On the Use of Context in Building Spoken Language Dialogue Systems for Large Tasks

Laila Dybkjær, Niels Ole Bernsen, Hans Dybkjær, Dimitris Papazachariou
{laila, nob, dybkjaer, dimitris}@cog.ruc.dk
phone (+45) 46 75 77 11, fax (+45) 46 75 45 02
Centre for Cognitive Science, Roskilde University
P.O.Box 260, DK-4000 Roskilde, Denmark

Abstract

Context is of crucial importance to language understanding in general and plays a central role in spoken language dialogue systems design. Context, however, is hard to define. In this paper context is viewed as denoting a collection of aspects or contextual elements each of which may be defined and analysed with respect to its specific contribution to dialogue understanding. Massive exploitation of context is essential in spoken language dialogue systems design for large tasks because the feasibility of such systems demands a high degree of control of the user-system dialogue. The paper discusses in detail how knowledge about contextual elements is used in system-directed dialogue design to achieve an optimal trade-off between technological feasibility and user acceptability and to enable controlled steps in the direction of mixed-initiative dialogue. The discussion is based on the design, implementation and test of system-directed dialogue for a spoken language dialogue system and on first experiments towards achieving mixed-initiative dialogue.

Keywords: Spoken language dialogue understanding, context, large tasks, system-directed dialogue, mixed-initiative dialogue.

1 Introduction

Context is a crucial issue in the process of language understanding no matter whether understanding is accomplished by humans or by computer systems. The concept of context itself remains diffuse and an exhaustive and precise definition is perhaps not even possible at this stage, the crucial issue being that of distinguishing between linguistic aspects, on the one hand, and their context on the other. For the time being, the best approach to obtaining a firmer grasp of context seems to be by studying its parts without worrying too much about whether, at the end of the day, a certain part of context should be properly categorised as linguistic or non-linguistic. We shall assume that context is constituted by a conglomerate of different aspects which may be called *contextual elements.* By identifying and analysing such elements, their nature and contribution to language understanding, it may be possible to incrementally obtain a clearer conception of the general notion of context and its multiple roles.

The number and types of contextual elements which play a role in language understanding probably depends on, i.a., whether linguistic communication is written or oral and whether all interlocutors are humans or one of them is a computer. In this paper, we focus on the system side of spoken human-machine dialogue. The aim of the paper is to analyse the contextual dialogue elements of an implemented, application-oriented spoken language dialogue system (SLDS) and discuss how knowledge about these elements can be used in dialogue design to produce optimal trade-offs between feasibility, technological and otherwise, and user acceptability.

The paper is based on experience with the design, implementation and test of the dialogue model of the Danish dialogue system and on first experiments performed to enable the design of a more advanced, mixed-initiative, dialogue system. Section 2 briefly outlines the Danish dialogue system. Section 3 presents the dialogue model in terms of the dialogue theory which has been articulated in parallel with system construction. In Section 4 the contextual elements which constitute the dialogue context are analysed with respect to how they influence the dialogue. Section 5 discusses how the contextual elements from Section 4 can be used to enable a controlled step in the direction of mixed-initiative human-machine dialogue and which additional contextual elements may be needed for this purpose. Indications from first experiments towards this aim are reported. Finally, Section 6 presents conclusions and issues for future work.

2 The Danish Dialogue System P2

The Danish dialogue system prototype, P2 (i.e. prototype No. 2), is a real-time, 500 word vocabulary, speaker-independent SLDS in the domain of Danish domestic airline ticket reservation. The system understands continuous and spontaneous spoken Danish. The system has been developed in the Dialogue project by the Center for PersonKommunikation, Aalborg University, the Centre for Language Technology, Copenhagen, and the Centre for Cognitive Science, Roskilde University. It runs on a PC and a DSP board and is accessed via the telephone. Recently the prototype, excluding the

recogniser, has been tested with naive users [Baekgaard et al., 1995, Bernsen et al., 1995a, Dybkjær et al., 1995c]. Tests of the full system will begin shortly.

The system consists of the following main modules [Dybkjær et al., 1995a]: A continuous *speech recogniser* based on Hidden Markov Models (HMMs) produces as output a 1-best string of lexical references. A chart *parser* makes a syntactic analysis of the string, extracts the semantic contents and produces frame-like structures called semantic objects. The *dialogue handling module* interprets the contents of the semantic objects and decides on the next system action which may be to send a query to the database, send output to the user, or wait for new user input. In the latter case predictions on the next user input are sent to the recogniser and the parser. The *database* contains information on customers, timetables and flight reservations as well as knowledge on how to use this information. *Output* is based on concatenation of pre-recorded words and phrases.

3 The P2 Dialogue Model

This section presents the dialogue model of P2 in terms of a theory of dialogue functionality. The theory departs from a simple task taxonomy and proposes a systematic set of concepts and implementation strategies for dialogue management. The theory was developed in parallel with the building of the P2 system and takes some first steps in the direction of an incremental dialogue theory. Thus, increasingly complex tasks impose increased demands on the functional inventory of dialogue theory. [Bernsen et al., 1994, Dybkjær et al., 1995b]. Throughout this section, the first occurrence of each concept from the theory is written in italics and illustrated through the example dialogue in Figure 1.

The dialogue models of today's SLDSs are all task-oriented. A *task* consists of one or more sub-tasks and tasks may be embedded in, and hence be sub-tasks of, other tasks. A task involves at least one dialogue turn, i.e. a user or system *utterance*, and often involves more than one. An utterance may consist of one or more *dialogue acts* such as assertions or questions.

Task complexity varies considerably and is, at least, a function of *task size* (i.e. number of sub-tasks) and *task structure.* There are limits to the task complexity which can be handled by current SLDSs. The reservation task

---

–

S1: Hello, this is the DanLuft reservation service for domestic flights. Do you know this system?

U1: No.

S2: The system can reserve tickets for Danish domestic flights. You use it by answering the system's questions. In addition you may use the two special commands "repeat" and "correct" to have the most recent information repeated or corrected. The system will only understand you when you answer its questions briefly and one at a time. Please state your customer number.

U2: Number 1

S3: Customer number 1. How many persons will travel?

U3: Only 1 person.

S4: 1 person. What is the id-number of this person?

U4: Number 3.

S5: Id-number 2, Laila Dybkjær. Where does the journey start?

U5: Correct.

S6: Id-number 2, Laila Dybkjær. Is that correct?

U6: No, id-number 3.

S7: Id-number 3, Niels Ole Bernsen. Where does the journey start?

U7: In Copenhagen.

S8: From Copenhagen. Where does the journey end?

U8: He is going to Aalborg.

S9: Sorry, I did not understand.

U9: Aalborg.

S10: To Aalborg. Do you want a return ticket?

U10: No thanks.

S11: On which date will the journey start?

U11: On the day after tomorrow.

S12: There are not 32 days in March. On which date will the journey start?

U12: On Monday.

S13: Monday, March 20. On which time of the day?

U13: In the morning.

S14: In the morning there are departures at 6:45 and at 7:30. Do you want one of these?

U14: Repeat.

S15: In the morning there are departures at 6:45 and at 7:30. Do you want one of these?

U15: The flight at 7:30.

S16: 7:30. You have now made a reservation for Niels Ole Bernsen from Copenhagen to Aalborg on Monday March 20 at 7:30. Shall the ticket be sent or will the traveller pick it up in the airport?

U16: Please send it.

S17: You will receive the ticket in a couple of days. Do you want anything else?

U17: No thanks. Bye.

S18: Goodbye.

–

Figure 1. Example dialogue with the Danish dialogue system. The example has been crafted to illustrate, in one dialogue, all the dialogue concepts discussed in the text but is otherwise representative of the transcribed dialogues from the user test of P2. Also the example has been translated into English.

handled by the P2 system is a relatively large, well-structured task. Well-structured tasks have a stereotypical structure that prescribes which information has to be exchanged between the interlocutors to complete the task and often also roughly a natural order in which to do this. Large but well-structured tasks may for this reason be acceptably managed by system-directed dialogue and are therefore well-suited for today's SLDSs.

A central problem in SLDSs design are the hard technological constraints which typically are being imposed by the speech recogniser. The P2 system had to satisfy the following technological constraints: to ensure real-time performance at most 100 words could be active in memory at a time, and to ensure an acceptable recognition rate an average and a maximum user utterance length of 3-4 words and 10 words, respectively, had to be imposed. This meant that other design goals, such as language naturalness, dialogue naturalness and dialogue flexibility had to be traded off against these constraints [Dybkjær et al., 1993].

The resulting system uses *system-directed dialogue*. This means that the system has the main *initiative* during domain communication and decides on the next topic of the dialogue. It was therefore necessary to carefully investigate the most natural order in which to exchange the needed information. The implemented task structure conforms to the most common structure found in human-human airline ticket reservation dialogues recorded in a travel agency.

P2 preserves the initiative by concluding all its turns by a non-open question to the user, i.e. a question which asks for a well-defined piece of information (cf. Section 4). Whenever needed, users may initiate *meta-communication* to resolve misunderstanding or lack in understanding, but they can do so only by using one of the two keywords 'correct' (U5) and 'repeat' (U14). The system may initiate meta-communication either by telling the user that it did not understand what was said (S9) or by issuing error messages (S12). *Domain communication* is communication within or about the task domain whereas meta-communication is communication about the user-system communication itself, usually undertaken for purposes of *clarification* or *repair*.

The sub-task or set of sub-tasks which the user is expected to address in the next utterance constitutes the *system focus.* System focus is hardwired in the P2 system. The system focus comprises the current sub-task, i.e. the one addressed by the system in its latest question, and the user-initiated meta-communication tasks. *Predictions* on what the user will say next assist the system in identifying the sub-vocabularies and sub-grammars to be used by the recogniser and the parser in decoding the next user utterance. These predictions are based on the set of sub-tasks which are currently in system focus.

In response to each user turn P2 provides *feedback* by repeating the key information provided by the user. Feedback is mainly given as *echo feedback* (e.g., S3 and S4) which is a direct way of repeating information provided by the user while avoiding to ask for explicit user confirmation. On closing the reservation task P2 offers *summarising feedback* to summarise the commitments made (S16).

A *dialogue history* is some kind of log of the information which has been exchanged so far in the dialogue. We distinguish between four kinds of dialogue history (cf. Section 5). A *linguistic dialogue history* logs the surface language of the exchanges (i.e. their exact wording) and the order in which they occurred. The P2 system does not maintain such a history. The *dialogue contents history* records the order of sub-tasks and the semantic contents of the latest user and system utterances. In the P2 system, this history is used whenever the user initiates repair meta-communication (U5). The keyword 'correct' can be used repeatedly so that not only the most recent user input can be corrected. A *task record* logs the task-relevant information that has been exchanged during a dialogue, either all of it or that coming from the user or the system, depending on the application. In the P2 system, the task record contains information provided by the user as well as information retrieved from the database. This information is, i.a., used in the summarising feedback (S16). A *performance record* updates a model of how well the dialogue with the user proceeds and may be used to modify the way the system addresses the user. P2 does not maintain a performance record.

In human-human dialogue each participant builds a model of the interlocutor to guide adaptation of the participant's dialogue behaviour to that of the interlocutor. Furthermore, participants sometimes have a model of the interlocutor prior to the dialogue. SLDSs may also need to have, or to build, a *user model* to guide their dialogue behaviour. In P2, the system's introduction to itself (S2) as well as information about different discount types (not shown in Figure 1 because discount is only possible for return tickets) can be skipped by users who already know the system (S1, U1). Figure 2 reviews the overall characteristics of the P2 system with focus on dialogue aspects.

4 Contextual Elements in P2

Many of the dialogue aspects discussed in Section 3 are contextual in nature and both their presence and the form in which they are present is crucial to the feasibility of building a workable system. The contextual elements of P2 will be discussed in this section.

The most important contextual element is the task which heavily influences all other contextual elements. The task determines what it is, and what it is not appropriate to talk about (the domain), and thereby also strongly restricts the semantics of the words used during dialogue. The task determines the order in which it is natural to address the

sub-tasks (task structure) and hence for any point in dialogue which sub-tasks should be in focus. The more complex the task the more information must be stored in the dialogue history to ensure dialogue flexibility. The task also strongly influences the respective roles of the interlocutors, such as who mainly provides information and who mainly responds (initiative), and role-related conditions of dialogue co-operativity, such as

System	Ordering (reservation) system. DIALOG.
Generic task type	Larger, well-structured task.
Task domain	Flight ticket reservation on Danish domestic flights.
Input/output modalities	Speech input/output. Telephone.
Language	Danish.
Speech recognition	Continuous, spontaneous. Speaker-independent.
Speech production	Pre-recorded speech.
Real-time	Yes.
Vocabulary size	Medium, 500 words.
Linguistic analysis	Task-dependent parsing. Semantic frames.
Discourse phenomena	Ellipses are handled but not anaphora.
Domain knowledge	Domain facts and rules.
World knowledge	None.
Dialogue initiative	System-directed domain communication. User-initiated meta-communication through keywords.
Feedback types	Echo. Order summary.
User dialogue act types	Statements answering system questions, (keyword) commands.
Predictions	Based on system focus.
System focus	Current sub-task plus meta-communication tasks.
Linguistic dialogue history	None.
Dialogue contents history	Yes.
Task record	Yes.
Performance record	None.
User model	Minimal: expert users can de-select the introduction to the system and information on discount types.
Meta-communication	*System-initiated repair:* "Sorry, I didn't understand". Error messages. Time-out elicits repetition of latest question. *User-initiated repair and clarification:* through keywords.
Comments	User tests done on the entire system excluding the recogniser.

Figure 2. Characteristics of the Danish dialogue system.

who, if any, is the domain expert, as well as special considerations due to partner asymmetry.

4.1 Domain and semantics

When it has been decided which task(s) an SLDS should handle, it has also been decided which domain the system should possess knowledge about. During the design of the system, much effort goes into the precise domain delimitation but already from having chosen the task it is possible to determine, at a coarse-grained level, whether some piece of domain information is relevant or not. The analysis of task and domain forms the basis of the first dialogue model. In the P2 system the dialogue model was henceforth developed through a series of Wizard of Oz experiments [Dybkjær et al., 1993]. The Wizard of Oz method is an iterative simulation technique which is well-suited to, and standardly used in, SLDS dialogue model development. [Fraser and Gilbert, 1991]

The task and the domain will roughly determine which sub-language is relevant and hence the semantics of the words in the vocabulary. The result is a tremendous reduction in semantic complexity. The Danish word for discount (rabat), for instance, has two senses one of which is ‘discount’ and the other is ‘roadside’. The latter sense clearly is beside the point in a flight reservation task and is not considered in the P2 system. During the Wizard of Oz process not only the domain but also the sub-language is defined in more detail.

4.2 Task structure and focus

As mentioned, the P2 reservation task is a well-structured task whose structure forms part of common knowledge.

The system's users (interlocutors) know and accept that a certain specific amount of information must be communicated to the system in order to succeed in making a valid ticket reservation. This makes it much easier to design a dialogue which users will find acceptable even if it is directed by the system, something which is further facilitated by the presence of common expectations as to the natural order in which to provide the information needed for reservation. For example, one should not address the sub-task of identifying the hour of departure before having identified an itinerary and a date of departure.

The focus set should cover the natural ways in which to continue the dialogue from a certain point. The degree to which this has been achieved influences the user acceptability of the system. The more natural restrictions there are on the task structure and the fewer possible next sub-tasks there are at any given point in dialogue, the easier it is to define a focus set which is both small and appropriate. The well-structuredness of the reservation task made it possible in P2 to meet the technological constraint of a maximum active vocabulary size of 100 words (cf. Section 3).

4.3 Dialogue history

P2 has a task record and a dialogue contents history. The task determines the precise requirements for what to put into the task record. As long as the system is missing information from the user, which is necessary to complete the reservation task it will continue to solve its ordered series of sub-tasks. When the last piece of information has been acquired from the user, the system will summarise the commitments made by the user and end the dialogue. The dialogue contents history, on the other hand, is maintained for meta-communication reasons and therefore is not determined by the task but rather by decisions on how flexible the meta-communication dialogue should be. The more flexibility, the more demands there will be on the dialogue contents history.

4.4 Interaction roles

The task, its structure and the roles it imposes on the interlocutors also influence initiative. By definition, the system is expected to be the more knowledgeable interlocutor on the domain and the procedures for making a reservation. It seems natural, therefore, that it is mainly the system which has the initiative and guides the dialogue to make it proceed as smoothly as possible. In the human-human dialogue recordings we made in a travel agency, the normal reservation procedure appeared to be that the customer initially stated what s/he wanted (e.g., two return tickets to Copenhagen) whereupon the travel agent took over the initiative to ask for the missing information. Apart from the initial user initiative, the P2 system behaves very much like the travel agent. The P2 dialogue is entirely and not only mainly directed by the system. This was necessary in order to satisfy the technological constraints on user utterance length and active vocabulary size. Initiative is a very strong means by which to control sublanguage, focus set and utterance length. If all initiative lies with the system, it fully controls what to talk about next, i.e. the focus set. Furthermore, the answers to questions, which constitute the expected user speech act in system-directed dialogue, will typically be shorter than the questions themselves because answers are often expressed elliptically whereas questions tend to be expressed in full sentences.

The P2 dialogue has been made system-directed by having the system conclude all its turns by a non-open question in order to preserve initiative. Non-open questions are questions which address a well-defined topic and ask for a specific piece of information. The P2 non-open questions may be categorised into four types. One type invites a yes/no answer, e.g.: "Do you want a return ticket?"

The second type is the multiple choice question where the user is expected to choose an element from an explicit list of alternatives, e.g.: "Shall the ticket be sent or will the traveller pick it up in the airport?"

The third type of question invites the user to state a proper name or something similar, such as the name of an airport or an id-number. Id-numbers are used instead of person names which cannot be dealt with because of vocabulary limitations. Users' names are looked up in the database by using the id-number as key. For instance: "Please state the id-number of the traveller."

The fourth type is the most open type, i.e. the one which allows the broadest variety of formulations but which still concerns a specific topic, such as date of departure. E.g.: "On which date will the journey start?"

None of these types of question invites the user to take over the initiative. The approach seemed to work in practice, both in the last Wizard of Oz iteration and in the user test of the final system. In both cases four user questions were found out of 881 and 998 user utterances, respectively. So with very few exceptions users actually limited themselves to answering the questions.

The roles adopted by the interlocutors are important to how a task is performed, and the roles are to some extent determined by the task itself. In the case of a reservation task one must expect one of the interlocutors (the system) to be a domain expert who knows everything about departure times, free seats, fares etc., and is able to execute reservations. The other interlocutor (the user) can be expected to have at least some basic knowledge about what is a reservation because this is part of common knowledge. Otherwise, we do not know if the user is highly experienced or whether it is the first time s/he attempts to make a reservation. The system must, however, be able to cope with users of all kinds from the very experienced to novices. Since users probably will have different needs depending on which part of the scale they belong to, the system must to some degree be user adaptive to ensure good performance. A user model helps the system pay attention to differences in user task skills by, e.g., providing more information to some users than to others. As mentioned, the user model for the P2 dialogue system is very simple. The system's introduction to the dialogue, and the discount information are the only pieces of information which can be left out by the expert user

and which provide the only additional support for the novice. All other system turns are strictly concerned with obtaining the necessary information from the user and must be included.

SLDSs are designed for co-operative users. If users are not co-operative, they cannot expect to have a successful dialogue with the system because it will not understand them. However, the best way in which to be co-operative depends on the properties of the interlocutor. There are, to be sure, generic principles of co-operative conversation or dialogue, such as the Gricean maxims [Grice, 1975]. But if, in a particular dialogue, there is anything special about one of the interlocutors, which the other interlocutor should take into account, then it is up to the dialogue partner who is in a special position or condition to tell the interlocutor about it so that the interlocutor can adapt his or her co-operative dialogue behaviour accordingly [Bernsen et al., 1995b]. An SLDS is in an asymmetrical, and hence special, relationship to the user by being an expert in a limited domain and knowing absolutely nothing about anything else. Moreover, the SLDS has limited linguistic capabilities. Such asymmetries should be communicated to users as early in the dialogue as possible in order that they may adapt their co-operative dialogue behaviour. The P2 system does this by providing a series of admonitions to users in its dialogue introduction (cf. Figure 1, S2).

4.5 Summing up

We conclude that the design of SLDSs as complex as P2 makes ample use of contextual elements. This would seem essential to successful design because, to be feasible at all, such systems require a high degree of control of the environment in which the dialogue with users takes place. Recent tests indicate that P2 satisfies the imposed technological constraints (cf. Section 3) and is, in most respects, functionally adequate and usable. The development of the system was made possible by the use of contextual elements. This is illustrated in Figure 3 which reviews the contextual elements enabled by the large well-structured P2 reservation task.

The primary remaining problem is that often the reservation task is not a pure reservation task but also includes information-seeking sub-tasks. Typically, the user knows facts such as who is going to travel and where the person(s) is going. Often, however, the precise hour of departure is not known by the user in advance, and sometimes the choice of departure date will depend on the available hours of departure. In such cases, it is likely that users will ask questions of the system in order to obtain the missing information which, again, will enable them to continue answering the questions posed by the system. Actually three of the four user questions asked in the user test of P2 concerned available departure times. For example:

System: On which time of the day?

User: Which possibilities are there?

But P2 does not understand user questions. In order to make P2 capable of handling the common reservation task which also to some degree include information-seeking (and hence user questions) it is necessary to introduce mixed initiative user-system dialogue.

In Section 5 we discuss the consequences of allowing users to ask for information during reservation.

5 Contextual Elements in Mixed-Initiative Dialogue

5.1 The problem of mixed initiative dialogue

As indicated at the end of Section 4, the task of making a reservation often includes the sub-task of seeking and providing information about that which can be reserved. To properly complete the task of booking a flight ticket, for instance, one often needs on-the-spot specific informa-

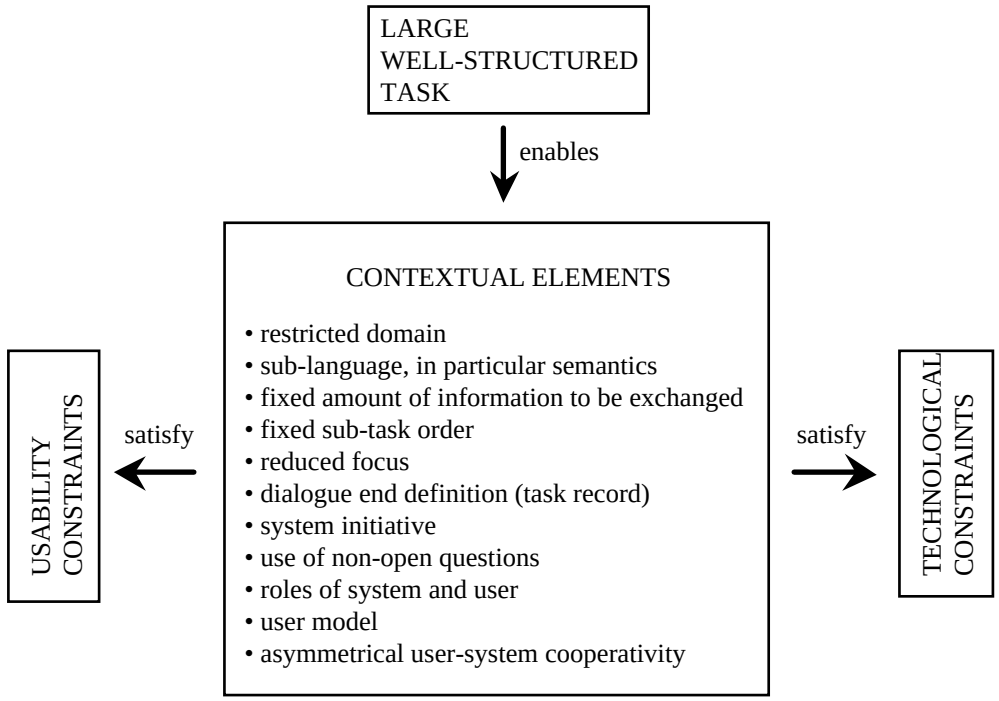

Figure 3. The well-structured task chosen for P2 enables contextual elements which help satisfy technological constraints and usability constraints.

tion about fares, timetables or other aspects of the airline travel domain. The task of reservation, in other words, is in many cases a task of *informed reservation.* System-directed dialogue is incapable of enabling the informed reservation task. The reason is that the system is unable to predict when, during the reservation dialogue, the user (interlocutor) might suddenly want to ask for some piece of information in order to be able to continue making the reservation. To ask for information means taking over the initiative from the system, so that the dialogue effectively becomes a mixed initiative dialogue.

The pure airline ticket reservation task belongs to the class of large well-structured tasks (cf. Section 3) which can be handled through system-directed dialogue. The informed airline ticket reservation task belongs to a different and more complex task category, namely that of large *ill-structured* tasks. Such tasks are characterised by a large number of optional sub-tasks. Each of these sub-tasks may be well-structured in itself but the overall task becomes ill-structured because of the optional character of the many sub-tasks it includes. This means that the system cannot have a valid stereotype that tells which sub-tasks the user wants to accomplish and in which order [Bernsen et al., 1994, Dybkjær et al., 1995b]. In the absence of such a stereotype, and to some extent also because the task size is large, system-directed dialogue is too inefficient for the negotiation of such tasks. In a nutshell, if you want to ask me (or the system) about something and if I have no idea about what you want to ask me about, it is infinitely more efficient that you pose me the question than that I have to question you to find out what you want to ask me about.

This leads to the main question of this section: how is mixed initiative SLDSs possible for large, ill-structured tasks and, in particular, how may contextual dialogue elements be used to enable the construction of mixed initiative SLDSs? A mixed initiative SLDS for informed reservation would require relaxation of the technological constraints of P2 (cf. Section 3). We shall assume that a limited enlargement of system focus and user utterance length, sufficient for the approach to be presented below, will be possible (cf. Section 3).

However, mixed initiative SLDSs are not currently feasible for large, ill-structured tasks in the general case. Current exploratory design projects include mixed initiative systems for small ill-structured tasks [Kanazawa et al., 1994, Smith, 1991], which are not really relevant to our problem. The reasons why systems like those described in Kanazawa et al. [1994] and in Smith [1991] can allow mixed-initiative without restriction are that the task is small and that the vocabulary is small. This allows all possible sub-tasks to be in focus at the same time, especially when word-spotting is being used instead of full syntactic-semantic representation of user utterances. The use of word-spotting makes it less relevant to consider limiting the length of user utterances.

Mixed initiative dialogue on large ill-structured tasks has been marginally realised in the SUNDIAL system [Peckham, 1993]. The full-fledged approach adopted in the Esprit PLUS project would seem to have failed [Grau et al., 1994]. The latter project demonstrates that the problems involved in solving the general case of mixed initiative SLDSs for large, ill-structured tasks not only derive from technological constraints on system focus and user utterance length, but derive as much from unsolved scientific problems in language processing.

5.2 Using contextual elements in addressing the mixed initiative problem

In view of the difficulties in solving the general case of mixed initiative SLDSs for large, ill-structured tasks, we are currently exploring ways to make a controlled step towards a limited solution in the case of informed reservation. The idea is the following: at the general level we assume that the user's goal is to make a reservation. This allows us to maintain the stereotypical structure of the reservation task as a 'backbone' for dialogue design. This means that the task context will (still) strongly constrain the dialogue behaviour of co-operative users. They can be expected to follow the overall system-directed course of the dialogue and to only take over the initiative when they need information from the system in order to proceed in making the commitments needed for reservation. Such questions, moreover, can be expected to primarily concern sub-tasks closely associated to the one currently addressed by the system. There remains, however, a number of important unknowns. Firstly, we need further constraints to ensure limited user utterance length in the cases where users take over the initiative from the system. Secondly, we probably need additional constraints to ensure that users will not be asking for arbitrary pieces of domain information at arbitrary points during the dialogue. And thirdly, we must make sure that the system has the linguistic capabilities to detect the shift in initiative which occurs when, at arbitrary points during dialogue, users request domain information.

When users have more initiative the utterance length can be expected to grow compared to system-directed dialogue in which an elliptical or otherwise brief answer typically will do. Since utterance length is a critical parameter we must ensure limited growth. Terse system language is known to have a positive effect on utterance length [Zoltan-Ford, 1991], and the mainly system-directed dialogue which proceeds through non-open questions that are only interrupted by user requests for information probably also will influence user utterance length in the right direction. In addition, the system will give admonitions to users to express themselves briefly in order to be understood and to ask only one question at a time.

The system will not be able to handle arbitrary requests for information at arbitrary points during dialogue, because of the uncontrolled growth in the focus set. Users should not ask, e.g., about departure times when the system addresses the destination. For some sub-tasks, such as number and names of travellers, we would expect no questions at all. Users can be expected to know who is going to travel without having to negotiate this with the system. For other sub-tasks, however, it may be highly relevant to ask for information. For instance, users often do not know the precise departure and arrival times and must be informed on these by the system. Or users will want to know about reduced-fare departures before committing to a specific departure time or even departure date. We assume that it will be possible to 'cluster' such dependencies between system questions and relevant user questions such that the system focus set can still be kept limited.

When requests for information are allowed during reservation dialogue more than one type of user dialogue act is allowed as well. The important point is that, due to the informed reservation task context, only two different basic types of dialogue act are relevant and must be distinguished by the system, i.e. reservation commitments and requests for information. The system must be capable of detecting when the user wants information and when the user provides a piece of information to fill a slot in the reservation record. We have begun a series of Wizard of Oz experiments to investigate how this might be possible. Two general cases may be distinguished.

In the first general case, the contents of a user utterance cannot be used to fill a slot in the reservation record (e.g. "What are the possibilities?") or provide partial information for a slot (e.g. "Which departures are there on Saturday morning?"). In such cases, the system should treat this user utterance as a request for information no matter whether it is phrased as a question or not. This rule actually solves the problem that some requests for information may be hard to detect because their status as a question is mainly expressed through intonation. Intonation is not yet exploited in realistic SLDSs although the possibility is subject to ongoing research. An example from our experiments of such a question mainly expressed through intonation is the following:

And on Saturday morning before 12?

In the second general case, the user's utterance provides information that could fill a slot in the reservation record. In this case there are two possibilities. The obvious possibility is that the utterance fills a slot in the reservation record. However, the utterance might ask for information instead, such as the following: "Is there a flight at 7.15?". Our small corpus does not contain examples of this kind so far. Possible utterances are of the following types:

a) Direct polar question, e.g.: Is there any flight at 7:15 on Monday?

b) Indirect question, e.g.: Can you tell me if there is a flight at 7:15 on Monday? or: Do you know if there is a flight at 7:15 on Monday?

c) A statement, e.g.: I want to know if there is a flight at 7:15 on Monday, or: Tell me if there is a flight at 7:15 on Monday.

All of these utterances refer to the felicity conditions of the requests for information, i.e. the direct polar question refers to the propositional content condition, the indirect question refers to one of the preparatory conditions, and the statement refers to the sincerity and essential conditions. Moreover, none of the utterances only addressing reservation could refer to the felicity conditions of the requests for information, thus acting as indirect requests for information [Searle, 1969]. This suggests that the system can use the semantics of the sentence to identify the cases in which it has to provide information. When the semantics of the sentence do not refer to the felicity conditions of requests for information, then the system may safely accept the utterance as a reservation commitment. In this situation, there will be no problem in identifying the following utterance as an answer and not as a request for information:

Could I have the Saturday morning flight at 9:20?

The background context as defined by the specific task and the roles of the speakers (user/system) can also predict the possible and impossible utterances that will refer to the felicity conditions of the requests for information. For instance, it would not be logical, and hence not co-operative, that a user would ask for information while questioning at the same time the nature of his/her dialogue act (i.e. the essential condition) or even his/her will to make the system provide the information (i.e. the sincerity condition). Also, it would not be logical, and hence not co-operative, that the user would engage in the communication if s/he questioned his/her belief in the abilities of the system. Moreover, it would seem redundant for the speaker to state either the system's ability or his/her belief in the system's ability, because these form the basis of the respective roles of user and system. However, a study of a large corpus is necessary in order to test the above predictions.

6 Conclusion

We have provided evidence on the crucial importance of context in the design of SLDSs for large tasks. This, again, illustrates the importance of context for language understanding in general. It remains unclear how to distinguish between linguistic and non-linguistic dialogue context. Arguably, however, most of the contextual elements discussed in this paper are non-linguistic contextual elements which, at best, belong to the poorly defined domain of pragmatics. Given these contextual elements, it has been possible to consider linguistic dialogue entities such as sub-language vocabulary and semantics (the lexicon), and basic types of dialogue act as dependent variables which depend on the nature of the task.

We have preliminarily discussed how to make a controlled step in the direction of mixed initiative dialogue exemplified by the informed reservation task. Due to the nature of this task we only need to distinguish two basic types of user dialogue act expressing either a request for information or a commitment to reservation. It remains to be seen if the linguistic mechanisms proposed above will be sufficient for enabling limited mixed-initiative dialogue.

Possible next steps would be to allow even more user initiative in domain communication, e.g. by letting the system ask more open questions, and to allow more flexible forms of meta-communication that are not based on keywords. The increased complexity due to increased user initiative in domain communication can be expected to concern focus and utterance length but not the basic distinction between two types of dialogue act. If, however, we allow non-keyword-based meta-communication we can no longer take all information-requesting dialogue acts to mean requests for information at the domain level.

Acknowledgements

The work described in this paper was carried out under a grant from the Danish Government's Informatics Research Programme whose support is gratefully acknowledged. The EU's Human Capital and Mobility Programme supports Dimitris Papazachariou's work at the Centre for Cognitive Science.

References

[Baekgaard et al., 1995] Baekgaard, A., Bernsen, N.O., Brøndsted, T., Dalsgaard, P., Dybkjær, H., Dybkjær, L., Kristiansen, J., Larsen, L.B., Lindberg, B., Maegaard, B., Music, B., Offersgaard, L., Povlsen, C.: The Danish Spoken Dialogue Project - A General Overview. *Proceedings of the ESCA workshop on Spoken Dialogue Systems,* Vigsø, Denmark, May 30 - June 2, 1995.

[Bernsen et al., 1995a] Bernsen, N.O., Dybkjær, H. and Dybkjær, L.: Exploring the Limits of System-Directed Dialogue. Dialogue Evaluation of the Danish Dialogue System. *Proceedings of EUROSPEECH '95,* Madrid, September 1995.

[Bernsen et al., 1995b] Bernsen, N.O., Dybkjær, H. and Dybkjær, L.: Co-operativity in Human-Machine and Human-Human Spoken Dialogue. (Submitted).

[Bernsen et al., 1994] Bernsen, N.O., Dybkjær, L. and Dybkjær, H.: A Dedicated Task-Oriented Dialogue Theory in Support of Spoken Language Dialogue Systems Design. *Proceedings of the ICSLP Conference,* Yokohama, Japan, September 1994, 875-78.

[Dybkjær et al., 1993] Dybkjær, H., Bernsen, N.O. and Dybkjær, L.: Wizard-of-Oz and the Trade-off between Naturalness and Recogniser Constraints. *Proceedings of EUROSPEECH '93,* Berlin, September 1993, 947-50.

[Dybkjær et al., 1995a] Dybkjær, H., Bernsen, N.O. and Dybkjær, L.: Dialogue Development and Implementation in the Danish Dialogue Project. To be published in *European Speech Projects,* Springer Verlag 1995 (in press).

[Dybkjær et al., 1995b] Dybkjær, L., Bernsen, N.O. and Dybkjær, H.: Different Spoken Language Dialogues for Different Tasks. A Task-Oriented Dialogue Theory. To be published in Human Comfort and Security, *Springer Research Report 1995* (in press).

[Dybkjær et al., 1995c] Dybkjær, L., Bernsen, N.O. and Dybkjær, H.: Scenario Design for Spoken Language Dialogue Systems Development. *Proceedings of the ESCA workshop on Spoken Dialogue Systems,* Vigsø, Denmark, May 30 - June 2, 1995.

[Fraser and Gilbert, 1991] Fraser, N.M. and Gilbert, G.N.: Simulating speech systems. *Computer Speech and Language 5*, 1991, 81-99.

[Grau et al., 1994] Grau, B., Sabah, G. and Vilnat, A.: Control in Man-Machine Dialogue. *THINK,* Vol. 3, Tilburg, The Netherlands, May 1994, 32-55.

[Grice, 1975] Grice, P.: Logic and conversation. In P. Cole & J.L. Morgan (Eds.), *Syntax and Semantics Vol. 3: Speech Acts* (pp. 41-58), New York: Academic

Press 1975. Reprinted in Grice, P.: *Studies in the Way of Words.* Cambridge, MA: Harvard University Press 1989.

[Kanazawa et al., 1994] Kanazawa, H., Seto, S., Hashimoto, H., Shinchi, H. and Takebayashi, Y.: A user-initiated dialogue model and its implementation for spontaneous human-computer interaction. *Proceedings of the ICSLP 94*, Yokohama, September 1994, 111-114.

[Peckham, 1993] Peckham, J.: A New Generation of Spoken Dialogue Systems: Results and Lessons from the SUNDIAL Project. *Proceedings of EUROSPEECH '93*, Berlin, September 1993, 33-40.

[Searle, 1969] Searle, J.R.: *Speech Acts*. Cambridge: Cambridge University Press, 1969.

[Smith, 1991] Smith, R.W.: A Computational Model of Expectation-Driven Mixed-Initiative Dialog Processing. Ph.D. Thesis, Department of Computer Science, Duke University, Durham, NC 27706, USA, October, 1991.

[Zoltan-Ford, 1991] Zoltan-Ford, E.: How to get people to say and type what computers can understand. *International Journal on Man-Machine Studies*, Vol. 34, 1991, 527-547.