HUMAN-COMPUTER CO-OPERATION AND LEVELS OF KNOWLEDGE

ERNEST A. EDMONDS
LUTCHI Research Centre
Department of Computer Studies
Loughborough University of Technology
Loughborough UK

Abstract

In a project funded by the UK's Alvey Programme, Human-Computer Co-Operation, considerable use was made of explicit knowledge in the system. This paper describes the underlying philosophy used in structuring and employing that knowledge and how it is being developed in current research into Knowledge Support Systems. The example of computer support for creative design exemplifies the argument.

Introduction

Many of the key results of the project, Human-Computer Co-operation, have recently been published as a special issue of a journal (Edmonds, 1993). In the first paper, Clarke and Smyth (1993) review the meaning of the word *co-operation* and draw out certain requirements for a computer system to be considered co-operative. They identify a small number of underlying mechanisms that the system must possess. Central to these is what they term the *agreed definition knowledge-base*. This knowledge-base provides the basis of a shared language that facilitates the necessary communication between human and machine. They note that it is essential for the human user to have access to this knowledge-base and for them to be able to update it. Related conclusions have been reached by others including, for example, Fischer (1993).

The Human-Computer Co-operation project also examined the issue of the manipulation of knowledge in the system by the end-user (Edmonds et al, 1993; Candy et al, 1993). Underlying this work was the recognition that the knowledge in the system was of different categories and that both the need for and mode of access varied according to the type of knowledge and the user in question. This paper explores these issues and briefly introduces the recent development in *Knowledge Support Systems* (Edmonds and Candy, 1993).

Knowledge in the System

A key question to ask about knowledge represented in the system is, what kind of entity is the knowledge about? Figure 1 illustrates the *levels of knowledge* used in the project in question.

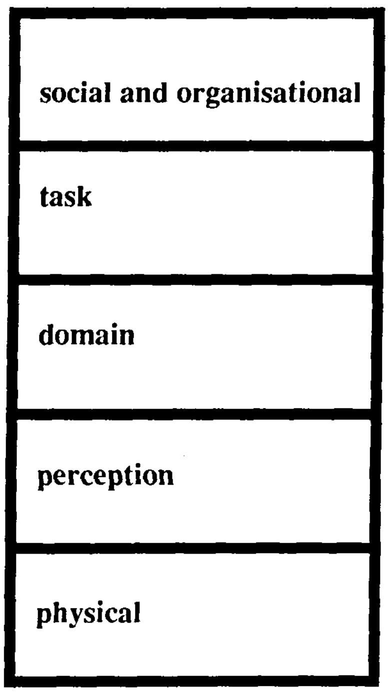

Figure 1: Levels of Knowledge

The levels of knowledge being employed can best be explained in relation to a specific example. Suppose that a user is engaged in deciding about the specific location of a proposed house and, for that purpose, appropriate map data is being displayed on the screen in support of that task. We might consider the specific examples of knowledge types shown in Figure 2. The *social and organizational* issues that must be considered include the relevant planning regulations, for example. The specific *task* is defined in terms of the requirements of the house and its relationship to its geographic site. When looking at the map on the screen, the *domain* knowledge that can be utilised in coming to a solution will be expressed in terms of objects such as houses, roads and so on. Below that level, however, the user will *perceive* lines, squares etc that, in the context of the domain, will have particular interpretations. Finally, these representations will be put into effect by *physical* events such as the display of a given pixel in an appropriate colour.

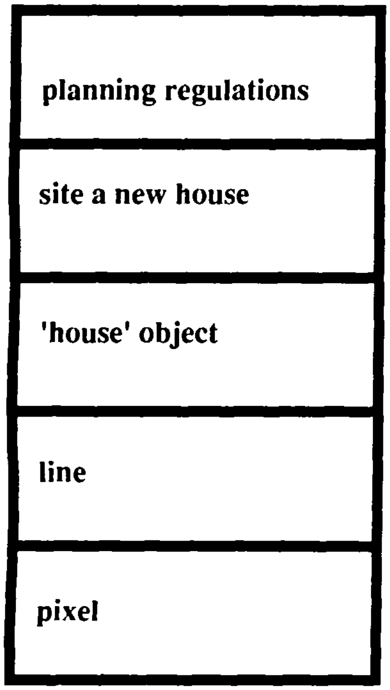

Figure 2: Examples of Levels of Knowledge

We may communicate with the system at many different levels of representation of the issues of concern. All are relevant to the solution of the problem. However, as far as the end-user is concerned, some may not be of any real interest. For example, the physical level would not normally matter and alternative representations of the data might be provided without any significant effect. At the other extreme, social or organizational knowledge may not be represented in the computer system at all. This might be entirely up to the user to provide, either explicitly or implicitly. Alternatively, we may find that a distinct computer process is used in relation to social and organizational issues, i.e. a candidate solution arrived at the task level may be submitted to an evaluation system that operates at the social and organizational level. In the next section we consider the degree of interaction with the knowledge in the system.

Interacting with the Knowledge

As mentioned above, a key issue is the ability of the user to interact with the knowledge in the system. Thus the picture must be enlarged to include a *User Interface Management* (UIM) system that enables that interaction (Figure 3).

In conventional computer systems, the fundamental knowledge, up to and including the domain knowledge is 'hard-wired' and therefore not available to the end-user for inspection or modification. A UIM does not provide access to much of the knowledge about

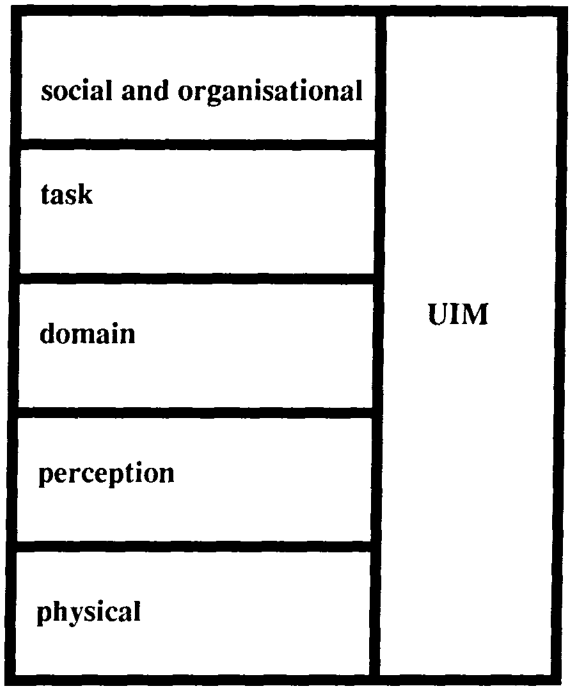

Figure 3: Interacting with the Knowledge

the domain, for example. Conventional wisdom and textbook knowledge is used to provide the framework in which most systems operate. In addition, the social and organizational issues are often not represented at all within the system. The role of the UIM may normally be reduced to that illustrated in Figure 4. However, in the ideal case, access might be provided as in Figure 3. In practice, the fact that *task analysis* is an important activity in system design indicates that it is normal for the user to be bound by strict limits, that are greater than Figure 4 suggests. In other words, the definition of just how the system supports the task may be very rigid and out of the user's control.

The extent of interaction that is required varies considerably in relation to the user, the task and the situation in which the work is being performed. Figure 5 gives an impression of this variation by the width of the right hand element. The greatest concern of the user is with handling the task in hand, but the social and organizational context also requires careful management and maintenance.

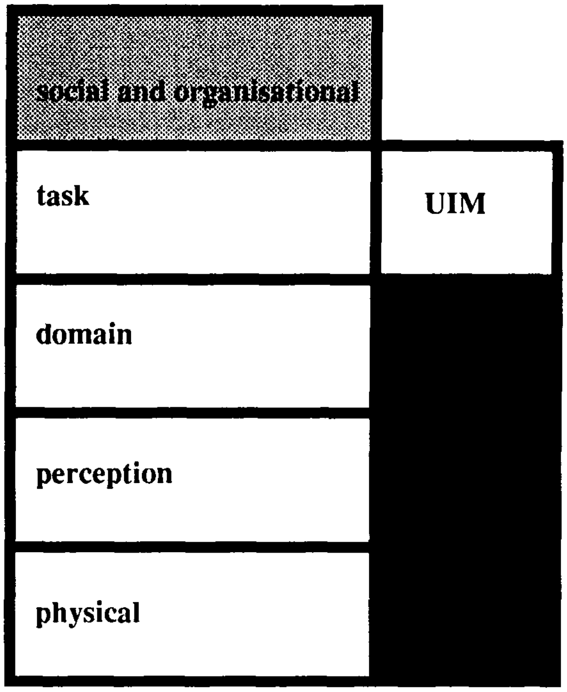

Figure 4: The typical scope of the UIM

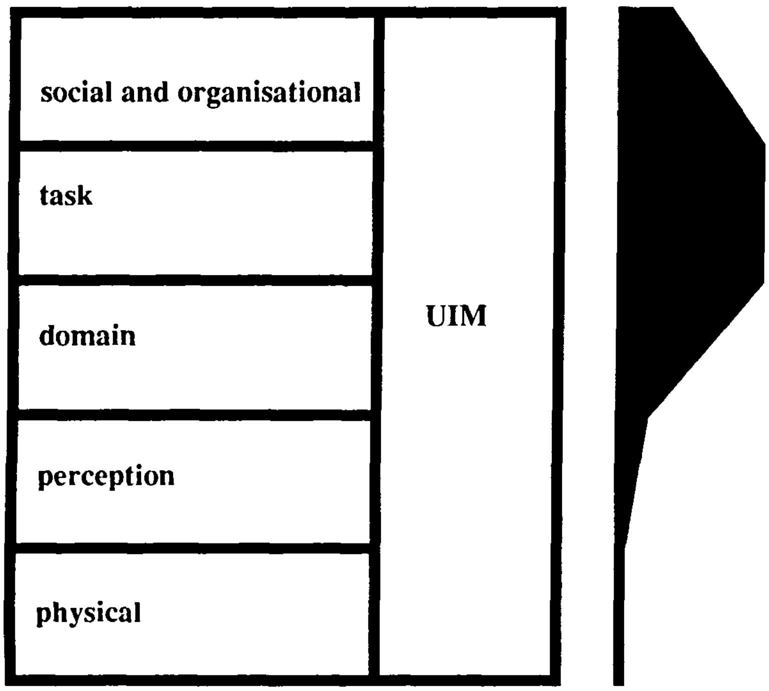

Figure 5: The required extent of interaction

Typically, the domain knowledge is determined on the advice of domain experts, the perception knowledge is built in on the advice of Human Factors experts and the physical issues are determined by computer system specialists. It is clear that more information in the system about the social and organizational context would be valuable. However, it is dynamic information and, hence, the issue of maintenance, i.e. the updating of the knowledge, that is of central importance in solving the problems under consideration.

Knowledge Support Systems

When implementing domain knowledge the characteristics of the particular user must be taken into account. Whilst a technician might be carrying out a task according to well specified procedures, a domain expert will also be reflecting upon those procedures whilst working (Schön, 1983). Thus the expert is not working with fixed, tightly defined, knowledge. Indeed, the evolution of that knowledge is central to their work.

This brings us to the concern of Knowledge Support Systems. The key point of such systems is that professional workers need to be able to modify the domain knowledge of any system that supports them. The definition of that knowledge, in fact, is close to the definition of their expertise. For such users, the UIM must provide access to levels of knowledge at least down to the domain level (Edmonds and Candy, 1993).

Knowledge Support Systems, where the end user manipulates machine representations of knowledge directly (Shaw and Gaines, 1988), have been developed in LUTCHI and applied to scientific exploration (O'Brien et al, 1992a; O'Brien et al, 1992b; Candy et al, 1993a; Edmonds et al, 1993). This work has clearly demonstrated a potential for supporting innovative knowledge work (Candy et al, 1993b). At the same time, other workers have began to study the approach in the specific context of design (Fischer, 1990; Hori et al, 1993).

Research into the design process has suggested that many of the requirements for support tools are similar to those observed in the study of scientific exploration above (Visser, 1992; Fischer, 1992). For example, Visser proposes that tools to assist in the management of memory load would positively support design. Her results relate closely to those of Candy et al 1993a) in the use of Knowledge Support Systems in the science domain.

Key design requirements for Knowledge Support Systems have been identified from the above research. The resulting underlying user interface architecture is designed to be generally applicable and easily tailored for specific domain applications. The work has extended our understanding of how knowledge-based systems can support Science and the current investigation is taking the same approach in Design. The cognitive findings are concerned with the nature and accessibility of knowledge to the domain expert. The important distinguishing characteristic of this work is that it is concerned with complex domains in which the expert does not have complete knowledge but, rather, where the approach enables them to extend and refine their own understanding. This issue is central to creative professional work and, in particular, to Design.

An Application to Design

In a current study of the application of Knowledge Support Systems to conceptual design, the work of Mike Burrows, designer of the model for the LotusSport monocoque bicycle has been investigated. This study has revealed a number of factors that must be taken into account if such systems are to provide true human-computer co-operation to designers.

Very briefly, it is important that certain stages of the process are supported and that the support is linked between these stages. They may be characterised as follows:--

- goals and motivations
- ideas generation
- problem formulation
- determination of solution strategy
- methods for problem solution
- the employment of knowledge and expertise

Each of these stages can be seen as one in which a particular type of knowledge is investigated, advanced and employed. The manner in which this occurs is also important. In particular, the study by Edmonds and Candy (1993) concluded that Knowledge Support Systems for conceptual design have important specific interaction requirements. They must allow, at any time, for the user to:-

- take an holistic view of the visual source data
- suspend judgement on any matter
- be able to readily make unplanned deviations
- return to old ideas and goals
- formulate, as well as solve, problems
- re-formulate the problem space.

Thus Knowledge Support Systems can offer improved human-computer co-operation with specific benefits, given that the important design requirements are met.

Conclusion

True co-operation between human and machine in context of complex tasks, such as Design, must involve the manipulation of knowledge in the system at quite a deep level. In particular, domain specific expertise must be addressed explicitly in a way that allows the expert user to modify and extend it. Knowledge Support Systems, which allow end users to manipulate knowledge represented in the system, are therefore proposed as the way forward in providing co-operative systems for professional people at work.

Acknowledgements

Mike Burrows, Linda Candy and the Staff of Lotus Engineering must be thanked for their important contributions to the work described in this paper. The research has been partly funded by the Alvey Programme, the Tri-Council Cognitive Science/HCI Initiative and by the ACME Programme.

References

Candy, L., O'Brien, S.M. & Edmonds, E.A. (1993). End user manipulation of a knowledge based system; A study of an expert's practice. Int. J. Man-Machine Studies, 38, 1, pp 129-145.

Candy, L., Edmonds, E.A. & O'Brien S.M. (1993). Amplifying creativity: the role of end user knowledge manipulation systems. In: Artificial Intelligence and Creativity. Dartnell, T (ed). Studies in Cognitive Systems, Kluwer Academic, Dordrecht (to appear).

Clarke, A.A. & Smyth, M.G. (1993). A co-operative computer based on the principles of human co-operation. Int. J. Man-Machine Studies, 38, 1, pp 3-22.

Edmonds, E.A., McDaid, E., O'Brien, S.M. & Bayley, T. (1993) Constructing end-user knowledge manipulation systems, Int. J. Man-Machine Studies, 38, 1, pp. 51-70.

Edmonds, E.A. & Candy, L. (1993). Knowledge support for conceptual design: the amplification of creativity. HCI International '93, Elsevier, Amsterdam, pp 350-355.

Fischer, G. (1990). Communications Requirements for Cooperative Problem Solving Systems. International Journal of Information Systems. 15, 1, pp 21-36.

Fischer, G. & Nakakoji, K. (1992). Beyond the macho approach of artificial intelligence: empower human designers - do not replace them. Knowledge-Based Systems. 5, 1, pp 15-30.

Fischer, G. (1993). Creativity enhancing design environments. Gero & Maher (editors), Modelling Creativity and Knowledge-Based Creative Design, Erlbaum, Hillsdale, New Jersey, pp 269-282.

Gaines, B. (1990). Knowledge support systems. Knowledge Based Systems, 3, 4, pp 192-201.

Hori, K., Sugimoto, M. & Ohsuga, S. (1993). Application of articulation aid to design. In : Information Modelling and Knowledge Bases IV, Oaakkola, H. et al (eds), IOS Press (to appear).

O'Brien, S.M. Candy, L., Edmonds, E.A., Foster, T.J. & McDaid, E. (1992a). End user knowledge manipulation systems: the speech knowledge interface. In: Proc. of 20th Annual Computer Science Conference, Agrawal, J.P., Kumar V. and Wallentine, V. (eds), The ACM Press, pp 359-366.

O'Brien, S.M., Candy, L., Edmonds, E.A. & Foster, T.J. (1992b). Knowledge acquisition and refinement using end-user knowledge manipulation systems. In: Applications of Artificial Intelligence X: Knowledge-Based Systems Conference, SPIE Proceedings Vol. 1707, Orlando, Florida. Biswas, G. (ed), pp 25-36.

Shaw, M. L. J. & Gaines, B. R. (1988). KITTEN: Knowledge initiation and transfer tools for experts and novices. In: Knowledge Acquisition Tools for Expert Systems. Boose, J. & Gaines, B. (eds). Academic Press, pp 309-338.

Schön, D.A. (1983). The Reflective Practitioner, Maurice Temple Smith, London.

Visser, W. (1992). Designers' activities examined at three levels: organization, strategies and problem-solving processes. Knowledge-Based Systems 5, 1, pp 92-104.