TYPES OF USER PROBLEMS IN DESIGN

A study of knowledge acquisition using the Wizard of Oz

Niels Ole Bernsen, CCI, Roskilde University and Risø National Laboratory

Summary: The paper describes and exemplifies the types of user problems which were addressed during the user-system interaction design phase in the development of a spoken language dialogue system. The user problem types are analysed with respect to the design commitments they violate and the justification for those commitments. Design commitments are represented in the CO-SITUE frame notation for the representation of evolving design space structure. It is shown how a limited number of design commitments may support the design reasoning necessary for resolving the large majority of usability design problems which were encountered.

Keywords: Usability engineering, user problems in design, design space structure, Design Rationale.

1. Introduction

The CO-SITUE framework is a framework for analysing information systems design processes from the point of view of usability engineering (Bernsen 1993a,b). The CO-SITUE frame notation provides a representation of the basic aspects of and constraints on the design space within which the design of a particular artifact takes place and supports an incremental representation of the evolving artifact during design. CO-SITUE stands for the following basic aspects of the design space and hence of the unfolding artifact:

C = Collaborative aspects.
O = Organisational aspects.
S = System aspects.
I = Interface (or more generally: system Image) aspects.
T = Task aspects including task domain aspects.
U = User aspects.
E = User experience aspects.

In addition, the CO-SITUE notation includes a number of generic constraints and criteria which are generally relevant to information systems design. These are:

- overall design goal;
- general feasibility constraints;
- scientific and technological feasibility constraints;
- designer preferences;
- realism criteria;
- functionality criteria; and
- usability criteria.

The CO-SITUE notation incrementally represents design commitments but does not represent the design *reasoning* which led to those commitments. Many attempts are currently being made in HCI to find ways of perspicuously capturing design reasoning or Design Rationale. In particular, CO-SITUE is compatible with, e.g., the Design Rationale/Questions, Options and Criteria (DR/QOC) approach to design space analysis which provides a notation for representing detailed analyses of design reasoning (MacLean et al. 1991, MacLean et al. 1993). The possible integration of the two approaches is being investigated as part of the ESPRIT Basic Research project AMODEUS-2 (Barnard 1992). In contrast to DR/QOC, CO-SITUE represents overall design space *structure* and hence aspects which need to be taken into account in design reasoning. The hypothesis is that, in general, Design Rationale approaches to design space analysis presuppose a CO-SITUE-like overall design space representation (Bernsen 1993b). In the present paper, a coarse-grained DR/QOC notation is used to represent design reasoning on trade-off problems (see Sect. 5 below).

This paper analyses the knowledge acquisition and dialogue specification phase of the design of a first spoken language dialogue system prototype (Dybkjaer and Dybkjaer 1993). The following sources of information feeding into design reasoning and commitments were involved in the usability engineering process of developing the first prototype:

- literature on and knowledge of previous systems, development methodologies and software tools;
- empirical results from field studies;
- designer craft skills;
- a strong empirical rapid prototyping method (the Wizard of Oz) based on user task scenarios was used for user-system interaction development;
- basic tasks were used as the backbone of user-system interaction development;
- step-by-step designer walkthroughs were carried out on specified tasks;
- the CO-SITUE notation was used to represent the evolving artifact during design as a result of design commitments;
- a 'lightweight' option/criteria Design Rationale representation was used to analyse and resolve complex design problems.

No science-based approaches such as cognitive walkthroughs (Lewis et al. 1990), claims generation support (Carroll and Rosson 1992) or user modelling (Barnard and May 1993) were used to support the usability engineering of the artifact. Rather, it is envisioned that the realistic design process material provided in this paper might serve as a testing ground for evaluating proposed analytical design support tools such as those just mentioned. This raises questions such as: could a particular analytical approach have discovered a particular user problem? Could it have solved the problem? If, for a given analytical approach, the answer is no in one or both cases, what might be done to extend the coverage and/or power of that approach? In addition, approaches to explicit design space representation and analysis such as those presented in this paper constitute, arguably, necessary bridges to the realistic application of science-based approaches to usability engineering (cf. Sect. 6 below).

A CO-SITUE analysis was already performed during the initial artifact specification phase preceeding knowledge acquisition (Bernsen 1993b). The current purpose is to complete the CO-SITUE analysis of the design process and arrive at a comprehensive view of the user problems which were identified. The elements involved are the following:

- the *results of the initial artifact specification phase* as represented in CO-SITUE notation;

- the *evolving user-system interaction (or dialogue) structure* as represented in graph notation;

- a set of *types of user problems* with the spoken language dialogue system as identified by designers or users during the knowledge acquisition and dialogue specification phase. Each type of user problem will be illustrated by examples;

- a set of *design commitments* the violation of which caused the identified user problems;

- a set of *justifications* of the design commitments.

Except for the dialogue graph notation which only came into existence in the design phase to be described below, these elements are already present in the CO-SITUE analysis of initial artifact specification and now have to be elaborated to take into account the design commitments made during the knowledge acquisition phase. The results of early artifact specification are summarised in Section 2. Section 3 describes methodology. Sections 4 and 5 describe and exemplify the types of user problems actually found during knowledge acquisition and present and justify the design commitments whose violation caused the identified user problems. Section 6 is a discussion of results.

2. Results of Initial Artifact Specification

Below follows a CO-SITUE representation of the artifact as it had evolved during the initial design phase.

CO-SITUE No. (4)

A. General constraints and criteria

Overall design goal:
- Spoken language dialogue system prototype operating via the telephone and capable of replacing a human operator;

General feasibility constraints:
- 9 m/y's available for the first version of the prototype;
- Limited machine power available;

Scientific and technological feasibility constraints:
- Limited capability of current speech and natural language processing;
- Open research questions;

Designer preferences:
- Use of the Dialogue Design Tool (DDL);

Realism criteria:
- The artifact should meet real and/or known user needs;
- The artifact should be preferable to current technological alternatives;
- The artifact should be tolerably inferior to the human it replaces, i.e., it should be at least usable (to be expanded in the usability criteria);

Functionality criteria:
- Make sure that the artifact *can do* the tasks done by the human it replaces;

Usability criteria:
- Maximize the naturalness of user-interaction with the system;

- Unless a naturalness criterion cannot be met for feasibility reasons, it should be incorporated into the artifact being designed;
- Constraints on system naturalness resulting from trade-offs with system feasibility have to be made in a principled fashion based on knowledge of users in order to be practicable by users;
- Constraints on system naturalness have to be clearly communicated to users;

B. Application of constraints and criteria to the artifact within the design space:

C = Null.
O = Null.

S = 500 words vocabulary;
Ability to repair if understanding fails;
Natural grammar;
Appropriate semantics;
Natural discourse handling;
Close-to-real-time response;
Limited speaker-independent recognition of continuous speech;
Clear and comprehensible communication of what the system can and cannot do;
Intelligible, practicable and principled limitations on natural system performance;
Make system limitations clear to users from the outset;

I = Spoken telephone dialogue

T = Obtain information on and perform booking of flights between two specific cities;
Use single sentences (or max. 10 words);
Understand user utterances in task domain;
Large enough task-related vocabulary;
Complete task domain information;

U =

E = Novices, walk-up-and-use users;
Intermediates;
Experts;

C. Hypothetical issues:
- How to accommodate experienced users?
- Is a vocabulary of 500 words sufficient to capture the sublanguage vocabulary needed in the task domain ?

D. Conventions:
The CO-SITUE aspects have been explained in Section 1 above.
CO-SITUE No. (n) indicates the number of the current CO-SITUE specification.
'Null' means that the artifact does not embody a certain aspect of CO-SITUE. Note that, in the present design process, the *C* and *O* aspects have been temporarily set to null during design of the first prototype and will be reconsidered during design of the second prototype.

Based on its predecessors CO-SITUE (1)-(3), CO-SITUE (4) rather succinctly represents the state of progress of artifact specification on completion of the initial design phase. Everything that is represented in CO-SITUE (4) is assumed to be actually or potentially relevant from the point of view of artifact usability. Each CO-SITUE frame (n) inherits the contents of its predecessor (n-1); adds design commitments to further constrain the design process; and ideally represents a consensus among the designers. Section (A) of a frame represents general constraints and criteria on the design process. These constraints and criteria have been expressed only to a point where they may still be seen as relevant to a larger class of design processes. Section (B) of a frame represents those general constraints and criteria *as applied (or interpreted) with respect to the particular artifact being designed.* This application process is a process of invention, constraint-application and trade-off each step of which results in further design commitments and hence in constraints to be taken into account during subsequent artifact specification. These constraints are being incrementally represented in a series of CO-SITUE frames. Note that, as said earlier, a CO-SITUE frame does not include design reasoning or justification of the particular design commitments it represents. From the point of view of adopting a CO-SITUE design space representation, the question of providing an appropriate and useful representation of Design Rationale is the question of finding ways to link two successive CO-SITUE frames in a series by succinctly representing the design reasoning which led from the former to the latter.

Whereas the general constraints and criteria which represent the overall design goal, general feasibility, scientific and technological feasibility, designer preferences, realism and functionality are easily understood and ubiquitous in artifact design, the criterion of usability represents the much sought-after criterion of HCI research which has been called 'user-friendliness' or 'naturalness' among many other names. Unfortunately, in many or most cases of artifact design, usability does not have a well-defined meaning and has to be replaced by more operational and often more domain- or task-specific terms which normally only succeed in capturing part of what we want to achieve in usability engineering. As it happens, however, the term naturalness does have a reasonably clear meaning in the particular context of spoken language dialogue systems (Bernsen 1993c). Speaking is one of our natural capabilities, whereas we are not naturally endowed with a capability for interaction with, e.g., many forms of complex multimodal systems. Thus the concept of naturalness may in this case serve to capture many, but not all, of the usability issues to be dealt with during spoken language dialogue systems design.

3. Methodology

One of the tasks during the knowledge acquisition design phase which follows the initial artifact specification phase and takes place concurrently with the more detailed specification of system architecture, is to design the structure of the dialogue between system and user. This design phase is crucial from a usability engineering point of view and deals, in effect, with the design of the *interaction* between user and system. Because of the simplicity of the interface, which is a telephone, and the cognitively sophisticated form of interaction between user and system through spoken natural language, this design phase lends itself to the use of a rapid prototyping technique called the Wizard of Oz. This technique allows designers to closely simulate the system to be designed through acting as if they were the system speaking to users. If and when the designed dialogue structure turns out to successfully meet a number of quite severe technological constraints (see CO-SITUE (4)), it can be directly implemented upon completion of the simulations. To achieve this result it is necessary to go through a number of iterations of Wizard of Oz simulations involving experimental subjects, dialogue recordings on tape, transcriptions and analyses of these, and system dialogue refinement based on analyses of previous generations of system dialogue structure. Although the Wizard of Oz technique has a number of shortcomings (Dybkjaer and Dybkjaer 1993), it is a powerful empirical method for development and testing of artifact usability prior to implementation. This method is particularly valuable in cases where the risk and cost of producing unusable prototype implementations are high. If only this technique could be generalised to handle all the types of user-system interaction to which it is in principle appropriate, we would have gone a long way towards realising an efficient empirical methodology for usability engineering. Unfortunately, such a generalisation has not yet been achieved

(Salber and Coutaz 1993). The Wizard of Oz technique is still mainly being used for spoken or written language dialogue systems design.

In addition to helping in creating the user-system dialogue structure, the Wizard of Oz technique supports the definition of a sub-language including the vocabulary and grammar to be implemented as part of the dialogue system. This sub-language has to satisfy opposing constraints some of which, such as number of word form types and average user utterance length, are directly measurable (cf. CO-SITUE (4)). During the knowledge acquisition process these parameters are continuously being measured and brought within the prescribed limits imposed by the speech recognition technology used, while at the same time attempts are made to respect the functionality and usability constraints defined during initial artifact specification (Dybkjaer et al. 1993). Users address the simulated system on the basis of written *scenarios* given to them in advance, according to which they are to perform specific domain-relevant tasks such as, in the present case, making flight reservations, obtaining flight information or changing reservations.

During our use of the Wizard of Oz technique, the user-system dialogue structure was incrementally developed and represented in a series of six conventional graph structures, each representing the entire user-system interaction as enacted by the wizard in a particular generation of Wizard of Oz simulations. The user-system interaction structure eventually developed into a complex graph expressing 4 basic tasks, namely, *Wellcome and Introduction to the System*, *Reservation*, *Change of Reservation* and *Information*. Identification of the user problem types to be presented below has been based primarily on analysis of these graph structures. For each change in a particular graph generation as compared with the immediately preceeding generation it was explored why the change had been made. Changes could have been made for a number of reasons, several of which are irrelevant to the existence of actual or potential user problems. For instance, changes were often made to achieve consistency with changes made elsewhere in the graphs, or changes were made to achieve increased economy or reduce reduncancy of system dialogue turns without any obvious user problems being involved (apart from such facts as that users (and designers) appreciate consistency and do not appreciate to have to answer the same question twice). I have also ignored a number of 'false starts' in the development of system functionality where it was obvious what should be done to satisfy a certain user need in the task domain but where the way to do this was less obvious, with the result that several design attempts were made before a proper solution was found. If, however, a graph structure change was made because of the discovery or anticipation of a user problem, the problem was analysed by asking the following questions:

(a) What is the user problem?
(b) Which design commitment(s) had been violated by the user-system interaction structure prior to the change of that (graph) structure? If the appropriate design commitment was already present in CO-SITUE (4), this commitment was referred to. If the design commitment was not present in CO-SITUE (4), a new commitment had to be expressed and entered into the CO-SITUE representation.
(c) What is the justification of the design commitment as expressed in terms of user properties?
(d) Which examples were found of this particular user problem type and how were the particular examples identified: through analysis by the designers themselves or through empirical discovery of user problems?

It may be of interest to note that, in a large number of cases, the user problems identified could be seen to arise from a simple violation of one specific design commitment. However, another class of user problems involved *trade-offs* between different design commitments or constraints. A 'lightweight' DR/QOC notation is used below to represent such trade-offs.

4. Types of User Problems Discovered During Knowledge Acquisition

1. (a) *The system specifically announces to have a larger task domain coverage than it actually has.*
(b) Violation of the CO-SITUE (4) design commitment: Clear and comprehensible communication of what the system can and cannot do.
(c) Justification: Risk of communication failure in case of lacking knowledge about what the system can and cannot do. In this case violation of the design commitment leads users to have exaggerated expectations about the system's task domain coverage which again leads to frustration during use of the system.

(d) Examples:
1.1. In the Opening Information about the system’s task domain: 'Danish domestic flights' should be 'domestic flights between Copenhagen and Aalborg'.
Discovered through design analysis.

Several identical examples of this user problem were found. No other examples were found.

2. (a) *The system has a smaller task domain coverage than it ought to have, given its general task domain purpose.*
(b) Violation of the CO-SITUE (4) design commitment: Complete task domain information.
(c) Justification: Risk of communication failure in case of lacking task domain information. Full task domain coverage is necessary in order to satisfy all relevant user needs in context. Otherwise, users will become frustrated when using the system.

(d) Examples:
2.1. The group travel option is not offered to groups of more than 9 persons (WoZ 5, i.e., iteration No. 5 among the six user-system interaction design iterations performed using the Wizard of Oz method). In WoZ 6 the group travel option is offered to groups of more than 9 persons which are told, however, that the system cannot handle such reservations and that reservation should be made by contacting the travel agent. The correct solution, however, is to announce this limitation in the Introduction to the System (cf. example 2.4 below). This would respect the CO-SITUE (4) design commitment: Make system limitations clear to users from the outset.
Discovered through design analysis.
2.2. Flight Times: The possibility was overlooked that all flights on a specific day might be fully booked (WoZ 5). In WoZ 6 a new sub-dialogue was introduced to handle such cases.
Discovered through design analysis.
2.3. Change of Reservation: The possibility was overlooked that a client may simultaneously have more than one reservation in the database so that the reservation to be changed would have to be identified among those.
Discovered through design analysis.
2.4. Clients are not warned from the outset that the system cannot handle reservations involving pets, special luggage or need of being accompanied by flight personnel. The correct solution, however, is to announce this in the Introduction to the System (cf. example 2.1 above). This would respect the CO-SITUE (4) design commitment: Make system limitations clear to users from the outset.

Many more examples of this type of problem could be quoted. Such problems are generally due to designers' lack of domain expertise and they were much more frequent in early WoZ versions of the system from whence they were gradually removed. This was the largest category of user problems in the studied material. Presumably, this result is typical of knowledge acquisition phases for artifact development.

3. (a) *The system uses different formulations of the same question to users.*
(b) New design commitment: Same formulation of the same question (or address) to users everywhere in the system's dialogue turns.
(c) Justification: Need for unambiguous system response (consistency in system task performance). The criterion is meant to reduce the possibility of communication error caused by users' understanding a new formulation of a question as constituting a different question from the questions they have encountered earlier.

(d) Examples:
3.1. "On which date will the travel begin?"; "On which date is the travel to start?" (WoZ 6).
Discovered through design analysis.

Many more examples could be quoted. They were much more frequent in early WoZ versions of the system and were gradually removed. That they were not systematically removed at an earlier stage seems to be due to the fact that the designers were not aware of them as constituting a category of its own until the final simulation.

4. (a) *There are limitations to the vocabulary material obtained through the Wizard of Oz method as a result of limitations in number and variation of the user scenarios as well as in the number of simulated dialogues.* This points to one of the limitations of the Wizard of Oz methodology itself. As such, this problem is a basic one which cannot be remedied by any known methodological extension. This raises the following problem:
(b) Problem for the application of the CO-SITUE (4) design commitment: Large enough task-related vocabulary.
(c) Justification: Need for principled limitations on natural system performance. If only part of the terms natural to users can be used in addressing the system, there is an unprincipled limitation on natural system performance, which is contrary to the CO-SITUE (4) design commitment: Intelligible, practicable and principled limitations on natural system performance.

(d) Examples:
4.1. Reservation: The system asks: "At what time of day?" (WoZ 6). There is no effective way of ensuring that we have identified all the possible phrases stating qualitative times on a given day which users might use.
Discovered through design analysis.
4.2. Information on travel times: "At what time of day?" (WoZ 6).
Discovered through design analysis.

5. (a) *Risk of lacking anticipation of users' relevant background knowledge.*
(b) New design commitment: Take users' relevant background knowledge into account.
(c) Justification: Need for adjustment of system responses to users' relevant background knowledge and inferences based thereupon. This is to prevent that the user does not understand the system's utterances or makes unpredicted remarks such as, e.g., questions of clarification, which the system cannot understand or answer.

(d) Examples:
5.1. In the system's confirmation of a reservation made, the weekday is not always mentioned in addition to the date. The user may want to know which weekday it is. This has actually been said earlier in the dialogue but it is not evident that the user will be able to remember that information.
Discovered through design analysis.
5.2. Information on 'red' (cheaper) departures on a given day. If a certain 'red' departure is fully booked it is not offered by the system. This might appear all right, but the risk is that the user knows one of the fully booked departure times as being a 'red' one and asks why it has not been offered. The system will probably not be able to understand this question. The problem was solved by inserting the phrase 'not fully booked' in the system's formulation of the information.
Discovered through design analysis.
5.3. Change of departure time: "At what date do you want to go back?" The system does not also ask for the weekday. The user may know the weekday but not the date and may ask if that will do instead, which the system will not understand. This example caused us to have a discussion on the comparative semantics of 'date' and '(week-)day' (in the Danish language) which probably couldn't have been resolved by looking into the linguistics literature. Our conclusion was the hypothesis that 'day (of the week)' is more easily understood as covering dates as well than is the covering of 'day (of the week)' by 'date'. In consequence, it was decided to use the term 'day (of the week)' throughout in the questioning of users and to use both weekday and date in informing users.
Discovered through design analysis.
5.4. Time: No distinction was made between asking users about the time (of day) and the *exact* time of day, and only the latter was used in the questioning of users. Users may know the time but not the exact time when they want to depart or arrive. The distinction was then introduced.
Discovered through design analysis.

6. (a) *'Semantical noise' in addressing users.*
(b) New design commitment: Avoid 'semantical noise' in addressing users.
(c) Justification: Need for unambiguous system response. The criterion is to reduce the possibilities of communication error caused by evoking wrong associations in users, which in their turn may cause users to make irrelevant statements which the system cannot understand.

(d) Examples:
6.1. "Are you *particularly* [stressed by the wizard] interested in making use of special fares?" (WoZ 6). The word 'particularly' was introduced in order to avoid that users who did not want to make use of special fares answered 'yes'. Experience with users had shown that this might otherwise happen. The change caused an improvement in this respect but the problem did not go away. In the discussion we came across the possibility that users interpret the question (with or without 'particularly') as the question whether they have an interest in travelling as cheaply as possible, which perhaps most people have. Several alternative design options were discussed, including:
Option a. Special fares are offered only after all the information relevant to reservation has been entered into the system's database. This will not do, however, as users who are interested in special fares may have to go through most of the reservation dialogue once again.
Option b. At an early stage in the dialogue the system asks if the user's choice of time of travel depends on the possibility of obtaining special fares. This solution was implemented.
Discovered from user problems.
6.2. In the basic task Change of Reservation: "Do you want to change time, price or destination?" Since the system presently covers only travels between Copenhagen and Aalborg, this question may cause various kinds of confusion in the user leading to irrelevant questions or remarks which the

system will not be able to understand. A solution is to replace 'destination' by 'outward journey or home journey'.
Discovered through design analysis.
6.3. To support interactive user-system problem-solving, an 'Interrupt' function had been introduced which gave access to the three functions 'Correct' (to be used, e.g., when the system had manifestly misunderstood a user request), 'Change Subject' and 'Help'. However, many users tended to interpret the 'Interrupt' function as the function 'End Now' (WoZ 5). The function was later removed (see Sect. 5 below).
Discovered from user problems.

7. (a) *The system does indeed possess the required domain knowledge but users do not have access to it or do not have access to it when they need it.*
(b) New design commitment: It should be possible for users to fully exploit the system's task domain knowledge when they need it.
(c) Justification: Risk of communication failure in case of inaccessible (or not easily accessible) task domain information. In such cases, users may pose questions which the system is unable to understand or answer.

(d) Examples:
7.1. Information on flight times: "At what time of day?" The user might be interested in being told *all* departures or arrivals on a specific day, but the system is not prepared for understanding that question. This problem has not been solved at the time of writing when the dialogue implementation work has begun.
Discovered through design analysis.
7.2. Information on prices: "Which price type do you want to be informed on?" The user does not have the possibility of asking about *all* price types at the same time. This problem has not been solved at the time of writing.
Discovered through design analysis.
7.3. Reservation: There is no link from the Reservation task to Prices for the user who, during reservation, turns out to be interested in special fares. This link was subsequently created.
Discovered through design analysis.

Many more examples could be quoted. They were much more frequent in early WoZ versions of the system and many of the identified problems have been solved. However, this type of user problem represents a major challenge in the design of current systems having limited speech recognition and natural language understanding capabilities. It seems that the only way to fully solve this class of problems is through allowing a much freer user-system dialogue than is currently possible (see also problem (11) below).

8. (a) *The possibility is not taken into account that users may make (possibly erroneous) inferences by analogy from related task domains.*
(b) New design commitment: Take into account possible (and possibly erroneous) user inferences by analogy from related task domains.
(c) Justification: Need for adjustment to users' background knowledge and inferences based thereupon. Users may otherwise fail to understand the system.

(d) Examples:
8.1. During reservation users are offered information on various types of special fare but not on stand-by fares. The reason is that stand-by tickets cannot be reserved on domestic flights. However,

the user might know that stand-by reservations can be made on international flights (within 24 hours of departure). As a result, questions might be posed which the system cannot understand. This problem has not been solved at the time of writing.
Discovered through design analysis.

This type of user problem seems to occur less frequently than most other types discussed in this paper.

9. (a) *The system provides insufficient instructions to users about how to interact with the system.*
(b) New design commitment: Provide clear and sufficient instructions to users on how to interact with the system.
(c) Justification: Risk of communication failure in case of unclear or insufficient instructions to users on how to interact with the system. Users may become (or remain) confused about the functionality of the system.

(d) Examples:
9.1. During the Introduction: Information to first-time users: "In addition, you may use the two special commands 'Repeat' and 'Correct' in order to have repeated or to correct the latest piece of information." This explanation of the functions 'Repeat' and 'Correct' is too brief to be immediately comprehensible to all users. More comprehensive instructions are now being provided.
Discovered through design analysis.

9.2. Introduction: Possibly unclear and not sufficiently emphatic request to the user to respond briefly to the system's questions one at a time in order that the system may understand the user (WoZ 5). In the earliest WoZs (WoZ 1 and 2), no instruction was given to users that they should be brief in addressing the system. The results were many violations of the CO-SITUE (4) design commitment: Use single sentences (or max. 10 words). WoZ 6 has improved the information to first-time users in this respect. They are told that they will only be understood by the system if they respond to its questions briefly and one at a time.
Discovered through design analysis. That is, users already addressed the system briefly enough but the designers believed that the final system needed a more emphatic admonition to users.

10. (a) *Non-separation between novice users who need introductory information about what the system can and cannot do and intermediate and expert users who may not or do not need such information and for whom listening to it would only retard task performance.*
(b) New design commitment: Separate whenever possible between the general needs of novice and expert users.
(c) Justification: There are major differences between the needs of novice and expert users, one such difference being that expert users already possess the information needed to understand system functionality.

(d) Examples:
10.1. Introduction: A new question to users was added: "Do you know this system?" First-time users may obtain additional information about the functionality of the system and about how to communicate with it. Other users may proceed directly with their task.
Discovered from user problems. Users complained that the system talked too much. Consideration of this complaint lead to the described design improvement (see also Sect. 5 below).

No further ways of differentiating between the needs of novice and expert users have been discovered so far.

11. (a) *The user is offered non-wanted information or ditto options which they have to respond to.*
(b) New design commitment: Avoid superfluous or redundant interactions with users (relative to their contextual needs).
(c) Justification: Need for non-superflous interaction with the system.

(d) Examples:
11.1. In WoZ (5) all users, independently of their interest in special fares, were informed on special fare options during flight time decision-making on the Reservation task. In WoZ (6) this has been avoided through the addition of a new sub-dialogue which can be side-stepped by those who do not want the special fare option.
Discovered through design analysis.
11.2. Information: Prices: A conclusion is missing to this information sub-task, which would allow users to ask for more information without returning to the main menu and go back down from there (WoZ 5). WoZ 6 has a conclusion to Information sub-tasks which allows users to immediately ask for more information. However, since information includes prices as well as travel conditions and times, an even more ideal solution would be to allow users to stay within, e.g., the time information domain if they so wish. This is currently not possible.
Discovered through design analysis.

More examples could be quoted of this type of user problem. Many of these are in fact trade-off problems which do not have simple solutions by appeal to one single design commitment. The reason is that adding more choice options to the graph has the negative effect of burdening with more dialogue turn-takings those users who do not want additional options but want to conclude the dialogue or return to other basic tasks as quickly as possible. This problem can only really be solved, it seems, by improving the language understanding capabilities (or the intelligence) of the system and in fact mirrors problem (7) above in which users cannot access information when they want to.

The current problem (11) has turned out to be much more pressing in the Information sub-task than in the other three basic tasks (Wellcome and Introduction to the system, Reservation, and Change of Reservation). The reason is that the three latter tasks, and in particular Reservation, necessarily have to involve transfer of a definite amount of information from the user to the system. Users know and accept that and field-studies have shown that, on Reservation tasks, human travel agents are almost as directive in dialogues with users as is our artifact (Dybkjaer et al. 1993). However, on the Information task users are often selectively searching for a *particular* piece of information and are not interested in anything else. Moreover, this piece of information is in some cases highly specific, in others, quite general. The more choices and dialogue turns users have to go through in order to retrieve that particular piece of information the more rigid, unintelligent and cumbersome the system comes to appear. It is worth remarking that this problem is analogous to the problem currently found in most graphical software packages which simply contain a huge list of information items on system functionality without offering any significant intelligent support for quickly retrieving a particular piece of information. The following example illustrates an (insufficient) attempt to begin to tackle this problem which can only be effectively overcome by increasing the system's natural language understanding capabilities:

11.3. At some point in the design process it was made possible for users to choose between general and specific information on prices. General information concerns prices of types of travels and clients whereas specific information concerns prices of particular travels made by particular clients. However, the Specific Information sub-dialogue is still rather complex and forces users into dialogue turns they do not need.

12. (a) *Lack of explicitness in communication of user commitments.*
(b) New design commitment: Be fully explicit in communicating to users the commitments they have made.
(c) Justification: Users need feed-back from the system on commitments made.

(d) Examples:
12.1. Confirmation: The system's description of the reservation made is not fully explicit. The exact hour and possible ‘red’/’green’ (reduced fare) conditions are missing (WoZ 5). In WoZ 6 a more explicit formulation is provided of the reservation made.
Discovered through design analysis.
The above analysis of user problems encountered during knowledge acquisition and user-system interaction design has so far led to identification of eight new design commitments with their accompanying justifications. Augmenting CO-SITUE (4) accordingly leads to the following CO-SITUE (5) representation of the design space.

CO-SITUE No. (5)

A. General constraints and criteria

Overall design goal:
- Spoken language dialogue system prototype operating via the telephone and capable of replacing a human operator;

General feasibility constraints:
- 9 m/y's available for the first version of the prototype;
- Limited machine power available;

Scientific and technological feasibility constraints:
- Limited capability of current speech and natural language processing;
- Open research questions;

Designer preferences:
- Use of the Dialogue Design Tool (DDL);

Realism criteria:
- The artifact should meet real and/or known user needs;
- The artifact should be preferable to current technological alternatives;
- The artifact should be tolerably inferior to the human it replaces, i.e., it should be at least usable (to be expanded in the usability criteria);

Functionality criteria:
- Make sure that the artifact *can do* the tasks done by the human it replaces;

Usability criteria:
- Maximize the naturalness of user-interaction with the system;
- Unless a naturalness criterion cannot be met for feasibility reasons, it should be incorporated into the artifact being designed;
- Constraints on system naturalness resulting from trade-offs with system feasibility have to be made in a principled fashion based on knowledge of users in order to be practicable by users;
- Constraints on system naturalness have to be clearly communicated to users;

B. Application of constraints and criteria to the artifact within the design space:

C = Null.
O = Null.

S = 500 words vocabulary;
Ability to repair if understanding fails;
Natural grammar;
Appropriate semantics;
Natural discourse handling;
Close-to-real-time response;
Limited speaker-independent recognition of continuous speech;
Clear and comprehensible communication of what the system can and cannot do;
Intelligible, practicable and principled limitations on natural system performance;
Make system limitations clear to users from the outset;
Provide clear and sufficient instructions to users on how to interact with the system.
Same formulation of the same question (or address) to users anywhere in the system's dialogue turns.
Be fully explicit in communicating to users the commitments they have made.

I = Spoken telephone dialogue

T = Obtain information on and perform booking of flights between two specific cities;
Use single sentences (or max. 10 words);
Understand user utterances in task domain;
Large enough task-related vocabulary;
Complete task domain information;
It should be possible for users to fully exploit the system's task domain knowledge when they need it.
Avoid superflous or redundant interactions with users (relative to their contextual needs).

U = *Take users' relevant background knowledge into account.*
Avoid 'semantical noise' in addressing users.
Take into account possible (and possibly erroneous) user inferences by analogy from related task domains.

E = Novices, walk-up-and-use users;
Intermediates;
Experts;
Separate whenever possible between the general needs of novice and expert users.

C. Hypothetical issues:
- How to accommodate experienced users?
- Is a vocabulary of 500 words sufficient to capture the sublanguage vocabulary needed in the task domain ?

D. Conventions:
See CO-SITUE (4) above.
Italics indicate new elements in CO-SITUE (n) as compared to CO-SITUE (n-1).

---

5. User Problems Causing Design Reasoning Trade-Offs

Let us proceed to describing the trade-offs found during the knowledge acquisition design phase.

5.1 Too much talk

At some stage during the WoZ simulations subjects began to complain that the system generally talked too much during its dialogue turns. This lead the designers to perform a series of trade-offs to reduce the amount of talking done by the system. In the notation above, a new design commitment was identified based on an empirically discovered user problem:

13. (a) *The system talks too much during individual dialogue turns.*
(b) New design commitment: Reduce system talk as much as possible during individual dialogue turns.
(c) Justification: Users get bored and inattentive from too much uninterrupted system talk.

Since the system already expressed itself rather succinctly on the topics it addresses, the solution does not lie in simply revising its formulations. Instead, the designers undertook a number of more or less hazardous trade-offs between the new design commitment and existing ones, as the following examples show. A simplified version of the DR/QOC notation is used for their presentation. Note that CO-SITUE design commitments here act as design *criteria* in the DR/QOC sense.

(d) Examples:
13.1. Information: Travel Conditions: The designers removed an admonition concerning prohibited luggage contents, regarding it as being 'relatively irrelevant'. The trade-off is the following:

Option 1: Remove admonition concerning prohibited luggage contents.
Option 2: Retain admonition concerning prohibited luggage contents.
Conflicting commitments involved:
(a) Reduce system talk as much as possible during individual dialogue turns.
(b) Complete task domain information.
Resolution: Option 1.
Justification: This admonition is relatively insignificant and self-evident. No third option was found which might remove the conflict between the two commitments.

13.2. Reservation: The designers removed the information to users that these would receive an invoice to be paid at the post office:

Option 1: Remove the information to users that they will receive an invoice to be paid at the post office.
Option 2: Retain the information to users that they will receive an invoice to be paid at the post office.
Conflicting commitments involved:
(a) Reduce system talk as much as possible during individual dialogue turns.
(b) Complete task domain information.
Resolution: Option 1.
Justification: This information is irrelevant to business customers who have an already agreed-upon method of payment. Other customers would ‘naturally’ expect the method of payment described. This justification was not found to be convincing. As a result, a third option was invented which resolves the conflict between the commitments above:
Option 3: When customers obtain their customer number (from a human travel agent), the method of payment is agreed upon.

13.3. Information: The designers decided to split the information on special fares and their conditions between the Price sub-task and the Travel Conditions sub-task. As a result, customers would have to carry out two different Information sub-tasks to obtain full information on special fares:

Option 1: Provide full information on special fares under the two different sub-tasks Prices and Travel Conditions.
Option 2: Provide different parts of the information on special fares under the two different sub-tasks Prices and Travel Conditions.
Conflicting commitments involved:
(a) Reduce system talk as much as possible during individual dialogue turns.
(b) It should be possible for users to fully exploit the system's task domain knowledge when they need it.
Resolution: Option 2.
Justification: Criterion (a) was assigned the stronger weight. This was not convincing, however, and therefore a third option was found which might resolve the conflicting commitments:
Option 3: Introduce a direct link between Prices and Travel Conditions via an additional dialogue turn. The system would ask: "Do you want to be informed on the corresponding Prices/Travel Conditions?" depending on what sub-task the user is currently engaged in.

13.4. Information: Check-In Time: The designers decided to only inform about special check-in times for travellers who accompany infants or are in need of travel personnel support. This information was not also provided under Information: Travel Conditions: Infants, and under Information: Travel Conditions: People in need of travel personnel support. As a result, customers would have to carry out two different Information sub-tasks to obtain full information:

Option 1: Provide full information on check-in times under the two different sub-tasks Information: Travel Conditions: Infants, and Information: Travel Conditions: People in need of travel personnel support.
Option 2: Provide different parts of the information on check-in times under the two different sub-tasks (1) Information: Check-In Time and (2) Information: Travel Conditions: Infants, and Information: Travel Conditions: People in need of travel personnel support.
Conflicting commitments involved:
(a) Reduce system talk as much as possible during individual dialogue turns.
(b) It should be possible for users to fully exploit the system's task domain knowledge when they need it.
Resolution: Option 2.
Justification: Commitment (a) was assigned the stronger weight. No third option was found which might remove the conflict between the commitments.

In applying the new design commitment on avoidance of too much system talk, the designers were led to reconsider a number of turn-taking sequences, trying to remove some of their elements:

13.5. Change of Reservation: An extra system question was removed which made sure that the user really wanted to cancel a reservation:

Option 1: Provide feed-back to users that a cancellation is about to be done and ask them to confirm their choice.
Option 2: Make cancellation when this has been asked for by the user.
Conflicting commitments involved:
(a) Reduce system talk as much as possible during individual dialogue turns.
(b) Avoid superflous or redundant interactions with users (relative to their contextual needs).
(c) Ask confirmation from users before important commitments are entered into the database (new candidate design commitment).

Resolution: Option 2.
Justification: Commitments (a) and (b) were assigned the stronger weight. However, a third option was found which resolves the conflict between the actual or possible commitments (a) to (c). A general 'Correct' function was introduced which always allows users to backtrack one step in the dialogue in order to make changes to the decisions made (see below).

13.6. Reservation and Information: Users were no longer asked if they knew the Flight Number as an alternative to their knowing the departure or arrival time:

Option 1: Ask users about the Flight Number as an alternative to the departure or arrival time.
Option 2: Do not ask users about the Flight Number as an alternative to the departure or arrival time.
Conflicting commitments involved:
(a) Reduce system talk as much as possible during individual dialogue turns.
(b) Avoid superflous or redundant interactions with users (relative to their contextual needs).
(c) Take users' relevant background knowledge into account.
Resolution: Option 2.
Justification: It is very uncommon for users to know about Flight Numbers. No third option was found which might remove the conflict between the commitments.

5.2 Prices of reserved tickets

Users could obtain full price information through the Information task. However, they might want to know the price of their tickets as part of the Reservation task. It should be remembered here that customers may make reservations for several persons at a time, for different itineraries for each person and of different price types. Many customers (mostly the professional travellers) are not interested in such price information.

Option 1: Provide full price information at the end of a Reservation task.
Option 2: Users who want to know the price of their reserved tickets have to compute the full price information themselves by performing the Information task and repeat most the information which has already been given to the system during reservation.
Conflicting commitments involved:
(a) Avoid superflous or redundant interactions with users (relative to their contextual needs).
(b) It should be possible for users to fully exploit the system's task domain knowledge when they need it.
Resolution: This is a direct clash between two different design commitments because of the existence of different needs in the user population. A third option was identified and selected:
Option 3: Always inform users about the total price of their reservation (but not its breakdown into the prices of individual tickets).
Justification: Compromise between the two relevant design commitments. Professional users loose time on an extra dialogue turn. Users wanting the price information have an important part of what they want when they want it.

5.3 Interactive user support functions

Ideally, interactive user support functions should allow users to obtain contextually relevant help at any time, move freely around within the task domain covered by the system, have system phrases repeated at will and change whatever commitments they have been making. Unfortunately, not all of this is possible given current scientific and technological constraints on spoken language dialogue systems. During the early Wizard of Oz generations the user-system dialogue was relatively free and natural and the wizard was able to offer an almost ideal interactive user support. Accordingly, the first attempt to add a set of interactive user support functions came close to the ideal just described. Users were offered an ‘Interrupt’ function (which they sometimes misinterpreted, cf. Sect. 4 above) subsuming the following support functions: ‘Correct’, ‘Change Subject’ and ‘Help’. However, the ‘Help’ function never came close to being operationally specified and was subsequently abandoned. The ‘Change Subject’ function, being context-independent, turned out not to be implementable in the DDL tool to be used for dialogue structure implementation (cf. CO-SITUE (5)).

Adjusting to technological constraints and designer preferences, the ‘Interrupt’ function was replaced by the two rather basic, context-dependent support functions ‘Repeat’ and ‘Correct’ (the latter having been part of the ‘Interrupt’ function) which are implementable in DDL. ‘Repeat’ makes the system repeat its latest utterance (dialogue turn) if, e.g., users believe not to have heard or understood it correctly. ‘Correct’ enables users to repeat their own latest utterance if, e.g., this turns out to have been misrecognised or misunderstood by the system. A third support function, ‘Local Help’ is equally feasible in DDL but will not be implemented in the first prototype for general feasibility (resource) reasons. Local Help, being context-dependent, would make the system explain its latest utterance to users by using different terms.

The logic of this situation may be represented as follows:

Design problem: How to provide natural interactive user support during user-system dialogue? This problem has a high-level justification in the general *design commitment:* unless a naturalness criterion cannot be met for feasibility reasons, it should be incorporated into the artifact being designed (cf. CO-SITUE (5)). The *options* for solving the design problem are generally (and drastically) reduced to strictly context-dependent ones by the *designer preference* to use the Dialogue Design Tool (DDL, cf. CO-SITUE (5)). This forced the designers to generate options conforming to DDL. They came up with the following:

Option 1: ‘Repeat’ which meets the design commitment: ability to repair if understanding fails (cf. CO-SITUE (5)).
Option 2: ‘Correct’ which meets the design commitment: ability to repair if understanding fails (cf. CO-SITUE (5)).
Option 3 ‘Local Help’ which could meet the design commitment: ability to repair if understanding fails (cf. CO-SITUE (5)) but which will not be implemented in the first prototype due to general feasibility (resource) constraints.

5.4 The number problem

The number problem arises in part from the fact that current speech recognition systems have no possibility of recognising the multitude of different names that persons have. Person individuation is crucial to billing, issuing of individual tickets, security and emergency management. People or institutions should be billed for the air travels made. Tickets are issued in the names of the individuals who will use them. And even if a traveller does not need a ticket, such as an infant accompanied by an adult, their identities should be registered somewhere when they go on a flight. Finally, individual tickets should be registered as such, i.a. because one and the same person may have booked several flights in the database. Also, flights have numbers, but these are only used internally by the system (cf. above). Even though current speech recognition systems cannot handle spoken names, they can handle spoken numbers. They can also handle spelling. However, the number of numerals is smaller than the number of letters in the alphabet, and persons can be individuated by a smaller number of numerals than the number of letters they usually have in their full names. Accordingly, the number problem was solved as follows:

A *customer number* is used for mailing bills. When given the number, the system retrieves the relevant name and address from the database. This number is assigned to individuals or institutions through their contacting a travel agent.
An *id-number* is used to identify an individual traveller by name and age. Every traveller has to have such a number. When given the number, the system retrieves the relevant name and age from the database. The number is assigned to individuals through their contacting a travel agent.
A *reference number* is used to identify each individual ticket which has been reserved. This number is relevant when customers want to change their reservations.

This numbering system is sufficient in the task domain. However, it has caused several design problems:

Number problem 1: Why should singles, in contrast to, e.g., companies and heads of families, have both a customer number and an id-number? There is no clear reason for that but the problem has not been solved. The straightforward solution is to have the travel agent inquire if a customer is single and if that person would be happy having only an id-number. This would make the system conform to the *design commitment:* Avoid superflous or redundant interactions with users (relative to their contextual needs), without any apparent trade-offs having to be made. However, the designers seem to have put more emphasis on a design commitment which has nothing to do with usability, namely, the simplicity and consistency of the numbering system.

Number problem 2: Users are not being told during the system's introduction to first-time users that they need (one or) two numbers in order to make a reservation. Instead, they are asked for their numbers after three dialogue turns on the Reservation task. If they do not have these numbers, they have wasted their time on the system and have to call a travel agent. The straightforward solution is to include information on number requirements in the system's introduction to first-time users. This would make the system conform to the *design commitment::* Avoid superflous or redundant interactions with users (relative to their contextual needs). However, the designers seem to have put more emphasis on the *design commitment::* Reduce system talk as much as possible during individual dialogue turns, supported by the questionable assumption that the number of users who would meet (only once!) with this problem will be relatively small. In addition, whereas the system's request for an id-number was originally put to the user at the start of the Reservation task, this request was subsequently moved till later in the Reservation task for a reason which has nothing to do with usability, namely, that this request 'naturally' belonged to the Persons sub-task which was located three dialogue turns down into the Reservation task.

Number problem 3: If, during the Change of Reservation task a customer cannot remember the relevant numbers or dates, what happens? Or, which number or numbers should the customer be able to remember in order to be allowed to change a reservation? At the moment, the customer is being refused if the only number (including the date of departure) remembered is the customer number. This is not a technological problem, but the relevant design commitments are presently unclear and no satisfactory solution has been found.

The final CO-SITUE (6) representation of the design process prior to implementation of the first prototype is shown below. It turned out that only one new design commitment with accompanying justification was necessary to handle the trade-offs discussed in this section. Note that the CO-SITUE frame now refers to accompanying documentation for presentation of the final user-system interaction graph structure which is being implemented as the first spoken language dialogue system prototype (Dybkjaer and Dybkjaer 1993). In addition, answers have been included to the hypothetical issues contained in section C of the frame. Again, part of this has been documented elsewhere (Dybkjaer and Dybkjaer 1993). CO-SITUE (6) has been augmented with a full explanation of its notation to make it a self-contained representation of the usability engineering aspects of the design process.

CO-SITUE No. (6)

A. General constraints and criteria

Overall design goal:
- Spoken language dialogue system prototype operating via the telephone and capable of replacing a human operator;

General feasibility constraints:
- 9 m/y's available for the first version of the prototype;
- Limited machine power available;

Scientific and technological feasibility constraints:
- Limited capability of current speech and natural language processing;
- Open research questions;

Designer preferences:
- Use of the Dialogue Design Tool (DDL);

Realism criteria:
- The artifact should meet real and/or known user needs;
- The artifact should be preferable to current technological alternatives;
- The artifact should be tolerably inferior to the human it replaces, i.e., it should be at least usable (to be expanded in the usability criteria);

Functionality criteria:
- Make sure that the artifact *can do* the tasks done by the human it replaces;

Usability criteria:
- Maximize the naturalness of user-interaction with the system;
- Unless a naturalness criterion cannot be met for feasibility reasons, it should be incorporated into the artifact being designed;
- Constraints on system naturalness resulting from trade-offs with system feasibility have to be made in a principled fashion based on knowledge of users in order to be practicable by users;
- Constraints on system naturalness have to be clearly communicated to users;

B. Application of constraints and criteria to the artifact within the design space:

C = Null
O = Null

S = 500 words vocabulary;
Ability to repair if understanding fails;
Natural grammar;
Appropriate semantics;
Natural discourse handling;
Close-to-real-time response;
Limited speaker-independent recognition of continuous speech;
Clear and comprehensible communication of what the system can and cannot do;

Intelligible, practicable and principled limitations on natural system performance;
Make system limitations clear to users from the outset;
Provide clear and sufficient instructions to users on how to interact with the system.
Same formulation of the same question (or address) to users anywhere in the system's dialogue turns.
Be fully explicit in communicating to users the commitments they have made.
Reduce system talk as much as possible during individual dialogue turns.

I = Spoken telephone dialogue

T = Obtain information on and perform booking of flights between two specific cities;
Use single sentences (or max. 10 words);
Understand user utterances in task domain;
Large enough task-related vocabulary;
Complete task domain information;
It should be possible for users to fully exploit the system's task domain knowledge when they need it.
Avoid superflous or redundant interactions with users (relative to their contextual needs).
See final user-system interaction graph set in accompanying documentation;

U: Take users' relevant background knowledge into account.
Avoid 'semantical noise' in addressing users.
Take into account possible (and possibly erroneous) user inferences by analogy from related task domains.

E = Novices, walk-up-and-use users;
Intermediates;
Experts;
Separate whenever possible between the needs of novice and expert users.

C. Hypothetical issues:
- How to accommodate experienced users? *Some modest measures have been taken.*
- Is a vocabulary of 500 words sufficient to capture the sublanguage vocabulary needed in the task domain ? *Yes. See accompanying documentation;*

D. Conventions:
C = Collaborative aspects.
O = Organisational aspects.
S = System aspects.
I = Interface (or more generally: system Image) aspects.
T = Task aspects including task domain aspects.
U = User aspects.
E = User experience aspects.

CO-SITUE No. (n) indicates the number of the current CO-SITUE specification.
"Null" means that the artifact does not embody a certain aspect of CO-SITUE.
Italics indicate new elements in CO-SITUE (n) as compared to CO-SITUE (n-1).
Note that, in the present design process, the *C* and *O* aspects have been temporarily set to null during design of the first prototype and will be reconsidered during design of the second prototype.

---

6. Discussion

Despite the fact that it describes a relatively significant design effort, the final CO-SITUE (6) representation remains relatively compact. It has turned out that the CO-SITUE representation needs accompanying documentation to describe the user-system interactional structure which has resulted from the knowledge acquisition design phase and to document that one basic, hypothetical design question can be answered in the affirmative. However, even when we add this background documentation the representation remains rather compact. It should be emphasized that compactness has been achieved partly by artificially reducing the usability engineering aspects of the design problem since no collaborative and organisational aspects of the artifact have been explicitly taken into account. Their representation would no doubt further complicate the CO-SITUE design space representation. Nevertheless, the observed compactness of CO-SITUE design space representation opens at least three possibilities of application:

1. Designers might in fact be interested in and capable of using the CO-SITUE frame notation in design practice because of its compactness, both during the design process and afterwards when the artifact has to be maintained, augmented or revised. This is a topic for future work.

2. Designers working on *similar* design projects in the field of spoken language dialogue systems might directly benefit from the detailed contents of the CO-SITUE representation. They might, of course, be interested in any form of documentation from similar design projects. The CO-SITUE representation in particular, however, constitutes a comprehensive representation of the usability issues which were dealt with in a spoken language dialogue system design process and might serve as a useful starting-point or point of comparison for similar efforts leading to a set of enhanced and/or diversified representations of the usability issues which are generally involved in the design of this type of artifact.

3. Apart from the potential usefulness of the CO-SITUE frame notation itself, use of filled CO-SITUE notational frames might be of interest to designers working on artifacts quite *dissimilar* to spoken language dialogue systems. Anyone looking at the specific design criteria, commitments and constraints which are represented in CO-SITUE (6) cannot avoid noticing that many of these are in fact ubiquitous in information systems design. One might envision that a limited series of completed CO-SITUE representations from different domains of information systems design might serve as a basis for 'intuitive generalisations' to other domains and artifacts, thus providing an economical means for enhancing designers' pool of hands-on experience. Again, this is a topic for future work.

It has been interesting to observe that most user problem design issues could be solved by reference to one single design commitment and that the number of design commitments needed to resolve the user problems identified and solved during the particular design process described above was found to be rather limited. This is no doubt due to the level of generality at which these commitments have been expressed. The hypothesis is that this level of generality is appropriate to usability engineering problems in realistic design processes. However, if the design commitments are expressed in rather general terms and the particular examples of user problems met with during a design process are quite specific, there is an important issue of how to apply the design commitments to the resolution of specific examples of user problems. From a CO-SITUE point of view, making the design commitments expressed in the frame notation so specific as to directly address each individual user problem instance and in this way overcome the application problem just mentioned, is no viable option. We would literally get lost in large numbers of quite specific design commitments and hence loose most of the potential for generalisation of CO-SITUE mentioned above. The objective of CO-SITUE development is to effectively inform design processes of usability issues, constraints and criteria. This, it would seem, can be done only at some suitable level of abstraction.

As we have seen, trade-off problems need a more complex analysis and representation than the straightforward application of one design commitment to some user problem. A simplified DR/QOC notation was found useful for representing trade-off design reasoning. It is worth noting also that this form of trade-off representation in some cases gave rise to ideas for changes to the artifact being designed. Whether the problems left unsolved above might be satisfactorily solved through further design space analysis remains, of course, an open question. It is no doubt possible to treat a number of the usability problems which were solved by reference to a single design commitment as constituting ‘hidden’ trade-offs. However, strong reasons have yet to be found for producing a more detailed design space representation of those problems than the one used above.

It is striking that rather few of the actual or potential user problems identified above were discovered empirically through observation of actual user problems during the Wizard of Oz simulations. One possible explanation is that the recorded and transcribed material generated by the simulations had not been sufficiently scrutinized during the design process because of resource constraints, which was in fact the case. However, given that the Wizard of Oz simulation technique, as used by us, is supposed to be an aid to design rather than a methodology for doing more exhaustive experimental psychological investigation, this state of affairs may reflect how the simulation technique can actually be expected to be used during realistic design processes. If this is true, then the Wizard of Oz technique, in addition to being a method for gathering sub-language material (on domain-specific vocabulary, grammar, semantics, discourse phenomena and discourse structure) which could not be gathered otherwise, may be as much a technique for allowing designers to reflect on their design as a method for the empirical discovery of user problems. It should be added, however, that the building of intelligent tools for processing (i.e., indexing, filtering, searching, etc.) the large amount of data collected during WoZ simulations might facilitate the timely use of this data during the design process (Salber and Coutaz 1993).

In conclusion, let us briefly reflect on the usability engineering problem of HCI as it appears from the point of view of the CO-SITUE approach reported in this paper. Three levels of addressing the usability engineering of IT artifacts seem clearly distinguishable from the bottom up. The higher we move upwards in this hierarchy of levels, the more unsolved problems we find. At *Level 1* we find the actual design process in which a variety of usability engineering approaches fit naturally such as those mentioned in Sect. 1 above. At *Level 2* we find the rapidly developing area of Design Rationale approaches to design space analysis as well as approaches to design space representation such as CO-SITUE. These approaches aim at making explicit design reasoning and design commitments in a way which is systematic and natural to use by designers. The approaches to design space analysis and representation at Level 2 provide, it may be rather safely argued, a necessary bridge to *Level 3* which contains science-based approaches to usability engineering such as those mentioned in Sect. 1 above as well as many others. Common to the approaches of Level 3 are that they seek to provide analytic and theoretically based leverage on the discovery and solution of design problems to do with usability.

References

Barnard, P. *Technical Annex* for Esprit Basic Research Project 7040, AMODEUS-2, 1992.

Barnard, P. and May, J.: Cognitive modelling for user requirements. In Byerley, P.F., Barnard, P.J. and May, J. (Eds.): *Computers, Communication and Usability: Design issues, research and methods for integrated services.* Amsterdam, North-Holland, 1993.

Bernsen 1993a: Bernsen, N.O.: Structuring design spaces. *INTERCHI '93 Adjunct Proceedings*. 211-12, 1993.

Bernsen 1993b: Bernsen, N.O.: The structure of the design space. In Byerley, P.F., Barnard, P.J. and May, J. (Eds.): *Computers, Communication and Usability: Design issues, research and methods for integrated services.* Amsterdam, North-Holland, 1993, 221-244.

Bernsen 1993c: Bernsen, N.O.: Design of a Spoken Language Dialogue System. A Study of the Initial Specification Phase. *Esprit Basic Research Action AMODEUS Working Paper* RP3-WP3,1993.

Carroll, J.M. and Rosson, M.B.: Getting around the task-artifact cycle: How to make claims and design by scenario. *ACM Transactions on Information Systems,* Vol. 10, No. 2, April 1992, 181-212.

Salber, D. and Coutaz, J.: A Wizard of Oz platform for the study of multimodal systems. *INTERCHI '93 Adjunct Proceedings* 95-96, 1993.

Dybkjaer, H. and Dybkjaer, L.: *Wizard of Oz Experiments in the Development of the Dialogue Model for P1*. Report 3, Spoken Language Dialogue Systems. STC Aalborg University, CLT Copenhagen University and CCI Roskilde University, February 1993.

Dybkjaer, H., Bernsen, N.O. and Dybkjaer, L.: Wizard-of-Oz and the trade-off between naturalness and recogniser constraints. *Proceedings of EUROSPEECH '93*, Berlin, September 1993, 947-50.

Lewis, C., Polson, P., Rieman, J. & Wharton, C.: Testing a walkthrough methodology for theory-based design of walk-up-and-use interfaces. *Proceedings of the CHI '90 Conference on Human Factors in Computing Systems*, pp. 235-42, 1990. New York: ACM.

MacLean, A., Young, R., Bellotti, V. and Moran, T.P.: Questions, Options, and Criteria: Elements of Design Space Analysis. *Human-Computer Interaction* Vol. 6, 1991, pp. 201-50.

MacLean, A., Bellotti, V. and Shum, S.: Developing the design space. In Byerley, P.F., Barnard, P.J. and May, J. (Eds.): *Computers, Communication and Usability: Design issues, research and methods for integrated services.* Amsterdam, North-Holland, 1993, 197-220.

Acknowledgements. I am grateful to Hans Dybkjaer, Laila Dybkjaer and Tove Klausen for helpful comments on earlier drafts. The work was done in part on a grant from the Danish Research Council for the Technical Sciences on 'Spoken Language Dialogue Systems' and in part under AMODEUS II, Esprit Basic Research Action 7040. The support is gratefully acknowledged.

List of publications in the series "Working Papers in Cognitive Science"

1991

91-1 *Kognitionsforskning ved AUC*. Per Hasle.
91-2 *Johnson-Laird's Theory of Mental Models*. Niels Ole Bernsen.
91-3 *Potential Flashbulbs: Memories of Ordinary News as the Baseline*. Steen F. Larsen.
91-4 *"Icons Make me Click" - The Problem of Semantics*. Ib Ulbæk.
91-5 *Mental Models in Human-Computer Interaction*. Niels Ole Bernsen.
91-6 *Two Games in Town. Systematicity in Distributed Connectionist Systems*. Niels Ole Bernsen and Ib Ulbæk.
91-7 *Bevidsthedsfilosofi og kognitionsforskning*. Ib Ulbæk.

1992

92-1 *Matching Information and Interface Modalities. An Example Study*. Niels Ole Bernsen.
92-2 *Systematicity, Thought and Attention in a Distributed Connectionist System*. Niels Ole Bernsen and Ib Ulbæk.
92-3 *Communication Failure and Mental Models*. Brynja Svane and Niels Ole Bernsen.
92-4 *Kognitiv semantik og diagrammatisk tænken: fra intuition til formalisering*. Ph.D. Thesis. Michael May.
92-5 *Design of a Spoken Language Dialogue System. A Study of the Initial Specification Phase*. Niels Ole Bernsen.
92-6 *Learning the time constants of a Dynamic Task*. Bernd Brehmer, Leif Løvborg, and Anders Winman.
92-7 *Cost Information in a Dynamic Decision Task*. Bernd Brehmer, Leif Løvborg, and Anders Winman.
92-8 *New Facilities in NEWFIRE*. Leif Løvborg and Bernd Brehmer.
92-9 *C.S. Peirce on the Possibility of a Temporal Logic*. Peter Øhrstrøm.
92-10 *Hypertid*. Peter Bøgh Andersen og Peter Øhrstrøm.

1993

93-1 *A logical approach to handling uncertainty*. Morten Elvang-Gøransson, Paul Krause, John Fox.
93-3 *Den kognitive revolution*. Niels Ole Bernsen.
93-4 *Systematicity in the vision to language chain*. Niels Ole Bernsen.
93-5 *The structure of the design space*. Niels Ole Bernsen.
93-6 *Errors in a team context.* James Reason and Niels Ole Bernsen.
93-7 *Talking to a Wizard: report from the design of a natural speech understanding system*. Tove Klausen.
93-8 *Eksempler på samspillet mellem simultane og sekventielle kognitive processer.* John Paulin Hansen.
93-9 *Foundations of Multimodal Representations. A Taxonomy of Pure Modalities*. Niels Ole Bernsen.
93-10 *A methodology for mapping information from task domains to interactive modalities*. Niels Ole Bernsen.
93-11 *A research agenda for modality theory.* Niels Ole Bernsen.

93-12 *Types of user probems in design.* Niels Ole Bernsen.