INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN
ICED 03 STOCKHOLM, AUGUST 19-21, 2003

EXPLORING THE STRUCTURE OF DESIGN PROBLEMS

Kees Dorst

Abstract

In this paper we will concentrate on exploring a subject that has been much neglected in Design Methodology: the structure of design problems. Design problems have been called 'ill-structured' or even 'wicked', but very little effort has been made to describe what 'weak' structure they have. This is an important subject, because this 'weak' structure must play a part in a designers' decision to tackle a design problem in one way or the other.
In a theoretical investigation we describe what we can find in Design Methodology about the structure of design problems. This will focus on the descriptions of design problems as 'illstructured problems', as 'underdetermined problems' and on the way design problems are treated within the two paradigms of design methodology. Then we develop an approach to the study of design problems, that takes the situated nature of design problems into account. A method for the empirical study of the structure of design problems is outlined, based on linkography.

Keywords: Design problems, Paradigms, Ill-Structured problems, Linkography

1. Introduction

Design Methodology has always had something of a blind spot for design problems: the focus in Design Methodology has almost exclusively been on the support of the *process* of designing. But any method for aiding design activities necessarily contains statements or assumptions about all three 'dimensions of design activities': the dynamics of a design process, the designer and the design problem [1], [2].
Within Design Methodology, the nature of design problems has been described as 'ill-structured', or even 'wicked' [3] , but little more has been said. Some process-focussed design methods, however, seem to incorporate strong assumptions about what design problems are (e.g. concerning the independence of subproblems, the objectivity of problems, the possibility to create an overvie of a design problem, etc). We need to know more about design problems, so that these assumptions need to be put to the test.
In this paper we will concentrate on exploring the *structure* of design problems. This is important, because if we can find a way to trace the structure of design problems, we can then match that to the way designers tackle those problems. This will lead to a much closer description and a much better understanding of the way designers work, and WHY they take the actions we see.

An initial theoretical investigation into design problems focusses on the question: (1) What can we find in Design Methodology about the structure of design problems? This will focus on the descriptions of design problems as 'illstructured problems', as 'underdetermined

problems' and on the way design problems are treated within the two paradigms of design methodology. This will lead to the second research question: (2) How can we further develop (i.e. combine) these approaches into a comprehensive study of the structure design problems? In this piece we will outline an approach that takes the situated nature of design problems into account. And then (3) we will ask ourselves how the structure of design problems can be explored in an empirical study.

2. Design problems

We will start out the exploration by mapping what we know of design problems. In the next sections we will focus on the descriptions of design problems as 'illstructured problems', as 'underdetermined problems' that can be found in Design Methodology, and on the way design problems are treated within the two paradigms that underlie Design Methodology.

2.1 Design problems as underdetermined problems

Design activities can be seen as the reasoning from a set of needs, requirements and intentions to a new bit of reality, consisting of a (physical) structure and an intended use. This process of reasoning is abductive: there is no closed pattern of reasoning to connect the needs, requirements and intentions with a form of an artifact and a mode of use ([4], [5]). This is called here the underdetermination of design problems. Upon closer inspection we can actually distinguish two ways in which a design problem is underdetermined:

- a description in terms of needs, requirements and intentions can never be *complete* (there can never be enough to define a form), and
- 'needs, requirements and intentions' and 'structure' *belong to different conceptual worlds* [6].

These two kinds of underdetermination alone may lead to the feeling that the solution of design problems must be well nigh impossible. But designers somehow overcome the underdetermination of design problems and the conceptual rift between 'need' and 'form' in their design processes. One of the problems designers face in doing this is that design problems are not completely determined, but also not completely free. Most design problems in fact seem to have a threefold nature:

(1) They are partly **determined** by 'hard' (unalterable) needs, requirements and intentions. A designer will have to reserve time in the early part of his design process to unearth these 'hard facts' by information gathering and analysis, and live with these specifications. This information can be seen as a necessary input at the start of the design porcess, and this type of interaction can very well be described and modeled within the rational problem solving paradigm.

(2) But a major part of the design problem is **underdetermined**. The interpretation of the design problem and the creation and selection of possible suitable solutions can only be decided during the design process on the basis of proposals made by the designer. These proposals thus entail both the possible interpretations of the design problem and possible solutions to those problems.

(3) Part of the design problem can be considered **undetermined**, in the sense that the designer is to a large extent free to design according to his own taste, style and abilities. (It is of course not the case that the designer would never have to defend these aspects of the design to others, but in these areas the designer is dominant in the sense that he also provides the criteria on which this aspect or part of the design is to be judged).

2.2 The ill-structuredness of design problems

In his paper 'The structure of ill-structured problems' [7], Herbert Simon defines well structuredness by six criteria:

> *(1) There is a definite criterion for testing any proposed solution, and a mechanizable process for applying the criterion.*
> *(2) There is at least one problem space in which can be represented the problem state, the goal state, and all other states that may be reached, or considered, in the course of attempting a solution to the problem.*
> *(3) Attainable state changes can be represented in a problem space, as transitions from given states to the states directly attainable from them.*
> *(4) Any knowledge that the problem-solver can acquire about the problem can be represented in one or more problem spaces.*
> *(5) If the actual problem involves acting upon the external world, then the definition of state changes and of the effects upon the state of applying any operator reflect with complete accuracy in one or more problem spaces the laws that govern the external world.*
> *(6) All of these conditions hold in the strong sense that the basic processes postulated require only practicable amounts of computation, and the information postulated is effectively available to the processes-i.e. available with only practicable amounts of search.*

Lawson [8], Van der Poel [9]and many others have held that design problems don't adhere to the first two basic criteria for well-structuredness, and therefore cannot adhere to the others either. In the next section we will describe the way ill-structured problems have nevertheless been described by Simon in the context of his view of designing as a rational problem solving process.

2.3 Design problems in the rational problem solving paradigm

To describe the ways in which design methodology has treated the whole issue of design problems we have to distinguish between the two fundamentally different paradigms that the field is based on.

The main paradigm of design methodology, in which design is seen as a *rational problem solving* process, was introduced by Simon in the early 1970s. In this paradigm, design is viewed as a rational search process: the design problem defines the 'problem space' that has to be surveyed in search of a design solution. Problem solving theory is concerned with the ways in which people or artificial systems arrive at solutions to problems they encounter. This theory can be captured by four propositions:

- *A few gross characteristics of the human Information Processing System are invariant over task and problem solver*
- *These characteristics are sufficient to determine that a task environment is represented as a problem space, and that problem solving takes place in a problem space*
- *The structure of the task environment determines the possible structures of the problem space.*
- *The structure of the problem space determines the possible programs that can be used for problem solving.* (from: [10])

If this theory is valid for design, design problem solving also takes place within a problem space that is structured by the structure of the task environment, which in its turn determines the 'programs' (strategies or methods) that can be used for designing.

In a later paper Simon [7] addressed some of the difficulties that might arise in applying the rational problem solving approach to design by defining design problems as 'ill-structured

problems'. Ill-structured problems are to be tackled in an 'immediate problem space'. This is a part of the total problem space which is deemed too large, ill-structured and ill-defined to be described. The immediate problem space is addressed and put together by an (unspecified) 'noticing and evoking mechanism'. The basic 'design' problem-solving process would however be basically the same as in other kinds of problem solving. With the exception that the goal of a design process is to arrive at a solution that is 'good enough', *'we **satisfice** by looking for alternatives in such a way that we can generally find an acceptable one after only moderate search.'*

2.4 Design problems in the reflective practice paradigm

A radically different paradigm was proposed fifteen years later, by Donald Schön [11], who describes design as an activity involving *reflective practice*. This constructionist theory is a reaction to the problem solving approach, specifically made to address some of the shortcomings Schön perceived in mainstream design methodology. Schön's starting point is his feeling that the paradigm of technical rationality hampers the training of practitioners in the professions. He believes that the design-component of the professions is underestimated, and that the nature of human design activities is misunderstood. He shows that in the training programmes of professional schools that recognise design as a core activity, design knowledge is defined in terms of generalities about design processes and declarative knowledge needed to solve design problems. No attention is paid to the structure of design problems and the crucial problem of linking process and problem in a concrete design situation. This 'action-oriented', often implicit knowledge cannot be described within the paradigm of technical rationality. But Schön insists that this kind of knowledge is vital for action-oriented professions like design. He does recognise, however, that this implicit 'knowing-in-action' is difficult to describe and convey to students. What can be thought about and taught is the explicit reflection that guides the development of one's knowing-in-action habits. This he calls *reflection-in-action*.

One of the basic assumptions of the theory of technical rationality is that there is a definable design problem to start with. Schön remarks that *'.... Although Simon proposes to fill the gap between natural sciences and design practice with a science of design, his science can only be applied to well-formed problems already extracted from situations of practice...'* Schön, on the other hand, does not make any such assumptions about the design problem. The description of design as a reflective conversation concentrates on the structuring role of the designer, setting the task and outlining possible solutions all in one ‘framing’ action. The strength of this framing action determines the amount of structure in the task. In reflective practice design tasks may be analysed and subdivided in a number of different ways, and there is no *a priori* way to determine which approach will be the more fruitful. Therefore, design task and solution are always and inherently developed together.
Schön thus seems to ignore the possible structure that design tasks and solutions might have, although he gives a table of 'Normative Design Domains' in 'The Reflective Practitioner'. These 'Normative Design Domains' could provide a categorisation for the description of design tasks, but unfortunately these domains are not connected to the core theories of reflective practice, and they are never mentioned again. Schön's failure to link the theories of reflective practice to a model of design tasks means that descriptions of design activities within this paradigm can not benefit from any structure that might be present in the design task.

3. Kinds of design problems

These paradigms of design methodology, Rational Problem Solving and Reflective Practice, can be the basis for a further exploration of the structure of design problems of design. A fundamental distinction between kinds of design problems can actually be constructed on the basis of the paradigms of design methodology themselves. This is a first, theoretical, step towards a typology of design problems.

3.1 The interpretation of design problems

The key to understanding how the different paradigms view design problems is rooted in their epistemologies. The rational problem solving paradigm is based upon positivistic epistemology, and the paradigm of reflective action is phenomenological in nature. Positivism and phenomenology differ quite strongly in the way subject (the acting person) and object (the outside world) are related. Positivism claims that a person lives in an objective world which can be known through his/her senses; the sensory data is then structured by an internal processing system. This structuring system interprets the data by using basic *a priori* categories. To know the objective world, a person should study it carefully and dispassionately, preferably with scientific methods. In phenomenology the person is not static, but an emotive social being with a history and an environment which heavily influences the person's construction of reality. And the subject is influenced (and in the end 'formed') by what he/she perceives. Therefore, person (often called subject) and object are inextricably connected (see [12], [13]). Positivism and phenomenology are on opposite sides of the spectrum ([14], [15]). Over the centuries a number of attempts have been made to bridge this gap - but none of these attempts has generally been recognised as successful. However, some relatively recent developments in hermeneutics could be useful when dealing with this dilemma in the specific case of design methodology.

Gadamer proposes to bridge this gap by defining the basic operation in the acquisition of knowledge (in this case about design problems) as ***interpretation***. And he claims that this interpretation is a dualistic activity: it is both a *'revealing of what the thing itself already points to'* and *'an attribution of value to something'* [16]. The 'revealing of what the thing itself already points to' could be called *'objective interpretation'*. This is the case when something outside (say, a piece of information) impresses its meaning upon the observer. The 'attribution of value to something' could be called *'subjective interpretation'*. This means that the subject, in an act of will, impresses meaning and value upon something.
If we apply this to design we can see that the type of interpretation that is dominant varies through the phases of design activity, and across design situations. The design activities in which 'objective interpretation' plays a major role are described well by the rational problem solving paradigm. Activities that involve 'subjective interpretation' are most easily described by the paradigm of reflective practice.

The decision whether a part of a design activity will involve 'objective' or 'subjective' interpretation ultimately rests with the designer working on the design problem. Empirical evidence has shown [17] that here are a number of influences on this interpretive behaviour of the designer:
• Design is an activity that is aimed at solving a problem for the outside world (increasing turnover and profit for the company, for instance). On the other hand, it is also a 'creative' process, underconstrained in the sense of giving the designer a certain measure of freedom. Inasmuch as a certain design project is a problem solving process for the outside world, it needs to be *controlled* and the design decisions must be justified to the stakeholders. In that case there is an emphasis to objectify the goals and decisions in the design project, to effectively eliminate

the implicitness and elements of 'subjective interpretation' from the design activities. Any perception and problem interpretation must then be made explicit and becomes a subject of negotiation between the designer and the stakeholders. Through this process of negotiating, design becomes a more or less 'objective' process, in which problem statements, programmes of requirements, ideas and design concepts are still *made* rather 'subjectively' and implicitly, but in the end are presented explicitly and evaluated in order to settle them and make them real objects in the world. This results in an explicitly controlled pattern of objective design activity which can be described accurately using a positivist approach like the paradigm of rational problem solving. 'Objectivity' of the steps in a design process and of the terms used to describe it can thus be considered an artificial *construction* by the designer(s) for special purposes.

• 'Subjective interpretation' can become very important in a design project (or phase) where the design problem is ill-structured. In such a case, subjective structuring is the only way to make sense of the problem. Structuring of the problem can be achieved by imposing personal goals of the designer into the design problem or by subjectively choosing priorities. In design there is no particular pressure to minimise the subjective interpretation of a design problem and its solution: the dominant goal in design is to produce a good design, on cost and on time. A designer thus has the privilege and the problem of working in both an 'objective' and a 'subjective' mode.

• Where a certain design project gives (or demands) freedom of choice to the designer, he/she has to depend on their own interpretation and perception of the problem to produce a result. Then design is essentially a subjective activity, which can be best described in terms of reflective practice. This is particularly true in the conceptual phase of many design processes, but this subjective approach could extend over whole design processes.

• Empirical studies [17][18] have shown that designers spend quite some time at the beginning of a design assignment to consider what kind of problem they had to deal with. They did this in terms of the constraints of the problem which imposed on their freedom to define their own goals. The freedom depended partly on the assignment that they were given, and partly on their personal style: some designers seemed more at ease with an 'objectivist' approach to design problems, others are more comfortable imposing influential frames on the project right from the beginning.

• Group- or organisational design processes tend to require a large number of objectifying statements and arguments [19] to keep everyone on track. This is even more extreme in multidisciplinary teams, where the basic level of shared understanding necessary for the completion of the job is more difficult to achieve.

• When a designer follows a procedure (a method or technique), there is no fresh interpretation of the design situation until the procedure is finished or breaks down. In episodes of a design project where there are few procedures, there are many more possibilities for the new, subjective interpretation of design problem and design situation. In routine or redesign work there are many procedures, which decreases the importance of subjective (re-)interpretation of design problem and situation.

3.2 Design problems as situated problems

This pragmatic way of dealing with the descriptions that the two paradigms of design methodology allow us to make of the structure of design problems and solutions is not really satisfactory. The dual description makes design look like something of schizophrenic activity, and it begs the question how the two descriptions of design are connected, or how they are possibly related to a single, underlying structure of design. We need to step back and look at the assumptions that underlie both paradigms of design methodology.

In the Rational Problem Solving paradigm, design is the searching and selection of a solution in a 'solution space'. Design problems may be ill-structured, but they should basically be treated in the same way as well-structured problems. In the Reflective Practice paradigm the stress is much more on the subjective interpretation and framing of the design situation by the designer. The frames that are brought to bear unpon a design situation determine the next steps in the solution process.
These approaches to design have been developed in the 60's and 70's, largely inspired by developments in AI and the cognitive sciences. The epic endeavour to build intelligent computer systems focussed on the ability of such a system to solve illstructured problems within an open context, somewhat comparable to designing. The systems were based on a Rational Problem Solving approach, representing the 'relevant aspects' of the world and setting up formal procedures that manipulate these reprentations to solve a problem. This approach (GOFAI) has failed [20]. Alternative approaches are now developed that are inspired upon the *situatedness* of problem solving activity [21], [15]. We will now explore what these approaches would mean for design research.

A fundamental choice that is associated with the approach of design as a situated activity is that the attention is refocussed, from the (formal) modelling of 'the design problem' towards the question what design problems are to the designer, seen through the eyes of the designer, in the design situation. We must first concentrate on the 'local' design problem that a designer faces, and bracket the 'overall' design problem as an abstraction (for now).
This means that we have to confront the vagueness (i.e.lack of overview) and subjectivity that is involved in local design actions and decisions. Seen from this perspective, 'The design problem' as such does not really exist as an objective entity in the world. There is an amalgamate of different problems that centers around the basic challenge that is described in a design brief. This amalgamate of problems is partly there to be discovered by the designer in the design process, and part of it has to be MADE by the designer. (In solving a such a complex of design problems, they are interpreted and 'framed' by the designer.) The process of 'approaching a design problem' or 'dealing with a problematic situation' becomes the vital clue to understanding what design problems are. The latter formulation is important: for much of the design project the problem solving steps can be quite logical, routine and implict, without a real choice for the designer. Dreyfuss holds that the REAL problematic situations are the results of a 'breakdown' in this normal, fluent problem solving behaviour (the problem becomes 'at hand', in Heidegger's terms). These 'breakdowns' then are the moments of real choice, the moments in which design probelem and design solution are (re) structured. So we need to focus on distinguishing and describing the nature of these breakdowns, the critical situations in design [22]. These breakdowns are the points that Schon describes as 'surprises', the turning points in the designer's reflective conversation with the situation. In the solution of these breakdowns 'objective' or 'subjective' interpretation can play a role. This is where the existing (but possibly implicit or unknown) structure of the design problem and the structuring actions of the designer meet. A well-structured problem 'leads' the designer (through deduction, or abduction with a clearly dominant result), an ill-structured problem requires something like a framing action.

4. Empirical study

To study the structures of design problems as they are perceived or constructed by designers, we need a research method that allows us to describe the behaviour of designers as operations

on the design problem. The research method should be neutral, and it should closely follow the design process on a very detailed level. With 'neutral' we mean here that it should be independent of the paradigms, and just monitor designing as closely as possible, using the terms the designer him/herself uses. We need to find a way to describe the design behaviour at a such a detailed level that allows us to see patterns of problem-related behaviour emerge, without being completely problem-specific (case-specific).
Other researchers before us have struggled with this problem, and a possible avenue to solve it is to concentrate on the linking behaviour of designers. The patterns of links that designers perceive and develop in the design problem can be traced with 'linkography', a method first used on design by Gabriela Goldschmidt, and later much refined and extended by Remko van der Lugt [23]. The latter has traced the way designers build upon eachother's ideas within a brainstorming session by constructing a 'link matrix' of such a session. In this method the ideas the designers had were listed on both axes of the matrix, and the connections between ideas could then be noted down as crosses in the cells of the matrix. This simple method already creates a wonderful overview of the way the brainstorming session progressed, and one could clearly see patterns emerge that were particular to certain brainstorming groups.

For the purposes of our study we do not want to follow a brainstorming session, but a complete design process, and we want to trace the linking behaviour of designers with regard to design problems, design solutions and the link between the two. This means that one matrix will not suffice, but that we need a system of three matrixes: one in which the design problems are on both the axes, one where the (elements of the) solution are on the two axes, and one in which the problems are on one axis, and the solutions on the other. If we make these matrixes, following the design process step by step, we will be able to trace the perceived problem and the design solution at hand at every moment of the design process. And we can distinguish patterns in the operations the designers perform upon these design problems and solutions.

If they really have a counterpart in design practice, the two kinds of design problems that are associated with the paradigms of Rational Problem Solving and Reflective Practice will also lead to different patterns in the matrixes, on a macro level. The Rational Problem Solving type problem would lead to design behaviour that first concentrates on the design problem, with only the occasional excursion into the solution, and that is very much aimed at establishing many links between parts of the problem. The Reflective Practice approach to a design problem would be visible from the many jumps between the problem and solution matrixes. Frames can be seen as a complete pattern of problem-solution connections that is imposed upon the existing problem and solution. An initial exploration using the team data of the Delft Protocols Workshop [18] seems to confirm these hypotheses.
It is interesting that this empirical method will allow us to view designing in a different way, as it were at a 90 degree angle to the 'normal', process-focussed methodology. All the design methods, techniques and strategies that we know will be mirrored in the patterns of the matrixes, and new patterns might emerge.

5. Conclusion

In this paper we set out to explore the structure of design problems, using the two paradigms of design methodology, Rational Problem Solving and Reflective Practice, as our basis. We have argued that to really understand design behaviour in practice, we also need to take the situated nature of design problems into account. Taking this fresh point of departure has farreaching consequences: it means that there simply IS no 'objective' and overall design

problem - so it is very little use looking for an extensive (complete) representation of the design problem in the head of the designer. If we can be convinced by Dreyfus that such a representation does not exist, the only 'design problems' we can study are clusters of 'local' problems that are mostly solved implicitly, except in situations of 'breakdown'.

If the structure of design problems is so very fragmented, and so closely linked to the solution process, this of course poses very real barriers for the developers of computer based problem solving systems - if there is no formal representation of the problem, how can you then define formal operations for solving a problem? For design methodology this need not be such an insuperable barrier. After all, we do not aim to make a designing computer system, but we want to investigate human designers to gather advice for other humans. We can still cautiously use both approaches, Rational Problem Solving and Reflective Practice, towards this goal. The hypothesis is then that Rational Problem Solving is better for describing the more determined problem stretches of problem solving, and a variant of reflective practice, with its sensitivity to interpretation and situatedness, could be used to pinpoint the structure in the underdetermined episodes of design thinking, especially the moments of 'breakdown' (or 'reframing'). But we have to be very careful in claiming that these two approaches to looking at design are valid representations of the design process as it really happens. They are only approximations of the real, situated design activity.

The next step forward in exploring design as a situated activity, is to empirically study the detailed process of design, and trace the structure of the design problem and solution that the designer can be seen to deal with in the moments the 'normal' (routine) problem solving behaviour breaks down. The first results of our empirical study have shown that the simple research scheme with the three matrixes does allow us to closely monitor the way designers link parts of the problem and solution. The different approaches that designers tend towards a design problem can be clearly mapped, and patterns of behaviour ('design strategies') do occur. By the time of the conference we will be able to present the results of this step.

This approach to design problems, and this empirical research method, could be a first step in opening up the research agenda in Design Methodology to include the structure of situated design problems, and the nature of 'critical situations' in design.

References

[1] Dorst C.H., "Comparing the paradigms of design methodoldogy", In: Hubka W (Ed), Proceedings of ICED 95, Heurista, Zürich, 1995.

[2] Roozenburg N.F.M. and Cross N.G., "Models of the design process - integrating across the disciplines", In: Hubka, V. Proceedings of ICED 91, Heurista, Zürich, 1991.

[3] Cross, N.G., (ed.) "Developments in Design Methodology", Wiley, Chichester, 1984.

[4] Simon, H.A., "Sciences of the Artificial", The MIT Press, Cambrigde MA, 1992.

[5] Cross, N.G. and Dorst, K., "Co-evolution of Problem and Solution Spaces in Creative Design", in: Gero, J. and Maher, M.L., (eds.); Computational Models of Creative Design, Key Centre of Design Computing and Cognition, University of Sydney, Australia p. 243-262, 1999.

[6] Meijers, A.W.M.,"The relational Ontology of Technical Artifacts", In: Kroes, P.K.

and Meijers A.W.M., The empirical turn in the philosophy of technology, Elsevier Science, Oxford, 2000.

[7] Simon, H.A., "The structure of ill-structured problems", Artificial Intelligence, 4, 181-201, 1973.

[8] Lawson, B., " How designers think: the design process demystified" (2nd ed) Butterworth, London, 1990.

[9] van der Poel., "De analogie tussen ethische problemen en ontwerpproblemen", In: Baars J. and Starmans R., (eds) Het eigene en het andere, Eburon Delft, 1999.

[10] Newell, A. and Simon HA,"Human Problem Solving", Prentice-Hall, Englewood Cliffs, 1972

[11] Schön, D.A., "The Reflective Practitioner", Basic Books, New York, 1983.

[12] van Manen M., " Researching lived experience", The Althouse Press, Ontario, 1990.

[13] Merleau-Ponty M., "Phenomonology of Perception", Routledge, London, 1992.

[14] Coyne R., "Designing information technology in the postmodern age", The MIT Press, Cambridge MA, 1995.

[15] Varela F.J. Thompson E., Rosch E., "The embodied mind", MIT Press, Cambridge MA 1991.

[16] Gadamer H-G. "The relevance of the beautiful and other essays" Cambridge University Press, Cambrigde, 1986.

[17] Dorst, C.H., " Describing Design - a Comparison of Paradigms", thesis TUDelft, 1997.

[18] Cross N.G., Christiaans H.H.C.M. and Dorst C.H. (Eds), "Analysing Design Activity" , Wiley, Chichester, 1996.

[19] Valkenburg A.C., "The Reflective Practice in Product Design Teams", Thesis TU Delft, 2000.

[20] Dreyfus H.L., "What Computers still can't do", MIT Press, Cambridge MA, 1992.

[21] Winograd T., Flores F., "Understanding computers and cognition", Ablex, Norwood NJ, 1986.

[22] Frankenberger, E. and Badke-Schaub, P., "Modelling design processes in industry – empirical investigations of design work in practice", In: Akin, O. and Saglamer, G. (Eds), Proceedings of DMD'96, Istanbul, 1996.

[23] Van der Lugt R., "Sketching in idea generation meetings", thesis TUDelft, 2001.

For more information please contact:

Kees Dorst Department of Philosophy, Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven, The Netherlands tel +31402475541 Email: c.h.dorst@tm.tue.nl.