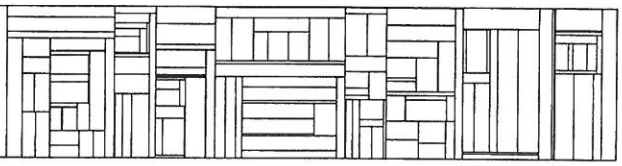
ISSN 0105-8517

User Support: From Surface to Structure

Riitta Hellman

DAIMI PB – 263
October 1988

AARHUS UNIVERSITY
COMPUTER SCIENCE DEPARTMENT
Ny Munkegade 116 – DK 8000 Aarhus C – DENMARK
Telephone: + 45 6 12 71 88 *Telex: 64767 aausci dk*



Abstract

It is not unusual that a user gets lost when using a computer system. Supporting users in difficult situations is therefor very important independently of whether they are working with personal applications or with an integrated information system. User support can be approached in many ways. One possible way is to develop help systems for assisting the users. I claim, however, that most current approaches to help systems tend to focus on plain data and information processing issues leaving the organizational context in which the use takes place without attention. It is asserted in this paper that it is highly relevant for the users to experience the organizational context and that it is possible to reflect the context in a help system. An approach to representing the context in connection with a help system is made by introducing the idea of a context database. A context database is parallel to the actual database and it contains information about task flows, task-connected information objects, and the like. (The information objects themselves are stored in the actual database.) In order to be able to realize a help system which takes the organizational context into account the analysis of work and information systems has to focus on information processing tasks, information objects, and other entities which are needed for the context database. A proposal for how to perform such analysis enterprises is presented. The proposal introduces five groups of objects of analysis: tasks, organization of work, job design, computer applications, and information media.

1. THE LACK OF OVERALL VIEW

End users who get lost when using a computer system are apparently common. An indication of this is the boom of the development of different help systems, the intensive research of the use of natural language, modelling the behavior of the users, and the like. It is, however, asserted in this paper that *the* problem among end users is the lack of overall view. By this I refer to the difficulty of perceiving the organizational context of (information processing) tasks. The (eventual need to and) tradition of building integrated ISs, where everything is directly or indirectly connected to everything else, apparently contributes to the disorientation of the users. The lower the level of a user in the organizational hierarchy is, the more difficult it is for her to understand the origin of the transactions which influence her job, and it is equally problematic to know what kind of consequences her actions will have. Parts of the explanation are the complex IS itself, the inadequate training that the users often have received, and the poor support that is provided in a concrete use situation. Persons working at higher levels of an organizational hierarchy, on the contrary, usually possess understanding of the organization's functioning and transaction flow. Information systems requiring such understanding on one hand and supporting it on the other hand are common at executive and management levels. Decision support systems can be mentioned as examples of such systems. (Decision support systems are not referred to as ISs of some ideal type!) As personal computing has become more and more popular, integrating personal systems to the information infrastructure of the whole organization has become a topical issue, too. In a situation like this, supporting the user so that the context of the use situation is perceivable for her, is undoubtedly of great importance as well as it is challenging.

These assertions and observations originate from the findings of the Knowledge and Work project. The project experimented with a method for supporting end users' comprehensive learning (Eriksson and others, 1987), and the main focus was on the organizational context of the use of the computer. It was found out that knowledge about the organizational context is at least as important as the ability to use the apparatus itself. The (in this context) relevant parts of our research program in user organizations were:

I The current state of an IS — its structure and functioning — was analyzed by studying available systems documentation and executing test runs of the application programs. Informal material, like letters and notes, was also used. Additional information was acquired by discussing with the edp-personnel.

II Analysis of, e.g., job contents, organizational communication, task coordination, and practices of collaboration was carried out by conducting semi-structured theme interviews of the users, managers and edp-personnel. The outcome of the interviews was complemented by observing the end users at work.

III A learning program emphasizing the organizational context of the IS was carried out, and the changes concerning the users' ability to utilize and understand the IS were recorded. This phase complemented the analysis performed during the earlier phases, too. Changes in the use situation were thus provoked without manipulating the technical system.

The importance of supporting users by means of, e.g., help systems is apparently generally accepted. It is claimed in this paper that most help systems do not, however, provide adequate support since the organizational context of task performance and computer use is not reflected in the design of these systems. On the contrary, current help systems *actually* draw on a quite narrow, technically oriented understanding of the use of computers. Two examples of this are given below.

Supporting end users' orientation in a work situation may be built upon different sophistication levels of the user. The following learning stages (Mozeiko, 1982) can be used as the criteria: learning the basics of the systems, using the system to obtain desired results with minimal prerequisite knowledge, progressing to more independent use of the system, probing into the more subtle or difficult features, and producing quality results within known system constraints. The task context in which the use at different levels takes place seem not to be reflected at all. Norman (1984), in turn, has described four levels of computer literacy, which can be regarded as *common apprehensions* of what is central: it is important to understand the general principles of computation, how to use computers, how to program computers, and the science of computation. In my opinion it is not farfetched to claim that many implemented ISs, and the support that is offered to the users, exhibit these principles in one form or another. The users' disorientation may, however, result from this kind of underlying ideas of what computers and computing are all about — principles which are more or less characteristic to the way of thinking of computer specialists, but which without question have negative consequences to the the computing reality in organizations.

Totally opposite types of approaches to the coping of the users can be observed, too. Discussions, where for example shop floor workers are described as end users being capable of managing complicated (use) situations by means of homemade solutions, informal organization, etc., are common. Nevertheless, these approaches contain certain restrictions. I do not criticize those authors who point out that the shop floor workers are *not unskilled*, but possess many skills and qualifications which are possible only after long experience, accumulated knowledge, and thorough understanding (see, e.g., Kusterer, 1978). It is just not fruitful to admire the development of tacit skills or elegant survival strategies, if as a consequence the fact that these users could have a more sophisticated information system *to begin with*, is ignored. Instead, I claim that a considerable amount of "elegance discrimination" between the information (sub)systems of, e.g., the

managerial level and the operational level is often built in ISs, and far too little attention is paid to redefining the reality. Making it possible for all users — independently of the the organizational level — in a use situation, e.g., to see 'what has happened' and 'what is going to happen', is essential (cf. decision support systems).

In order to tackle the problems of confused end users a new perspective to help system design is approached. The main objective of this paper is to propose a more profound concept of *context sensitivity* in the area of user support than currently is the situation. By context sensitivity I mean qualities which reveal the organizational context of tasks. This type of context can be represented by relations mainly between tasks, tasks and information objects,and information objects and tasks. Sensitivity also refers to different levels of detail that may be visualized in connection with inquiries to a context sensitive support facility. In addition, no user group-connected restrictions or privileges are associated with the concept of such a support system. This means that that the tasks and task connected information objects of all users are included. I am confident about the importance of the idea of context sensitive support systems on one hand, and the feasibility of such systems on the other hand.

It is suggested in this paper that the realization of a context sensitive support system could be based on a *context database*. This database contains information about the task flows of the users in terms of task interconnections, the relations between information objects and information processing tasks, and the relations of various information objects. (The information objects themselves are naturally stored in the actual database to which the context database is parallel.) Combining these two types of databases functionally with each other lays the basis for the operation of the context sensitive support system. The idea of the context database is later discussed in more detail.

The discussion goes as follows: In the next section some representative approaches to user support are discussed in the light of examples. It is also asserted that the majority of the current help systems do not employ the concept of organizational context. In the section 'Building contextuality' a proposal for a conceptual framework for realizing a context sensitive support system is presented. A structured approach to analyzing information and work systems in respect to the development of context databases is introduced in section 'Structuring the analysis'. Finally, the implications of context sensitive support systems are discussed.

2. USER SUPPORT PRACTICES REVISITED

The rules of data processing are usually invisible for the user, although the degree of visibility varies according to the programming languages and tools that have been used and the designs that are realized. The most static type of application is a conventional transaction processing system where the application has been implemented by using a traditional programming language. Here only the input and output ends of the data processes (cf. preparation of input data and printing out reports) are concrete for the user, and she usually is neither able nor allowed to change the system. Generally spoken, this is often the worst case from the end user's point of view as far as system perceivability is concerned. Database systems represent a step towards more understandable ISs. Now the way of storing and processing information, as well as the schema according to which this is done, are relatively "graspable". A majority of database operations is usually preprogrammed, but modern database systems and query languages also allow the user to create new or modify existing operations. When this is the case, the user *has to* know/understand what she is doing/causing. Application generators can be considered in this context, too (see, e.g., Martin, 1982). There are usually several more options for the users to develop the applications, but by and large, these fall into the same category as database systems.

Both database systems and systems created by using application generators may be quite powerful. There also exists other possibilities to implement ISs, although the degree of "momentum" of the tools and hence the applications varies. Examples of smaller systems are spread sheet applications and file systems where the application is composed in a card index manner. Here the basic idea is that the user herself is able to define, realize, and maintain her own applications, although it is not always as simple as generally assumed: among other things, Brown (1987) observed that the biggest problems concerned composing the formulas which actually *represent* (parts of) the information infrastructure of the organization. This may be an indication of the lack of overall view in the use situation. The results of the study of Canter and others (1986) may be interpreted in the same manner. Here the users were tailoring a database system. Preprogrammed alternatives (menus), where understanding the interconnections between pieces of information was not required, were clearly preferred by a majority of the users. It is also becoming possible to integrate different tools with each other. In this sense a single user can have a connection to larger organizational information processes. Now the user support issue is approached. As Lehman (1985) states, the juxtaposition of personal and organizational information processing should not be regarded as, e.g., a problem of choosing hardware, but rather as issues concerning the context of use. In the following, current help approaches are analyzed in terms of their support to the context of use.

In connection to different types of application environments, different styles of user support practices and underlying assumptions can be observed. In each type of environment, several levels of help facilities may be available. Supporting end users' orientation in a use situation can for example be seen as a problem of knowing (Nievergelt and Weydert, 1980) what data are accessible, which commands are available, and what the trail has been, or at an even more detailed level (Riley, 1986) as a question of having knowledge about the action structure of a command, the syntactic structure of a command, how a command works, and objects like files, programs, and buffers. These are good examples of how to concentrate on the technical aspects of ISs. Help functions which focus on data and command accessibility provide local and/or situational "enlightenment", but this is done free from a larger context.

An approach towards manageable use environments is also bringing menu networks under control by laying an orderly transition design (see, e.g., Brown, 1982), and possibly making this network visible to the users. Trail history and transition network commands show the user a layer on the top of the IS. This layer is often so global that it necessarily becomes very thin. Hence the organizational context of the current tasks is very poorly supported. It can of course be claimed that data is needed for performing job tasks, commands are necessary to initiate data processing, the trail is also a task trail, and a menu network is a network of tasks and task relationships. This is, however, a relatively artificial interpretation, and has not been the original idea with this kind of help functions.

Other guidelines (Christie, 1985) which are seen as supportive to the user's work situation suggest that the system should, e.g., make use of recognition, capitalize upon the facility humans have for forming incidental associations (incidental learning) by encouraging processing of items at the same time the user puts them into the electronic store, encourage the user to process items to be filed deeply rather than shallowly, encourage elaborate processing at the time the user files an item away, form many incidental associations that can be used to find items later, be capable of using many different sorts of prompts, such as color, shape, size, type, font, orientation, and texture, offer help for reconstructing the context in which the item was filed, and not require the user to recall information (such as an exact file name). This approach focuses more on the work situation. Task contents and task flow are analyzed, too. The emphasis is, however, on a *particular user*. These issues are often referred to as dealing with human factors because of the use of user psychology. Psychological aspects are experienced as important, since developing for example intelligent advice-giving systems requires at least following types of knowledge built in an IS (Carroll and McKendree, 1987): knowing how advice should be given to people in general and in particular, knowing the tasks that are to be supported by giving advice, and knowing what kind of a person gets and uses the advice.

Focus on visuality and service can also be perceived as user support. A typical example is the Apple Macintosh interface, which is functionally consistent in most Macintosh applications. In addition to the way things are represented — icons, pull-down menus, interaction windows, etc. — help is available in the form of an itemized user's manual, which can be opened on the screen as a new window (cf. Microsoft Word). Electronic user's manuals are also available in older application environments (see, e.g., Mozeiko, 1982) and in connection with some operating systems (cf. Unix; see, e.g., Morgan and McGilton, 1987). The Macintosh user may often define macros and customize menus, too. Service in "serious decision situations" is available automatically: questions about whether or not the user *really* wants to do something are posed. The style of representation contains features which, if implemented in a more conventional way, could be called help functions.

A lot of effort has also been used on modelling the user's behaviour when she navigates through a system along more or less curly routes (see, e.g., Alty, 1984; Canter and others, 1985). This approach has some common features with trail analysis, although the time perspectives are different (trail: historical routes; navigation: future routes). This branch of research indicates that "the lost user" is a common phenomena, and attempts to solve the problem are made. Anyhow, I claim that modelling the user is not sufficient. Instead, it should be studied *why* the users get lost, and then try to cure the *reasons*.

I regard all user support approaches which were discussed above as important progress in the area of user support. Most of them are, however, relatively restricted in scope and static in functionality, especially when analyzed from an organizational perspective where the focus is not only on a particular work situation or information processing task, but on a large network of tasks — the organizational context. Attention to context sensitivity has been paid, but the treatment of the topic is often restricted to concern a certain area or feature of an application. One example of contextuality are window systems (Reichman, 1986), where context support means supporting the relations between objects, i.e., windows and items. Dzida and others (1987) treat contextuality similarily: work context is a component of a tool interface which is managed by using window techniques. Carter and Schweighardt (1987) tie the concept to constructing the actual functions according to the user's request, the applicational situation, current data, and such parameters. These approaches include important indications in the direction of the notion of context. Nevertheless, I claim that a still broader context oriented approach is needed in order to properly facilitate the functioning of an organization *as a whole*. Inquiries into this issue will be made in the following.

3. BUILDING CONTEXTUALITY

In the following I will use the concepts of communication, coordination, and collaboration to build a framework for realizing context sensitive support systems. A central diligence in an organization is collaboration. (This does of course not imply that conflicts are excluded.) Collaboration is here interpreted as coordination of organizational tasks. I focus on cases where task coordination can be understood for instance as an act of transferring the *responsibility or control of performance* from one organizational unit (i.e., person, working group, department, etc.) to another, giving or receiving an impulse (*triggering* action), or *joining* commensurable, material and/or information *objects* together. (These restrictions entail that relatively bureaucratic organizations are focused on; see Ciborra (1983) and Ouchi (1980).) Task coordination, in turn, fails without some kind of communication. The realization of a context sensitive support system is here approached by using this conceptual framework. Both the analysis of the current work and information system and the implementation of a support system must draw on this framework.

In connection to the above mentioned concepts, another important assumption is made: the network of organizational transactions must be understood as *human acts*. (Nurminen (1982) provides my source of viewpoints on actor-centered ISs.) In other words, (information processing) transactions must not be regarded as autonomous incidents. Thus the computer may not appear as a self-sufficiet actor, either. The user as a responsible performer of (information processing) tasks is a central conception. The network of transactions is defined *as* and *by* a network of collaborating users. The information system may thus be interpreted as having a mediating role: it mediates (direct and indirect) communication and thus task coordination. Perceiving an organizational texture of this kind *in a use situation* can be facilitated by introducing a context sensitive support system which is capable of illustrating both human actors' work tasks and the inter- and intra-job task relations. Context sensitivity is thus seen as a *compound* of two processes: *information processes* and *work processes*. (The types of help functions which were discussed earlier in this paper belong mainly to a *subset* of the first one.) Information processes can be characterized as the actor-originated (inter- and intra-job) "behavior" of information. Work processes are composed of the human actors' performance of job tasks.

Realizing the notion of context can be approaced by applying related work. Reichman (1986) provides a characterization of context support. She claims that context support entails two things: knowing when things should be interpreted together and knowing when things should be interpreted separately. In addition, is is necessary to provide visual reflection of interconnection. Although these ideas are presented in a window management context, they can be generalized. I suggest

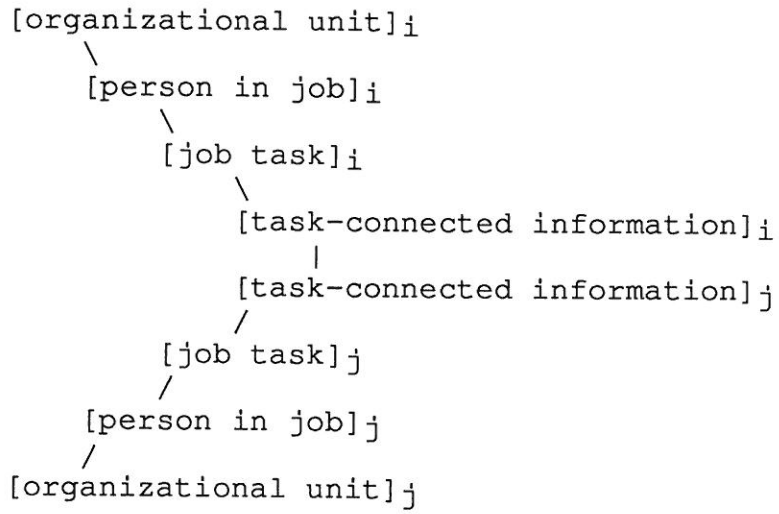
similar qualities, although designed so that the context covers both work and information processes and the interface combining these.

Lutze (1987) presents an approach to a contextual help system in a task context. He distinguishes between the following help services: situational services (explaining the current situation of the task), actional services (supporting, e.g., the understanding of presuppositions, effects and consequences), comparative services (e.g., contrasting actual achievements and original intentions), categorical services (performance classification), locative services (emphasizing the summarization and abstraction of the most relevant parameters of a task situation), and illocutive services (extracting, elaborating and completing the user's goals with respect to his original intentions. The type of functionality that is embedded in these services is covering, although the 'context' in Lutze's work can be interpreted as referring to information processes only. Stretching this type of functionality to yield the work process description, too, comes close the kind of contextuality that is proposed in this paper.

Realization of context sensitivity yields both the front end (user interfaces) and the back end (database).The user interface of a system like this takes in my opinion a graphical form: its power is in visuality. Independently of the degree of detail, both processes can be illustrated graphically. There exists actually several possible views in a support situation: the processes themselves may be studied with varying amount of detail, and the illustration of the chosen perspective may be rather abstract or very concrete. Coutaz (1987) presents an approach to contextuality which is applicable here as an example. An object oriented dialogue design where the design is based on the three-fold structure of abstraction, presentation and control, is presented. The control part of the structure keeps track of the linkages between the abstract (or formal) descriptions of objects on one hand and the the object as they are presented to the users. The context part maintains contextual information which is composed of the following categories: history, help, explanation, customization, and multi-threaded dialogue. A concept of context is thus introduced in accordance to information objects, but it is possible to broaden this type of presentation to yield objects which represent, e.g., job tasks, too.

The functionality of a context sensitive support system can be based on the mappings of work processes and information processes in the back end. This can be realized by introducing a context database which is functionally combined with a database. This database is to make possible and manage the multi-level support to the users. Communication may be regarded as the "glue" in an organization since it is possible to interpret it as a set of *information relations*. *People relations* (task coordination and organizational collaboration) may thus be reconstructed by using the information relations. This idea is based on the assumptions made about the conceptual construct of organizational collaboration. Simplified to some de-

gree, for instance the following kind of mapping could be incorporated with the context database and then visualized:



(The indexes i and j represent the symmetry between the opposite elements of the chain.)

The results of inquiries to the support system might then be presented in terms of information, task, and people relations (i.e., at the levels of information-information, information- task, task-task, task-person, and person-person relations).

In an office environment the following kind of chain could be visualized to the users: [In the marketing department's office] — [a secretary] — [registers the contract information] — [and produces an order file for the customer.] — [This information is used to produce the bill for the shipment received] — [when invoicing the goods] — [by a reckoner] — [in the office]. In a manufacturing organization, in turn, the above mentioned construct could take the form of the following chain: [In the packing unit] — [a worker] — [registers the packed production] — [and produces an identification label for the pallet] — [The identification information is necessary for giving the pallet a shelf position address] — [when storing the products] — [by an inventory worker] — [in the inventory]. These are *examples of possible* chains. So, in addition to these, there are several other feasible chains. It would also be possible, using the same construct as above, to illustrate for instance the (coarse) task flow of a job, a detailed description of an information object and its relations to other obects, a composition of a working group in terms of jobs and their definitions, etc. The main point is that the contextual connections of information processing in the organization become perceivable.

4. STRUCTURING THE ANALYSIS

In order to map the work and information processes and to lay an orderly design for a context sensitive help system, the analysis of the IS has to be structured in one way or another. I propose the usage of five groups of objects of analysis which are presented in the following. The main function of these groups is to aid the analysis so that the entities needed for a context database could be identified. The groups can be utilized in combination with, e.g., interviews and observations or as a supplement to more structured methods of analysis, which, anyhow, are often very integration oriented and focus more on such phenomena as data flow than for instance task analysis, communication, or computer support to work. The groups are first of all to be understood as *check lists*, which help in organizing the effort.

The groupings result from ideas synthesized from several different sources as well as they originate from the empirical research work that I have done in the case organizations of the Knowledge and Work project. The main areas of focus with respect to the groups are: 1: task analysis; 2: organization of work; 3: job design; 4: computer applications; 5: information media. The enterprise of organizing the analysis in a relatively general manner is first of all based on the assumption that proper understanding of the primary (work and information) system is vital in order to develop a decent secondary system (a help system).

The identification of the groups of entities is mainly based on the following arguments: Firstly, whether the current activity is pre- or post-design analysis it is helpful to focus on *coherent areas of phenomena.* So, the risk of working randomly is reduced at the same time as the analyst's freedom to interpret the reality in accordance with the specific organization is retained. This is because the groups are far from normative: they indicate *what* is to be focused on, but not *how* to. Secondly, the phenomena which are studied have to be identifiable before the analysis is started. Especially the use of qualitative methods, such as interviews and observations, suffer easily from improvised performance and "ad-hoc" interpretation of results. Anyhow, I consider structured *support* in many cases more acceptable than the use of highly quantitative methods or strict modelling techniques.

The groups are internally structured in the following way: The blank line in connection with 'a job' indicates that task analysis is primarily conducted within a certain job (with a name), secondarily between jobs. Tasks are supposed to have unique numbers (#) or codes of other kind. Moreover, certain task attributes either exist or do not. This (and other characteristics of this type) is indicated by a "check-box" ([_]). As a consequence of this structure, the groups may also be interpreted as forms (which can be used repeatedly and recursively; cf. subtasks).

Group no. 1 focuses on the examination of the task flow, the arrangements in terms of task interconnection, and jobs which are composed of the tasks. Distinguishing the individual/collective areas of the organizational information processing infrastructure is supported. Job-connected task analysis is more intensively analyzed than what is done by using conventional ISD methods (which seldom describe the users' jobs, but, instead, model the flow of data and transactions in terms of integrated information processing). My purpose, on the other hand, is to be able to analyze particularly *human activity*. I define a task as an identifiable activity which has a meaning for the performer and which is executed in the context of other tasks. The grain of analysis is rather fine since studying the *actual flow and context* of tasks is naturally as complex as the work procedures themselves. (Suchman (1983) provides a survey of (office) work procedures and confirms that the actual dimensions of work are far broader than generally assumed.) The main idea with this group is to facilitate the creation of an overall view of the *work processes*, and to support defining a clear area of related tasks and task responsibility for each user. This level of analysis provides first of all the task-task, and task-job (and hence job-person) linkages for the context database of a support system.

```
WITHIN A JOB _________________, FOR EACH IDENTIFIABLE TASK:

  Immediately before (another)  task # __
  Immediately after (another).. task # __
  Preparatory to (another)..... task # __
  After care of (another)...... task # __
  Complementing (another)...... task # __
  Compensating (another)....... task # __
  Superior to (another)........ task # __
  Subordinate to (another)..... task # __

  Performing job(s)...  _____________________________
  Responsible job(s)..  _____________________________
  Know-how in job(s)..  _____________________________

  Information processing task.. [_]
  Material management task..... [_]

  Automated task.............. [_]
  Computer supported task..... [_]
  Manual task................. [_]
```

Firstly, group no. 2 is designed for studying the formal and informal features of task performance and work organization. Official job descriptions and organization charts are thus necessary for comparisons. Confronting the formal organization with the informal one is used for acquiring information about deliberately produced and autonomously developed work practices which may concern as well

material as information processes. The placement of responsibility is an important issue. Performing tasks and being able both to evaluate and to take the responsibility of the consequences is essential. Also having a "good" distribution of different kinds of tasks is fundamental: it is for example well-known that as well too few as too many decision making tasks may end in disastrous frustrations. It is also apparent that making decisions and taking responsibility is easier if adequate support is available. Moreover, if a job contains for instance plenty of planning activities it is desirable to have corresponding support (like, e.g., project management can be supported by scheduling applications). So, different types of tasks (even within the same job) need different kinds of support. This, in turn, may have consequences to how the support system should be designed.

Secondly, communication, coordination, and collaboration — and especially the breakdowns and the reasons for these — are focused on. Conflicts and error situations are of importance, because the role of (lacking) coordination (maybe because of poorly arranged communication) is exceptionally visible when conflicts arise. Communicational aspects of work are central since organizational collaboration is largely based on these. Conflict and breakdowns are not interesting in themselves, but when it comes to supporting work, the analysis of problems gives valuable information on which to build: an important function of a context sensitive support system is to support organizational collaboration. Communication is here studied in two senses: direct communication (letters, telephone, face-to-face, electronic mail, etc.) and indirect communication (like via computerized information objects). This distinction provides a profound approach to information-information linkages.

```
WITHIN A JOB _________________, FOR EACH IDENTIFIABLE TASK:

  Communication linkage to (other)..    tasks # __
    Communication of type
        direct.............   [_]
        computer-mediated     [_]

  Coordinating between (other).......   tasks # __

  Cooperation with job(s) ____________________________

  Specialist task.....  [_]
  Collective task.....  [_]
  Operative task......  [_]
  Planning task.......  [_]
  Decision making task  [_]
  Knowledge work task   [_]

  Formally defined....  [_]
  Informally organized  [_]
```

```
Error prone task.....  [_]
Conflict source......  [_]
Survival purpose.....  [_]
```

Group no. 3 concentrates on the analyst's and, most important, the workers' perception of the tasks using well-known job design criteria (see, e.g., Buchanan, 1982; Mumford, 1983). The connections to sociotechnical ideals are obvious. For example lessening routine work and eliminating rudimentary tasks is desirable. Providing necessary support to demanding work tasks is also important. Currently available support to work may give indications of the areas where special assistance is needed. This group has a relatively vague *direct* relation to the idea of a context sensitive support system. However, meaningful jobs are far more fundamental than any help facility.

```
WITHIN A JOB _____, FOR EACH IDENTIFIABLE TASK:

  Task in its environment
    identifiable whole....  [_]
    fractional............  [_]
    rudimentary...........  [_]
    isolated..............  [_]
    autonomous............  [_]
    dependent.............  [_]
    authoritarian.........  [_]
    subversive............  [_]
    bureaucratic..........  [_]
    exceptional...........  [_]

  Level of stimulation
    interesting..   [_]
    meaningless..   [_]
    challenging..   [_]
    routine......   [_]
    altering.....   [_]
    neglected....   [_]

  Available support of type
    colleagues.......   [_]
    manuals..........   [_]
    help functions...   [_]

  Prevalent knowledge of type
    factual......   [_]
    proficiency..   [_]
    tacit........   [_]
```

Groups no. 4 and no. 5 support analyzing the information processing related areas of work. These forms take the information objects and the (eventual) com-

puter applications as points of departure. Neither group is exclusive in nature. It is important to notice that connections to job tasks are continuously maintained.

Group no. 4 focuses on applications. The types of applications that are listed on the form are the most typical kinds, and the applicational domains of these cover several types of organizations: both industrial and office uses are on view, and thus the names of applications are very general. For instance a design tool may refer (a) to an application used by an architect designing a house, or (b) to a software package used by a production manager for "experimenting" with different factory designs. The relations to the tasks in connection to which the applications do or will exist have to be kept visible since a central goal is to support work by introducing appropriate tools for the particular tasks. Careful analysis of the use of different applications is directly connected to the analysis of information-information and hence information-task linkages. A difference is also made between individual and shared applications. This is essential since the underlying idea is to support work which is defined by well-defined job domains. The applicational work area of a user should be commensurable with corresponding job tasks.

```
WITHIN A JOB _________________, FOR EACH IDENTIFIABLE TASK:

  Filing tool.............  [_]
  Application generator...  [_]
  Database application....  [_]
  Spread sheet............  [_]
  Hyperdocument...........  [_]
  Text editor.............  [_]
  Electronic mail.........  [_]
  Calendar system.........  [_]
  Project coordinatior....  [_]
  Statistical package.....  [_]
  Painting tool...........  [_]
  Drawing tool............  [_]
  Designing tool..........  [_]
  Decision support system   [_]
  Language compilers......  [_]
  Operating system level..  [_]
```

Group no. 5 focuses on information processing media used in connection to different tasks. Again, general terminology is used. The context in which the analysis is performed defines the medium in more detail. Document archives may refer to invoice folders, files to card indexes, layouts to factory floor designs, software documents to user's manuals or data dictionaries, etc. It is essential to study the level of computerization, i.e., whether the medium is manual or not: both media contain information relations which lay the basis for a context database. Sørgaard (1988) has pointed out the importance of focusing on computerized material and coordination that takes place via it. He focuses on the design of

computerized (shared) material and demands that the design has to explicitly support the cooperative aspects of using such material. The idea of a context database serves related purposes, although it goes beyond the use situation itself: a context database is to contain task-connected information about the use of the computerized and non-computerized media, too.

```
WITHIN A JOB _____, FOR EACH IDENTIFIABLE TASK:

  Document archives..... [_]
  Layouts............... [_]
  Calendars/Coordinators..... [_]
  Files................. [_]
  Memorandums........... [_]
  Diaries............... [_]
  Statistics............ [_]
  Graphics.............. [_]
  Letters............... [_]
  Text documents........ [_]
  Technical drawings.... [_]
  Organization charts... [_]
  Job descriptions...... [_]
  Software/hardware documents.. [_]
```

The formulation of these groups serves also the following purposes: Firstly, this type of groups allow combining several styles of presentation. As mentioned above, the groups may be used for supporting many techniques of analysis. On the other hand they themselves yield as well the use of codes (e.g., for combining tasks) as verbal legends (e.g., in task descriptions). Coding the "observations" may be used as a *grouping factor* for instance in connection to the definition of jobs on the basis of closely related tasks, or when "reconstructing" the communication channels that are being established around electronic mail. Legends may be introduced *as* a context and *in* a context.

Secondly, whenever such techniques as interviews or observations are intensively used, the amount of research material (e.g., tapes and diaries) tends to accumulate "wildly". Organizing such material *in an operational way* is often a non-trivial task. Using groupings like these as an organizing criteron is a feasible approach to structured documentation: combining the codes of, e.g., tapes and the groups may increase the usability of *both* materials.

The third point is that the groups are simple. The message is that instead of automatically relying on ready-made solutions (i.e., "methods") it is possible to create self-made techniques. (These groups can be used for instance for designing forms or other more formal techniques.) Religious belief in well-known methods results — as also well-known — in extremely clumsy standard ISDs. This is because the methods draw mainly on "objective models of the reality". I claim that

using *creativity* in connection with (a) developing new techniques and (b) using the techniques is to an unnecessarily high degree degenerated skill among ISD professionals.

Finally, the use of this type of support is not reduced to a certain phase of an ISD process. As the level of detail of the forms indicate the focus is not on issues which *directly* facilitate the IS implementation activities (such as, e.g., programming the user interface). *Indirectly* there is, however, a relation. Realization is necessarily based in analysis. There are means for sophisticated implementation of ISs (like object orientation), but analysis which makes the (re)design possible and goes beyond the ad-hoc level is equally important. It is sometimes unavoidable to start from scratch several times, but stages at which a "finer grain" is operational are many. Thus making it possible to (re)focus the analysis yields certainly full return.

5. DISCUSSION

The two most interesting consequences of a support system like the one described above are that the information system design itself becomes visible, and that the support system actually is an apparatus in an apparatus. ISD is something that becomes "real" when the system is implemented. For some the design is a collection of user interfaces and certain tasks performed at a terminal. To another group it may appear as a construction of physical equipment and the management of a database system. The system is naturally composed of all such elements and much more. But as stated earlier, the end user is probably the one who has the lousiest overall picture of the IS as a whole. Realizing context sensitivity makes the design visible for the "ordinary" end users, too. The help system is actually a kind of "living" documentation of the IS and its design. It is thus an apparatus which is functionally connected to the IS but with which one can do *different* things than direct task performance.

The fact that e.g. questions 'where did this piece of information come from', or 'who will be influenced by this operation' become feasible to answer, has also concrete consequences. In a situation where the correct course of action is not known, it is practical to be able to ask what happens (and to whom) if this-and-this is done. Firstly, as these questions can be posed to the support system, the spreading of risky transactions can be reduced, and thus alternative solutions developed. Secondly, exceptional transactions with which people normally would hesitate may be allowed if the help system tells that no seriously harmful transactions are to be carried out. These are the kind of functions that are more common in for example decision support systems, although there based on parameterized models.

The value of a context sensitive support system may also be evaluated in the light of support that is given to different types of tasks. Croft and Lefkowitz (1984) provide an (office) task taxonomy which is practicable here, too. According to their taxonomy a context sensitive support system would facilitate assistance with complex tasks, handling unusual actions, context switching, status inquiry and explanation, and handling multiuser tasks. The other types of functions — automation of routine tasks, adaptability, and task invocation — are not directly supported.

Focusing on communication and collaboration implies a possibility to shift the focus from an individual to a group. A network of organizational connections is established by the use of the IS, and the network covers both horizontal and vertical relationships. Hence, another assumption is made simultaneously: a context sensitive help system can give support to different kinds of organization.

Organizational work processes take place within several types of organizations (Ciborra, 1983; Ouchi, 1980) where the scope yields from the most hierarchical/bureaucratic systems to systems where the informal aspects of organizing work are prevailing (groups and collectives). Even though the global organization could be classified as being a hierarchical or a matrix organization, the formation of various local organizations is more than probable. A group which functions autonomously with certain "inputs" and "outputs" can have a very antibureaucratic internal organization of work. Perceiving this highly social and human reality of an organization is not supported by conventional help systems.

A true context sensitive help system allows the end user to see the spreading of transactions as well horizontally as vertically. It also makes it possible for a user to analyze the consequences of her doings and non-doings in regard to the information processes and work processes — her own and those of her colleagues. This is possible by incorporating the database with context data which describes task coordination and collaboration practices *between people*. A help system can support getting an overall view, which of course is based on the known rules of information and work processes (i.e., bureaucracy), but which from another point of view is apt to make the global organization less bureaucratic in an interpretative sense as possibilities for understanding increase. Especially things which earlier have simply been learned in may gain meaning in a larger context.

The idea of help functions, which support users at many levels of the organization, and which reveal many levels of the organizational universe of discourse to the user, originate from empirical observations of users at work (Eriksson and others, 1987). I experience the use situation's quality as essential for the individual and the whole organization. In addition to comprehending one's own use environment, it is important for the users to perceive the organizational context of their work, e.g. the connections of their tasks to the tasks of other workers.

A context sensitive help system can by its context database linkages be used for

studying the contextual definitions of other jobs, too. The increase in system perceivability is not (as such) enough for prohibiting the creation of chaotic views. As it becomes possible to see from different perspectives and with more or less details what happens within the IS, it also becomes very important to point out the areas where each user is "at home". This means defining the personal areas of information (sub)system "ownerships" and responsibility, i.e. explicitly frame the jobs and corresponding information (sub)processes of individuals or closely collaborating groups. The degree of detail needs not always yield to the individual, since often a group of users works in so tight collaboration with each other that focusing on single users would be artificial. In such a case, a collective system without personal domains might be enough. An example of a situation like this is a self-steering group or a group rotating jobs.

In spite of the fact that information technology is nowadays frequently used and that attitudes towards it are slowly becoming "peaceful", the use itself is often seen as something separate from the "normal" or "real" job tasks. In many concrete use situations of integrated ISs it is continuously difficult to employ the famous *tool* abstraction (see, e.g., Sundin, 1980) even though these are situations where such abstractions could be very supportive for the users. A context sensitive help system can be used to fade the boarderline between information tasks and other tasks, because the information and work processes are so seamlessly interconnected (see Nurminen, 1982). One can take a perspective which focuses on the information processing practices, but the user is never left alone with these aspects: the connection to the job tasks is visible all the time. This applies of course the other way around, too, when one chooses the work process perspective. Also the different types of questions that can be posed to the system at all levels emphasize the fact that each job contains different tasks (e.g., planning, decision making, and operational tasks). This can, in turn, be regarded as broadening the scope of so-called knowledge work.

Studying the social and human character of ISs — especially different forms of communication, coordination and collaboration mediated by the IS may from an individual worker's point of view include something very important. Emphasizing the role of knowledge work at all levels in an organization is essential. At managerial levels it is generally accepted that unstructured and complex tasks are part of everyday work. Accordingly, so-called tailor-made ISs are quite common. However, at operational levels in organizations, routine work, fractioned jobs, distant information processing, scattered responsibility, etc. can be seen. Supporting the elimination of this kind of features by context sensitive help functions increases the quality of work at all levels of organizations. This can certainly be called promoting human factors.

ACKNOWLEDGEMENTS

I would like to thank the Knowledge and Work Project, the Academy of Finland, the University of Turku, and the Doctoral Education Programs of Information Technology in Finland for financing my studies. Pål Sørgaard provided constructive and useful comments.

LITERATURE

ALTY, J.L.(1984). The application of path algebras to interactive dialog design. Behaviour and Information Technology, Vol.3, No. 2, pp. 119-132.

BROWN, J. W. (1982). Controlling the Complexity of Menu Networks. Communications of the ACM, Vol. 25, No. 7, pp. 412-418.

BROWN, P. S. and J. D. GOULD (1987). An Experimantal Study of People Creating Spreadsheets. ACM Transactions on Office Systems, Vol. 5, No. 3, pp. 258-272.

BUCHANAN, D. A. (1982). The development of job design theories and techniques. Gower, Hants, England.

CANTER, D., R. RIVERS, and G. STORRS (1985). Characterizing user navigation through complex data structures. Behaviour and Information Technology, Vol.4, No. 2, pp. 93-102.

CANTER, D., J. POWELL , J. WISHART, and C. RODERICK (1986). User navigation in complex database systems. Behaviour and Information Technology, Vol.5, No. 3, pp. 249-257.

CARROLL, J. M. and J. MCKENDREE (1987). Interface Design Issues for Advice-Giving Expert Systems. Communications of the ACM, Vol. 30, No. 1, pp. 14-31.

CARTER, J. A., JR. and M. SCHWEIGHARDT (1987). The Basis for User Oriented, Context Sensitive Functions. In BULLINGER, H.-J. and B. SHACKEL (Eds.). Human-Computer Interaction - INTERACT '87. North-Holland. Pp. 1027-1032.

CHRISTIE, B. (1985). Personal Information Systems.In CHRISTIE, B. (Ed.). Human Factors of Information Technology in the Office. John Wiley & Sons, New York. Pp. 127-144.

CIBORRA, C. (1983). Markets, Bureaucracies and Groups in the Information Society. Information Economics and Policy, No. 1, pp. 145-160.

COUTAZ, J. (1987). PAC, an Object Oriented Model for Dialog Design. In BULLINGER, H.-J. and B. SHACKEL (Eds.). Human-Computer Interaction - INTERACT '87. North-Holland. Pp. 431-436.

CROFT, W. B. and L. S. LEFKOWITZ (1984). Task Support In an Office System. Communications of the ACM, Vol. 2, No. 3, pp. 197-212.

DZIDA, W., C. HOFFMANN, and W. VALDER (1987). Mastering the Complexity of Dialogue Systems by the Aid of Work Contexts. In BULLINGER, H.-J. and B. SHACKEL (Eds.). Human-Computer Interaction - INTERACT'87. North-Holland. Pp. 29-33.

ERIKSSON, I., R. KALMI, and M. NURMINEN (1987). A Method for Supporting Users' Comprehensive Learning. In DEGROSS, J. and C.H. KRIEBEL (Eds.). Proceedings of the Eighth Annual International Conference on Information Systems. Pittsburgh, Pennsylvania, December 1987. Pp. 195-217.

KUSTERER, K. C. (1978). Know-How of The Job: The Important Working Knowledge of "Unskilled" Workers. Westview Press, Boulder, Colorado.

LEHMAN, J. A. (1985). Personal Computing vs. Personal Computers. Information & Management, No.9, pp. 253-259.

LUTZE, R. (1987). Customizing Help Systems to Task Structures and User Needs. In BULLINGER, H.-J. and B. SHACKEL (Eds.). Human-Computer Interaction - INTERACT'87. North-Holland. Pp. 871-878.

MARTIN, J. (1982). Application Development Without Programmers. Prentice-Hall, Englewood Cliffs, New Jersey.

MORGAN, R. and H. MCGILTON (1987). Introducing UNIX System V. MacGraw-Hill, Singapore.

MOZEIKO, H. (1982). A Human/Computer Interface to Accomodate User Learning Stages. Communications of the ACM, Vol. 25, No. 2, pp. 100-104.

MUMFORD, E. (1983). Designing Human Systems for New Technology. Manchester Business School. United Kingdom.

NIEVERGELT, J. and J. WEYDERT (1980). Sites, Modes, and Trails: Telling the User of an Interactive System Where He Is, What He Can Do, And How He Gets to Places. In GUEDJ ET AL. (Eds.). Methodology of Interaction. North-Holland, Amsterdam. Pp. 327-338.

NORMAN, D. A. (1984). Worsening the Knowledge Gap: The Mystique of Computation Builds Unnecessary Barriers. In PAGELS, H. L. (Ed.). Computer Culture. The Scientific, Intellectual, and Social Impact of the Computer. Annals of the New York Academy of Sciences, Vol. 426, November 1.

NURMINEN, M. I. (1982). Human-scale Information Systems. Lecture notes. Institute for Information Science. Universiy of Bergen. Norway.

OUCHI, W. G. (1980). Markets, Bureaucracies, and Clans. Administrative Science Quarterly, Vol. 25, March, pp. 129-141.

REICHMAN, R. (1986). Communication Paradigms for a Window System. In NORMAN, D. A. and S. W. DRAPER (Eds.): User Centered System Design. New Perspectives on Human Computer Interaction. Lawrence Erlbaum Associates, Inc., Publishers, Hillsdale, New Jersey. Pp. 285-313.

RILEY, MARY S. (1986). User Understanding. In NORMAN, D. A. and S. W. DRAPER (Eds.): User Centered System Design. New Perspectives on Human Computer Interaction. Lawrence Erlbaum Associates, Inc., Publishers, Hillsdale, New Jersey. Pp. 157-169.

SUCHMAN, L. A. (1983). Office Procedure as Practical Action: Models of Work and System Design. ACM Transactions on Office Information Systems, Vol. 1, No. 4, October, pp. 320-328.

SUNDIN, B. (Ed.) (1980). Is the Computer a Tool? Report from a Symposium in Stockholm/ Sigtuna, Sweden, June 1979. Almqvist & Wiksell, Stockholm, Sweden.

SØRGAARD, PÅL (1988). Object Oriented Programming and Computerised Shared Material. To appear in GJESSING, S. and K. NYGAARD (Eds.). Proceedings of the Second European Conference on Object Oriented Programming (ECOOP'88). Oslo, Norway, August 1988. Springer-Verlag, Heidelberg.