INFLUENCE OF THE PROBLEM CONTEXT*

Aravind K. Joshi

Department of Computer and Information Science
Room 268 Moore School
University of Pennsylvania
Philadelphia, PA 19104

My comments are organized within the framework suggested by the Panel Chair, Barbara Grosz, which I find very appropriate. All of my comments pertain to the various issues raised by her; however, wherever possible I will discuss these issues more in the context of the "information seeking" interaction and the data base domain.

The primary question is how the purpose of the interaction or "the problem context" affects what is said and how it is interpreted. The two separate aspects of this question that must be considered are the <u>function</u> and the <u>domain</u> of the discourse.

1. Types of interactions (functions):

1.1 We are concerned here about a computer system participating in a restricted kind of dialogue with a person. A partial classification of some existing interactive systems, as suggested by Grosz, is as follows. I have renamed the third type in a somewhat more general fashion.

	Participant P1	Participant P2
	(Computer system)	(Person)
Type A	Expert	Apprentice
Type B	Tutor	Student
Type C	Information provider (some sort of large and complex data base or knowledge base)	Information seeker

Each type subsumes a variety of subtypes. For example, in type C, subtypes arise depending on the kind of information available and the type of the user. (More on this later when we discuss the interaction of constraints on function and domain).

1.2 It should be noted also that these different types are not really completely independent; information seeking (Type C) is often done by the apprentice (Type A) and student (Type B), and some of the explaining done by tutors (Type B) is also involved in the Type C interaction, for example, when P1 is trying to explain to P2 the structure of the data base.

1.3 The roles of the two participants are also not fixed completely. In the type C interaction, sometimes P2 partly plays the role of an expert (or at least appears to do so) believing that his/her expert advice may help the system answer the question more 'easily' or 'efficiently'. For example[1], in a pollution data base P1 may ask: <u>Has company A dumped any wastes last week?</u> and follow up with advice: <u>Try arsenic first.</u> In the expert-apprentice interaction, the expert's advice is assumed to be useful by the apprentice. In the data base domain it is not clear whether the 'expert' advice provided by the user is always useful. It does however provide information about the user which can be helpful in presenting the

response in an appropriate manner; for example, if arsenic indeed was one of the wastes dumped, then, perhaps, it should be listed first.

1.4 The interactions of the type we are concerned about here are all meant to aid a person in some fashion. Hence, a general characterization of all these types is a <u>helping</u> function. However, it is useful to distinguish the types depending on whether an <u>information seeking</u> or <u>information sharing</u> interaction is involved. Type C interaction is primarily information seeking, although some sharing interaction is involved also. This is so because information sharing facilitates information seeking, for example[2], when P1 explains the structure of the data base to P2, so that P2 can engage in information seeking more effectively. Type A and B are more information sharing than information seeking interactions.

1.5 Another useful distinction is that type C interaction has more of a <u>service</u> function than types A and B which have more of a <u>training</u> function. Training involves more of information sharing, while service involves more of providing information requested by the user.

2. Information about the user:

2.1 By user we usually mean user type and not a specific user. User information is essential in determining expectations on the part of the user and the needs of the user. Within each type of interaction there can be many user types and the same information may be needed by these different types of users for different reasons. For example, in type C interaction, preregistration information about a course scheduled for the forthcoming term may be of interest to an instructor because he/she wants to find out how popular his/her course is. On the other hand, the same data is useful to the registrar for deciding on a suitable room assignment. The data base system will often provide different views of the same data to different user types.

2.2 In general, knowledge about the user is necessary, at least in the type C interaction in order to decide

(i)   how to present the requested information,

(ii) what additional information, beyond that explicitly requested, might be usefully presented (this aspect is not independent of (i) above),

(iii) what kind of responses the system should provide when the user's misconceptions about the domain

(i.e., both the structure and content of the data base, in short, what can be talked about) are detected.

(More about this in Section 5).

3. Conversational style:

3.1 In the type C interaction, the user utterances (more precisely, user's typewritten input) are a series of questions separated by the system's responses. By and large, the system responds to the current question. However, knowledge about the preceding interaction i.e., discourse context (besides, of course, the information about the user) is essential for tracking the "topic" and thereby determining the "focus" in the current question. This is especially important for determining how to present the answer as well as how to provide appropriate responses, when user's misconceptions are detected.

Type A and B interactions perhaps involve a much more structured dialogue where the structure has its scope over much wider stretches of discourse as compared to the dialogues in the type C interactions, which appear to be less structured.

3.2 The type of interaction involved certainly affects the conversational style; however, little is known about conversational style in interactive man/machine communication. Folklore has it that users adapt very rapidly to the system's capabilities. It might be useful to compare this situation to that of a person talking to a foreigner. It has been claimed that natives talking to foreigners deliberately change their conversational style[3] (for example, slowing down their speech, using single words, repeating certain words, and even occasionally adopting some of the foreigner's style, etc.). It may be that users treat the computer system as an expert with respect to the knowledge of the domain but lacking in some communicative skills, much like a native talking to a foreigner.

Perhaps it is misleading to treat man/machine interactive discourse as just (hopefully better and better) approximations to human conversational interactions. No matter how sophisticated these systems become, they will at the very least lack the face to face interaction. It may be that there are certain aspects of these interactions that are peculiar to this modality and will always remain so. We seem to know so little about these aspects. These remarks, perhaps, belong more to the scope of the panel on social context than to the scope of this panel on the problem context.

4. Relation of expectations and functions:

4.1 In the information seeking interaction, usually, the imperative force of the user's questions is to have the system bring it about that the user comes to know whatever he/she is asking for. Thus in asking the question Who is registered in CIS 591? the user is interested in knowing who is registered in CIS 591. The user is normally not interested in how the system got the answer. In the type A and B interactions the imperative force of a question from the user (apprentice or student) can either be the same as before or it can have the imperative force of making the system show the user how the answer was obtained by the system.

4.2 In the data base domain, although, primarily the user is interested in what the answer is and not in how it wa obtained, this need not be the case always. Somet..es the user would like to have the answer accompanied by how it was obtained, the 'access paths' through the data base, for example.

4.3 Even when only the what answer is expected, often the presentation of the answer has to be accompanied by some 'supportive' information to make the response useful to the user[4]. For example, along with the student name, his/her department or whether he/she is a graduate or undergraduate student would have to be stated. If telephone numbers of students are requested then along with the telephone numbers, the corresponding names of students will have to be provided.

5. Shared knowledge and beliefs:

5.1 The shared beliefs and goals are embodied in the system's knowledge of the user (i.e., a user model). It is important to assume that not only the system has the knowledge of the user but that the user assumes that the system has this knowledge. This is very necessary to generate appropriate cooperative responses and their being correctly understood as such by the user. In ordinary conversations this type of knowledge could lead to an infinite regress and hence, the need to require the shared knowledge to be 'mutual knowledge'. However, in the current data base systems (and even in the expert-apprentice and tutor-student interactions) I am not aware of situations that truly lead to some of the well known problems about 'mutual knowledge'.

5.2 As regards the knowledge of the data base itself (both structure and content), the system, of course, has this knowledge. However, it is not necessary that the user has this knowledge. In fact very often the user's view of the data base will be different from the system's view. For large and complex data bases this is more likely to be the case. The system has to be able to discern the user's view and present the answers, keeping in mind the user's view, while insuring that his/her view is consistent with the system's view.

5.3 When the system recognizes some disparity between its view and the user's view, it has to provide appropriate corrective responses. Users' misconceptions could be either extensional (i.e., about the content of the data base) or intensional (i.e., about the structure of the data base)[4]. Note that the extensional/intensional distinction is from the point of view of the system. The user may not have made the distinction in that way. Some simple examples of corrective responses are as follows. A user's question: Who took CIS 591 in Fall 1979? presumes that CIS 591 was offered in Fall 1979. If this was not the case then a response None by the system would be misleading; rather the response should be that CIS 591 was not offered in Fall 1979. This is an instance of an extensional failure. An example of intensional failure is as follows. A user's question: How many undergraduates taught courses in Fall 1979? presumes (among other things) that undergraduates do teach courses. This is an intensional presumption. If it is false then once again an answer None would be misleading; rather the response should be that undergraduates are not perm .ted to teach courses, faculty members teach courses, and graduate students teach courses. The exact nature of this response depends on the structure of the data base.

6. Complexity of the domain:

6.1 In each type of interaction the complexity of the interaction depends both on the nature of the interaction (i.e., function) as well as the domain. In many ways the complexity of the interaction ultimately seems to depend on the complexity of the domain. If the task itself is not very complex (for example, boiling water for tea instead of assembling a pump) the task oriented expert-apprentice interaction cannot be very complex. On the other hand data base interaction which appear to be simple at first sight become in-

creasingly complex when we begin to consider (i) dyna-
mic data bases (i.e., they can be updated) and the
associated problems of monitoring events (ii) data
bases with multiple views of data, (iii) questions
whose answers require the system to make fairly deep
inferences and involve computations on the data base
i.e., the answers are not obtained by a straightforward
retrieval process, etc.

NOTES:

1. As in the PLIDIS system described by Genevieve
   Berry-Rogghe.

2. As in Kathy McKeown's current work on generating
   descriptions and explanations about data base
   structure.

3. For example, by R. Rammurti in her talk on
   'Strategies involved in talking to a foreigner'
   at the Penn Linguistics Forum 1980 (published in
   Penn Review of Linguistics, Vol. 4, 1980).

4. Many of my comments about supportive information
   and corrective responses when misconceptions about
   the content and the structure of the data base
   are detected are based on the work of Jerry
   Kaplan and Eric Mays.