the user's communication with the system (i.e. what the user is allowed to ask of the system and how and when the user is allowed to ask it). The extent to which an interface based on such a model can be said to be adaptive depends upon the extent of these constraints. The introduction of AI technologies may facilitate the building of systems with much greater freedom to adapt within the constraints of a static user model.

Traditional expert systems demonstrate this type of co-operative behaviour in the way they adapt their problem solving behaviour during consultation to the responses of the user. For such systems it is not the user that the system is adapting to but the situation described by the user. Decisions are based on preconceived notions of generic consultations and generic users. Mycin adapts to the 'current situation" by invoking heuristics to decide which questions to ask based on the effectiveness of competing questions (Shortliffe, 1976). Effectiveness is determined by the ease with which the user can supply the data and the probability of the answer leading to a major conclusion.

At the other end of the spectrum from systems which adapt to the current situation, are systems which adapt to the user. Such systems generate dynamic user models during consultation. The interface is then adapted based on the information within this user model. The Adaptive Intelligent Dialogue (AID) project sponsored under the UK Alvey initiative, is an example of work in this field. At its simplest the interface adapts to user words in the same way as systems described under Terminology clarification (section 4.2). In the case of the AID project an adaptive interface to an electronic mail system, adapts to the user's experience with other mail systems. Communication is improved by customizing the system's interpretation and explanation capabilities to take advantage of this.

Between the two extremes described above lies an approach which avoids dynamic user modelling whilst promising greater adaptability to user requirements. Communication is based on the former generic or static models of the user. Adaptability is provided by extending the types of interaction the user is allowed to engage in. The result is a redefinition of the notions of generic consultation and generic user. The system is designed to support a greater range of user input and hence a greater range of co-operative response.

Analysis of naturally occurring consultations has suggested some of the questions asked of human experts but not supported by current technology (Kidd, 1984):

1. *"What is the fault?"*
2. *"What is the remedy?"*
3. *"Why is X important?"*
4. *"Why did the fault occur?"*
5. *"Why did the remedy work?"*
6. *"Why didn't the remedy work?"*
7. *"Will remedy X work?"*
8. *"How can I test for X?"*

Early systems were built to answer only the first three of these questions with the result that user's were allowed very limited access to the knowledge being used to address their problem. Clearly, if the other five were to be supported the user would enjoy greater access to this knowledge. In this way systems would become more flexible to the demands of individual users.

Supporting an extended range of user enquiry adds to the complexity of the required dialogue management system. For early systems the first two questions were implicit in the design of the system. These two questions were asked indirectly as soon as the system was "switched on". The third question was the only question the user was allowed to ask directly. To support a range of questions the dialogue management system must contain: a theory which relates the type of question with the information being sought; and theory about how to convey that information to the user.

A step towards the former theory is the following classification of the above surface level questions into their more generic form. The numbers in square brackets refer to the corresponding questions above. The labels in the round brackets are taken from Lehnert's classification of question types (1982).

[*2, 8*]	*A.*	*"What will achieve Y?"*	("Enablement")
[*1, 4*]	*B.*	*"What did achieve Y?"*	("Causal Antecedent")
[*3, 5*]	*C.*	*"Why will/did X achieve Y?"*	("*Causal Relation*")**
[*6*]	*D.*	*"Why didn't/won't X achieve Y?"*	("Expectational")
[*7*]	*E.*	*"Will X achieve Y?"*	("Verification")

[** *This category is suggested by the author*]

So far this discussion has focused on adaptive responses to questions posed by users. I should now like to turn to the potential for extending the range of user responses to the system's request for data. Current systems ask the user for data by presenting a set of optional responses. One of these might be "Don't know" which if selected causes the system to proceed without the data. A further step towards co-operativity is to furnish the user with information relevant to the question. This covers the situations when the user has forgotten the answer or when the user doesn't know how to get the data. This is usually achieved by allowing the user to ask "HOW?" and then providing procedural knowledge about how to obtain the data. If this procedure is a lengthy process then the user may want to ask "WHY?" in order to find out whether an accurate answer to the question is critical. In current systems this usually results in a rule trace. The onus is on the user to decipher this representation of the system's reasoning and make the decision. An alternative solution would be to let the system decide whether the data is important and provide extra prompts if it is. For example:

d.28 *P1. "Have you ever paid royalties to divisions outside Europe?"*

P2. "I can't remember"
P1. "Have you ever filled in a yellow royalties form?"

In some situations it may be useful to allow the user to offer information (uncued) which may help the system to deduce the answer:

d.29 *P1. "What's your location code?"*
P2. "I'm a member of the Leo project"

This luxury afforded by a natural language medium may be approximated in other media by presenting a cue menu of facts from which the answer could be deduced.

4.6 Co-operative search

Co-operative search is a form of Task Driven Co-operativity in which the system takes the initiative to provide useful information extra to the immediate response requirements. That is, the system provides supplements to the answers to direct questions, which are intended to help the user in the search for information.

The key primitives of this type of co-operative behaviour are the constraints imposed on the search. The search could be for data from a database or for expertise from an expert system. The constraints in either case describe the object of the search. The work on databases (Motro, 1986) builds on Kaplan's (1982) work (cf. section 3.3). Kaplan's COOP you may remember checked the existence of entities in the database. The added sophistication of co-operative search techniques is to relax the constraints supplied by the user if the original query returns a "null" response. This requires a data model of the information contained in the database. Motro proposes what he calls a "loose structure data model". This comprises a set of facts and relations represented as prolog clauses.

If as a result of relaxation a non-null answer is found it is presented as a partial solution to the query. If after successive relaxations a null response is still returned, the system presents this answer with the most relaxed form of the query. The latter response is aimed at correcting what is perceived to be the most general misconception reflected in the query. For example, if the query was:

d.30 *P1. "Which red Jaguars were registered before 1984"*,

the system might make following responses:

P2. "I don't know of any, but I do know of other Jaguars registered before 1984"
P2. "I don't know of any vehicles registered before 1984"

The first response offers the partial solution of non-red Jaguars, and the second attempts to correct the misconception that there are any vehicles in the database, registered before 1984.

Motro describes how relaxation might be achieved by means of "generalization" and "elimination" but fails to discuss how these processes might be controlled in a principled way. Elimination is as the name suggests relaxation of the total query specification by eliminating one of the constraints. In the above example a partial solution was obtained by eliminating the colour constraint. Generalization can take two forms depending on the existence of taxonomies within the data model. If taxonomic hierarchies exist then constraints can be relaxed by iteratively ascending them. Again, in the example above Jaguars was relaxed to vehicles before the second response was produced.

The second form of generalization, if taxonomies don't exist in the data model, requires the construction of neighbourhoods of similar entities. Neighbourhoods are constructed by selecting the population of entities with the maximum number of overlapping arguments with the target entity. In the absence of a vehicle hierarchy a neighbourhood constructed around Jaguars may have included Rolls Royces and milk floats. It could equally have included microlights and speed boats if the common arguments were size and seating capacity.

Clearly, without the underlying principles to control constraint relaxation the system is not guaranteed to come up with a co-operative response. In generating examples of co-operative relaxation it is easy to think of useful relaxations because we have a context in mind. To develop systems which can do the same, it will be necessary to identify the context and how to use it.

Work which is not presented as constraint relaxation but which has a similar feel to the previous approach, is that on "Scalar Implicatures" (Hirschberg, 1986). In the situations described, co-operative responses are generated by reference to known partial orderings. Consider the following example (taken from Hirschberg, 1986):

d.31 *P1. "Has the Acme shipment come in?"*
P2. "No"
P1. "Have they sent it?"
P2. "No"
P1. "Was it ordered?"
P2. "No"
P1. "Was the requisition filed?"
P2. "Yes"

Clearly P2 would be open to an accusation of stone-walling and is certainly not conforming to Grice's maxim of quantity. The co-operative response to the original question would have been:

P2. "The requisition has been filed"

Scalar implicature relies on the partial ordering of the preconditions. Stating that one of the preconditions has been met implies that all the previous conditions have also been satisfied and all the subsequent ones have not. The result of these implications is to extend the co-operative responses possible beyond the state referenced by the utterance. That is the original query in the example above could have been:

P1. "Has the Acme shipment been ordered?"

to which a possible co-operative response could be:

P2. "The Acme shipment has come in"

The reason for including this work here is its similarity to constraint relaxation described above. The partial ordering provides a predetermined relaxation path. The more relaxed form of "the shipment in" is "the shipment sent" etc.

An interesting consequence of viewing this work as constraint relaxation is the implication for co-operative responses based on constraint focusing. It must be noted though that the necessity for control information is as critical for focusing as it is for relaxation.

An alternative to constraint relaxation as a means of directing the search is to give the user information about the search space. This may include domain information about locally active constraints, alternative strategies, or the consequences of permissible actions. This approach touches on the work of Pollack (1985) described in section 4.3 in that the sort of information required is the same as that required to support the concepts of "generation" and "enablement".

The type of co-operative response produced using this information is best characterized by the work of Kidd (1985) which identifies common co-operative supplements to direct "yes/no" answers:

d.32 "*Will X achieve Y*" (cf. section 4.5)

Possible replies:
1. "*No, not unless Z*"
2. "*Yes, as long as Z*"
3. "*No, but W will*"
4. "*No, but X will achieve V*"
5. "*Yes, but so will W*"
6. "*Yes, but side effect U*"
7. "*Yes/No because*"

Unlike constraint relaxation where the system takes the initiative, the strategy here is to provide the user with a mapping of the query and its underlying constraints onto the active part of the search space. The user is then left with the initiative to make decisions based on this information. It is here that the approach diverges from Pollack's, in which the system takes the initiative in generating an appropriate query.

To support such a capability the dialogue management system would have to address the problems of: 1. deciding when each of the possible answers would be appropriate; and 2. how to limit the instantiations of "U", "V", "W", "Z" in a principled way. The following analysis is a possible step towards addressing these:

Table 3 Conditions required for co-operative responses to d.33

Response	*When*	*Instantiation*
[1, 2]	"Z" is a precondition	"Z" is essential and not satisfied
[3]	"W" is an alternative	"W" is possible in the current state
[4]	"V" is a consequence	"V" is a user goal or will enable "Y" *
[5]	"W" is an alternative	"W" is more easily achieved from the current state
[6]	"U" is a consequence	"U" is undesirable or will constrain future decisions
[7]	"Z" is an elaboration (see 3.6 below)	"Z" contradicts a false assumption held by the speaker or is based on a tutoring principle

[* *enablement is a term used by Pollack (cf. 3.3). Briefly, A enables B, means A puts you in a position to achieve B.*]

Though the above analysis is still vague, it does begin to illuminate the sort of information required to support this type of co-operative response (e.g. the currently active constraints, the user's goals, the consequences of actions on the current and future states). It may turn out to be impractical to capture all the information required to reproduce many of these responses. However, there is no reason why we shouldn't exploit the trade-offs to provide those responses that are computationally feasible.

4.7 A note on explanation

As a central aspect of co-operative dialogue the issues normally subsumed under this heading have been addressed implicitly in the work reported in this review. The narrow view of explanation adopted by 1st generation expert systems focused on one generic question: "*Why will/did X achieve Y?*" (cf. 4.5). Dialogue management in such systems is therefore straightforward. The propositional (or domain information) content of the input from the user, is restricted to: the data requested in the form specified; or a query about the system's reasoning. An explanation in response to the latter comprises a rule trace whatever the position in the dialogue.

The limitations of this narrow focus and the possibilities for extending the range of explanation capabilities have been discussed. Supporting these capabilities requires a dialogue management system of increased complexity. The problems of analysis now include: how to recognize the sort of explanation being requested; and how to communicate the information in a co-operative way. The tasks can be summarized thus:

(1) Analysis
(Identification of question type from the utterance);
(2) Co-operative communication
(Identifying the relevant tools to co-operatively communicate the required information)

The wording reflects the fact that questions are the major tool of communication with advice giving systems. To extend the issues above to dialogue in general, requires only the swapping of the word "question" in 1 for the word "communication".

It is important to note that the above tasks do not include retrieval of the domain information which is to make up the propositional content of the answer. The general role of dialogue management is to act as a communication channel between the system and the user. Thus, the emphasis is on the detection of the illocutionary force, adherence to the conventions of co-operative communication and identification of the type of information to be retrieved.

Retrieval of the relevant domain information is being addressed within other subfields of AI concerned with: ICAI (e.g. Clancey, 1986; Sleeman and Brown, 1982); and "Deep knowledge"/"Causal reasoning" (e.g. Chandrasekaran, 1985; Davis, 1984; de Kleer and Brown, 1984; Kuipers, 1984). The issues within each of these subfields are beyond the scope of this review. The knowledge structure generated by the dialogue management system (e.g. user beliefs or discourse model) may be useful for identifying the domain information required. However, the principles by which this identification is achieved and the information content of the answer derived, are the tutorial and reasoning principles of the afore mentioned subfields.

5 Conclusions

The emergence of co-operative dialogue management research is in part, due to the potential flexibility of communication offered by natural language technology. The maturing of this latter technology over the last ten years has prompted a number of theses devoted to the principles of co-operative communication. To date, this work has been spread very thinly over a number of co-operative dialogue phenomena. Most of these have been loosely linked to general theories of discourse.

This review has drawn together some of the work on dialogue management to acknowledge the progress that has been made and to highlight the research that is still to be done. Six classes of co-operative dialogue were presented in section 4. These were cast in the light of two underlying goals (BMC and TDC) in order to illustrate the knowledge required to support their principled use.

Paraphrasing was described as a useful tool for BMC though there is little evidence of its use in expert systems. It is exploited in some natural language front ends to database systems. However, in these systems it is used blindly due to the lack of control information which determines when it should be used.

Clarifying terminology was shown to be a more fundamental problem than simple synonym generation and have implications for the selection of domains for knowledge based applications. The more tractable applications are those which do not require the negotiation of fundamental levels of abstraction.

Two sets of production rules were suggested as a basis for BMC through the assumptions which underlie individual utterances. These rules were used to illustrate how such assumptions are managed to correct and avoid misconceptions. To correct misconceptions in this way requires the solution to the problems of bounding and identification. Current systems were shown to deal with these problems by limiting the assumptions considered, to those concerning the existence of objects and relations in the database or knowledge base.

Joshi's addition to Grice's maxim of quality was shown to be the basis for co-operative responses intended to prevent misconceptions. It was suggested that such responses rely on three types of default reasoning about: the tools; rules; and content of communication. The knowledge required for this reasoning was shown to include: the way we represent the world; the inferences behind specific sentence constructions; and social convention.

User modelling was introduced as the enabling technology for dialogue which adapts to the communication needs and capabilities of a user. Two approaches were described: dynamic and static user modelling. A middle ground was suggested. This approach considered improving the communication channel in two ways: by extending the range of questions a user can ask a system and the range of responses the user can make in response to the system's requests. To support these capabilities a system would have to include a theory which related question type to the information being sought and a means of inferring the data required from indirect answers.

Two approaches to co-operative search were presented. The first described ways of achieving TDC in data retrieval applications and the second, expert system applications. In the case of database retrieval the main tool was constraint relaxation. A system was described which generalized database queries in response to null answers. A problem to be solved was shown to be that of control. That is, how to generalize queries in a principled way. For expert systems co-operative search is achieved by providing the user with information about the search space. This includes the preconditions and effects of proposed actions and their relation to task constraints. Progress will therefore, draw heavily on the techniques developed within the planning subfield of AI.

The utility of the above classes of co-operative dialogue for software systems is still to be demonstrated. The systems described in this review are experimental. For the transference of this technology to commercial systems this work must be drawn together into a design methodology which describes: when different types of co-operative behaviour are appropriate; how they and the principles which govern their use can be implemented in computer software; and for what applications it is useful to do so. So far implementations have amounted to surface simulations of isolated phenomena. Ultimate success will require a computational theory of co-operative dialogue. The data is still to be collected towards this theory. The analyses presented in this review have pointed to the type of data that is required.

Acknowledgements

I should like to thank Steve Whittaker, John Fox, Elaine Rich and Marion Owen for their invaluable comments on earlier drafts of this paper.

6 References

Allen, J. and Perrault, P.C. (1980). "Analysing intention in utterances". *Artificial Intelligence* **15**, pp. 143–178.

Appelt, D.E. (1985). *Planning English Sentences.* Cambridge University Press.

Austin, J.L. (1962). *How to Do Things with Words.* Oxford University Press.

Bartlett, F.C. (1932). *Remembering: A Study in Experimental and Social Psychology.* Cambridge University Press.

Boguraev, B.K. and Sparck Jones (1984). "A natural language front end to databases with evaluative feedback." In G. Gardarin and E. Gelenbe (Eds), *New Applications of Data bases.* Academic Press, pp. 159–182.

Brueker, B.J. and Wielinga, J.A. (1986). "The use of models in the interpretation of verbal behaviour," in A.L. Kidd (Ed.) *Knowledge Elicitation for Expert Systems: A Practical Handbook.* Plenum Press.

Chandrasekaran, B. and Milne, R. (1985). "*Reasoning about structure, behaviour and function.*" *SIGART Newsletter 93.*

Clancey, W.J. (1986). "Qualitative Student Models." *Stanford KSL working paper KSL-86-15.*

Clark, H. and Carlson, T. (1982). "*Hearers and speech acts.*" *Language* **58**(2).

Clark, H. and Marshall, C. (1981). "Definite reference and mutual knowledge," in A.K. Joshi, I. Sag, and B.L. Webber (Eds). *Linguistic Structure and Discourse Setting.* Cambridge University Press.

Cohen, P.R. (1978). *On Knowing What to Say: Planning Speech Acts.* PhD Thesis, Toronto.

Cohen, P.R. and Levesque, H.J. (1987). "Persistence, Intention and Commitment." *SRI technical note 415, Feb. 19th.*

Coombs, M. and Alty, J. (1984). "Expert systems: an alternative paradigm," *Int. Jnl. Man Machine Studies* **20**, pp. 21–23.

Davis, R. (1982). "Expert systems: Where are we and where do we go from here"? *A.I. Magazine* **3**, p. 22.

Davis, R. (1984). "Diagnostic reasoning based on structure and behaviour." *Artificial Intelligence* **24**(1–3).

de Kleer, J. and Brown, J. (1984). "A qualitative physics based on confluences." *Artificial Intelligence* **24**(1–3), pp. 205–280.

Dijkstra, E.W. (1976). *A Discipline of Programming.* Englewood Cliffs, N.J., Prentice Hall.

Fawcett, R. (1986). "Systemic Functional Grammar." *Seminar given at HPLabs BRC 12/12/85.*

Finin, T.W., Joshi, A.K. and Webber, B.L. (1986). "Natural language interactions with artificial experts." *Proc. IEEE* 74(7), pp. 921–938.

Gal, A. and Minker, J. (1985). "A natural language database interface that provides co-operative answers." *Proc. 2nd IEEE Ann. Conf. A.I. Applications.*

Grice, H.P. (1975). "Logic and conversation," in P. Cole and J. Morgan (Eds), *Syntax and Semantics.* Academic Press, pp. 41–58.

Grosz, B. (1977). "The representation and use of focus in dialogue understanding." PhD Thesis, Univ. Cal. Berkeley.

Halliday, M.A.K. and Hasan, R. (1976). *Cohesion in English.* Longman: London.

Hasan, R. (1971). "Text in the systemic-functional model," in W. Dressler (Ed.). *Current Trends in Textlinguistics*, De Gruyter, Berlin.

Hendrix, G., Sacerdoti, E., Sagalowicz, D. and Slocum, J. (1978). "Developing a natural language interface to complex data." *ACM Trans. Database Sys.* **3**(2), pp. 105–147.

Hess, M. (1985). "About the role of control information in natural language question answering systems." *Natural Language Understanding and Logic Programming.* Elsevier Science Publishers (North Holland).

Hirschberg, J. (1986). "Anticipating false implicatures: co-operative responses in question-answer systems," in L. Kerschberg (Ed.), *Expert Database Systems.* Benjamin Cummings Publ. Co. Inc.

Hobbs, J. (1976). A *Computational Approach to Discourse Analysis.* City Univ. New York.

Joshi, A.K. (1982). "The role of mutual beliefs in question answering systems." In N. Smith (Ed.), *Mutual Belief.* Academic Press.

Kaplan, S.J. (1982). "Co-operative responses from a portable natural language database query system," in M. Brady (Ed.), *Computational Models of Discourse*, MIT Press.

Kidd, A.L. (1984). "Dialogue and explanation in expert systems"—Invited paper for Joint MMI/IKBS Alvey Workshop, Cambridge (Nov.).

Kidd, A.L. (1985). "What do users ask? Some thoughts on diagnostic advice," in M. Merry (Ed.), *Expert Systems '85*, Cambridge University Press.

Kidd, A.L. (1986). "Leo: A research programme in co-operative knowledge based system design." *Tech. Report HPL-BRC-TR-86-015.*

Kidd, A.L. and Sharpe, W.P. (1986). "Goals for expert system research: An analysis of tasks and domains." *Tech Report HPL-BRC-TM-86-041.*

Kuipers, B. (1984). "Commonsense reasoning about causality: Deriving behaviour from structure." *Artificial Intelligence* **24** (1–3), pp. 169–204.

Lehnert, W. (1981). "A computational theory of human question answering," In A.K. Joshi, B.L. Webber and I. Sag, *Elements of Discourse Understanding*, Cambridge University Press.

Levin, J.A. and Moore, J.A. (1977). "Dialogue-Games: Metacommunication structures for natural language interaction." *Cogn. Sci.* **4**, pp. 395–421.

Linde, C. (1980). "The organisation of discourse," in T. Shopen and J.M. Williams, *Style and Variables in English.* Winthrop Publ. Inc.

McKeown, K. (1983). "Paraphrasing using given and new information in a question-answering system." *Am. Jnl. Comp. Ling.* **9**(1), pp. 1–11.

McKeown, K. (1985). "Discourse strategies for generating natural language texts." *Artificial Intelligence* **27**, pp. 1–41.

Mays, E. (1980). "Failures in natural language systems: Application to database query systems." *Proc. 1st Nat. Conf. A.I. (AAAI).* Stanford (Aug.).

Mercer, R. and Rosenberg, R. (1984). "Generating corrective answers by computing presuppositions of answers, not questions." *Proc. Can. Soc. Comp. Stud.*, Univ. Western Ontario (Apr.).

Minsky, M. (1975). "A framework for representing knowledge," in P.H. Winston (Ed.). *The Psychology of Computer Vision*, MIT Press.

Miller, P. (1984). "A Critiquing Approach to Expert Computer Advice: Attending." *Pitman Advanced Publ. Prog.*

Moore, R.C. (1980). "Reasoning about knowledge and action." *SRI International A. I. Center Tech. Note 187.*

Motro, A. (1986a). "Supporting goal queries in relational databases." *Proc 1st Int. Conf. Expert Database Systems*, pp. 85–95.

Motro, A. (1986b). "A query generalisation: A method for interpreting null answers." In L. Kerschberg, *Expert Database Systems* (Ed.), Benjamin Cummings Publ. Co. Inc.

Muekstein, E.M. (1983). "Q-Trans: Query translation into English." *8th I.J.C.A.I.*, Karlsruhe (Aug.).

Petrie-Brown, A. (1986). *Dialogue games and context spaces.* 1st Workshop on Co-operative Dialogue Management, Bristol (Oct.).

Pollack, M. (1985). "Information sought and information provided." *Proc. C.H.I. '85*, A.C.M., San Francisco (Apr.), pp. 155–160.

Power, R. (1979). "The organisation of purposeful dialogues." *Linguistics* 17, pp. 107–151.

Rawlings, C.J., Taylor, W.R., Nyakairu, J., Fox, J. and Sternberg, M.J.E. (1985). "Reasoning about protein topology using the logic programming language Prolog." *Jnl. Molecular Graphics* **3**(4), pp. 151–157.

Reichman, R. (1985). *Getting Computers to Talk Like You and Me.* MIT Press.

Rosenchein, J.S. (1981). "Abstract theories of discourse and the formal specification of programs that converse." In A.K. Joshi, B.L. Webber and I. Sag. *Elements of Discourse Understanding*, Cambridge University Press.

Rummelhart, D.E. (1975). "Notes on a schema for stories," in D. Bobrow and A. Collins (Eds), *Representation and Understanding*, Academic Press.

Sacks, H., Shegloff, E. and Jefferson, G. (1974). "A simplest systematics for the organisation of turntaking for conversation." *Language* **50**(4), pp. 696–735.

Schank, R. and Abelson, R.P. (1977). *Scripts Plans Goals and Understanding*. Hillside N.J. Lawrence Earlbaum Associates.

Searle, J.R. (1969). *Speech Acts*, Cambridge University Press.

Searle, J.R. (1971). What is a speech act? in J.R. Searle (Ed.), *The Philosophy of Language*, Oxford University Press.

Shortliffe, E. (1976). *Computer-based Medical Consultations: MYCIN*. New York, Elsevier.

Sidner, C. (1979). "Towards a computational theory of definite anaphora comprehension in English discourse." *PhD Thesis, MIT.*

Sleeman, D. and Brown, J.S. (1982). (Eds.). *Intelligent Tutoring Systems*. Academic Press, New York.

Steels, L. (1984). *Building second generation expert systems in 1985*, One day tutorial, Expert Systems '84, BCS/SGES Warwick (Dec.).

Weiner, J.L. (1980). "BLAH, A system which explains its reasoning." *Artificial Intelligence* **15**, pp. 19–48.

Winograd, T. and Flores, F. (1986). *Understanding Computers and Cognition: A New Foundation for Design*. Ablex Publ. Inc.

Young, R.M. (1984). *Human interface of expert systems*. Invited talk at the 1st Expert Systems Theme Workshop (5th–6th March).