Proc. Int. East-West Conf. on IT in Design, Moscow, Sept'94

II. CONCEPTUAL DESIGN

КОНЦЕПТУАЛЬНОЕ ПРОЕКТИРОВАНИЕ

Requirements Identification: a Central Issue in Design Research

Amaresh Chakrabarti
Engineering Design Centre
Department of Engineering, Trumpington Street
Cambridge CB2 1PZ, UK

ABSTRACT

Establishing the requirements is as important as solving for them. The main emphasis of today's design research is in producing tools that aid in problem solving, and rarely so in establishing the problem. Consequently, clarification of the task, as this activity is termed by the systematic design research community, is the most ill-defined of all design activities. The long-term goal of the project reported in this paper is to support designers in producing design requirements more reliably than presently possible. Central to this is an understanding of what is meant by a design problem, what the usual problem-types are, and possible mechanisms by which these problems are identified. In this paper, a definition of a design problem is proposed (in terms of a transformation between the expected course of history without the design and the desired course brought about by the design), a number of possible problem-types and their probable identification-mechanisms are illustrated, a number of possible design-aids have been proposed, and issues that need be resolved to develop these are discussed.

MOTIVATIONS, BACKGROUND AND SCOPE OF THE PAPER

Requirements identification is a central issue throughout the design process. Designing begins with a discontent. This could be a dissatisfaction about the way certain things work presently, thereby a wish to change this for the better. Alternatively, this could be satisfaction about the way things presently are, and a wish to preserve this by restricting the tendencies leading to their change [Gasparski, 1992]. Humans act based on a value system. There are various factors which contribute to producing the value system through which a given situation is perceived as unacceptable [see Praxiology in design, esp. Gasparski, 1992, for details]. There are other belief factors which decide what is achievable as an acceptable situation [see Vincenti, 1990; Laudan, 1984].

Given that a discontent exists (for whatever reasons), one must clarify what it means in terms of requirements or task. This process of clarification and establishment of requirements, though principally done at the beginning of the design process, spans over the entire design activity, and has feedbacks from more detailed stages leading to modification of the requirements. What underlies this process is the process of *identification of problems* (with the existing situation, product, methods etc.), which leads to the establishment of requirements as a task of circumventing or solving the problem.

The problem identification, therefore, is as essential as problem solving. However, the main thrust of present design research, with few exceptions [Iwata & Onosato, 1989], is in producing tools and techniques that aid in problem solving, rather than establishing the problem. Consequences include: (i) ad hoc attempts to produce design requirements which might have little or no clear relationship with the higher level requirements from which they should have ensued in the first place- one consequence of this is the danger of solving non-problems; (ii) a skewed view of the design process, giving the impression that if 'somehow' a design problem could be established, the rest of the design is 'merely' a problem-solving activity.

Problem identification process is ill-defined, and, as will be argued in this paper, is hard. It is therefore an important part of design research to develop tools and techniques to aid this process. Central to this is an understanding of what is meant by design problems, and of possible mechanisms by which these are identified. In this paper, a definition of a design problem is proposed; this is defined as a transformation between (i) the changes expected in a given environment without the desired design, and (ii) the changes desired to take place with the introduction of the desired design. Following this, a number of problem-types and their possible identification mechanisms are proposed and illustrated; these are: (i) action identification problems, (ii) transformation problems, (iii) implementation problems, (iv) functional problems, (v) side-effects and additional problems, (vi) Analogous problems, and (vii) mistakes. Finally, possible supports for problem identification are proposed, and issues are discussed which need to be resolved, as a pre-requisite to this.

DESIGN PROBLEM AND ITS CLARIFICATION

What is a design problem? Before we describe this, we need to define a set of concepts that would be used in the following discussion. These are:

Object: (mental models of) a physical thing; these could be described in terms of a set of attributes, or lower-level objects and their relationships. In fig.1a, there are two objects: a support, and a abstract object *ground* (which is immovable).

Parameter: attributes such as colour, density, shape, velocity, etc., used to describe (the state of) things. In fig.1, for instance, the support could be described as a horizontal plane of finite dimensions.

Relationship: describes relationships between objects, such as the spatial relationships between various members of a design. In fig.1a, there is a fixed-connection between the two objects, which implies that they can only move together.

Situation: a set of objects, relationships, and (potential or active) processes with unique values of their parameters. For the case in fig.1a and 1b, this includes the objects, their relationships, and equilibrium as the active process as a result of a balance between the weight of the support and the reaction to this force from the ground (see situation-1 in fig.1c).

Processes: those which change a situation: could be described in terms of the action, input situation(s), and output situation(s); for instance, if an unbalanced force acts on an object, the object's state of velocity changes.

Action: changes introduced into a situation; this could be in terms of changes in the objects (including the introduction of new objects), or changing relationships between objects. If a block is introduced into the above situation such that a touch-connection is established between the support and the block (see fig.1b), this action will activate the weight (gravity process) to act on the support, leading to the activation of reaction process as a consequence; these two forces then activate the process of equilibrium, thereby allowing the situation to be as in fig.1b with the three objects, their two relationships with the whole system as static.

Scenario: An ordered set of situations. Fig.1c describes the scenario for transition from situation-1 (in fig.1a) to situation-2 (in fig.1b).

Essentially, a designer has a mental model of what the present situation is, and how this would change (including the scenario where the situation does not change), given a set of assumed actions. One could call this the *otherwise-expected* scenario (the scenario without the design). The designer also has a picture of a desired situation or situations, which may or may not be the initial situation of the otherwise-expected scenario, but certainly is not the final situation in the otherwise-expected scenario. Let us call the scenario, initially defined as the transition between the initial situation (same as that of the otherwise-expected scenario) and the desired situation, as the *desired* scenario. A design problem is defined here as the transformation between these two scenarios, see fig.2. The outcome of the design process is a body of information, possibly validated through experiments, which can be used to describe the actions, objects and relationships, involved in the desired scenario, in sufficient degree of detail so as to warrant (i.e., justify and enable) the later processes such as manufacturing to take place. Clarification involves processes by which one could (i) identify the situations in the desired scenario as well as in the otherwise-expected scenario, and (2) identify and specify the actions involved, for a design problem. Let us take the example of the design of a means by

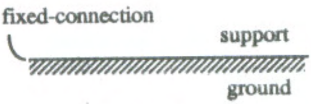

fig.1a Situation-1

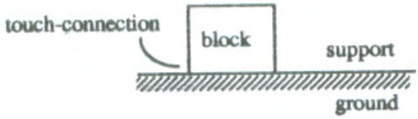

fig.1b Situation-2

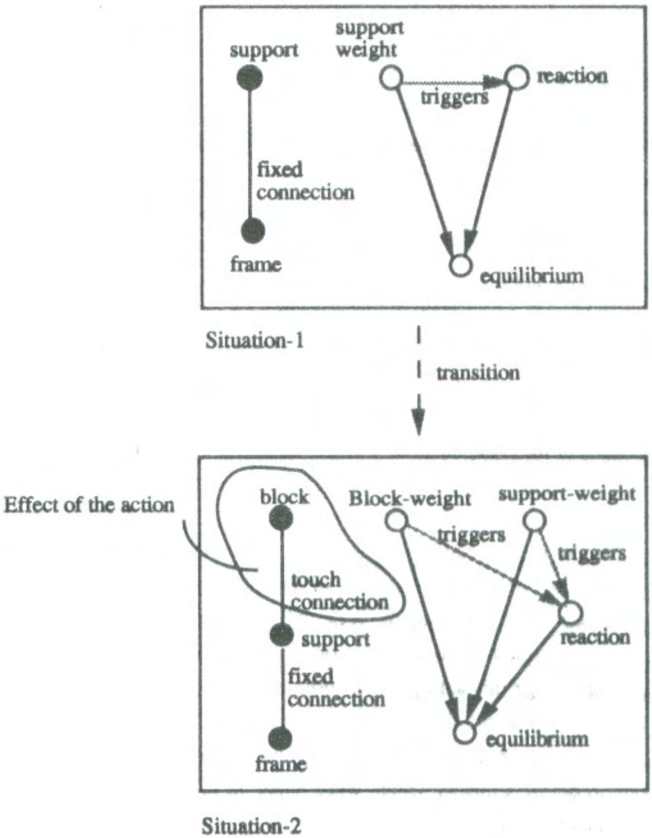

fig.1c Scenario as a transition from situations 1 to 2

Fig.1 An example for actions, situations and scenarios which unwanted people could be prevented from entering a specified area. what is meant by "entering", or "unwanted" people? One immediately begins to see that these need clarification, and often specification. If the "specified area" is a piece of land as part of a larger land mass, the immediate mode of action for "entering" seems to be walking or running. In such a case, walking and running could be taken as clarification of the action of "entering". However, if a wall is used to prevent "entering" by walking, these "unwanted" people might aspire to "enter" by climbing the wall! In the context of the wall solution, then, "entering" might have to be clarified as using the actions of walking, running as well as climbing. Thus, clarification of the requirements involves understanding or defining high level requirements (such as "entering") in terms of lower-level requirements (such as walking, climbing, etc.), in the context of given situations (in this case, where a specified area, a wall and certain relationships between them are involved), and this is sometimes impossible without the assumption, or the context, of specific solutions.

PROPOSED PROCESSES OF PROBLEM IDENTIFICATION

What are the possible ways by which design problems are identified? In order to explore them, let us follow through a possible scenario that a designer might have to go through even before he can start considering the possibility of designing a bottle-opener such as a cork-screw. The design problem may be initially defined in terms of the desired scenario shown in fig.3. The goal is to transfer wine from the bottle to another container (which could be one's mouth). The initial situation could be defined in terms of the initial state of the bottle (say a wine bottle with cork) and its surrounding: the objects are the bottle, cork, wine, table and ground. The bottle has different functional surfaces such as bottom, neck (outside and inside), etc.; the other objects could be defined similarly. The bottle is connected via the inside of its neck to the (cylindrical) outside surface of the cork; the connection is a friction connection. Similarly, the table is connected (i) to the bottom of the bottle via touch connection, and (ii) to the ground via touch connection at its legs. The potential or active physical processes include friction, gravity, reaction, equilibrium, etc., and it is possible to produce a graph showing the processes at work. For instance, at the cork-bottle interface, the weight of the cork is counteracted by friction, producing equilibrium, thereby leaving the cork without motion. In order to define the desired situation, we have to define the modes of action by which to transfer wine; this then brings us into asking what change this action would bring into the present situation. If the actions are assumed to be first "bringing the bottle above the container" and then "turning the bottle around", and we concentrate on the latter action, we find that wine does not come out of the bottle; we thus want to devise a means which, when acted on the initial situation, would bring it to a state where the above action would allow for wine to come out of the bottle. As one might have noticed, the requirements have already begun to be clarified in the above case. The process involves comparing the desired scenario with the present scenario.

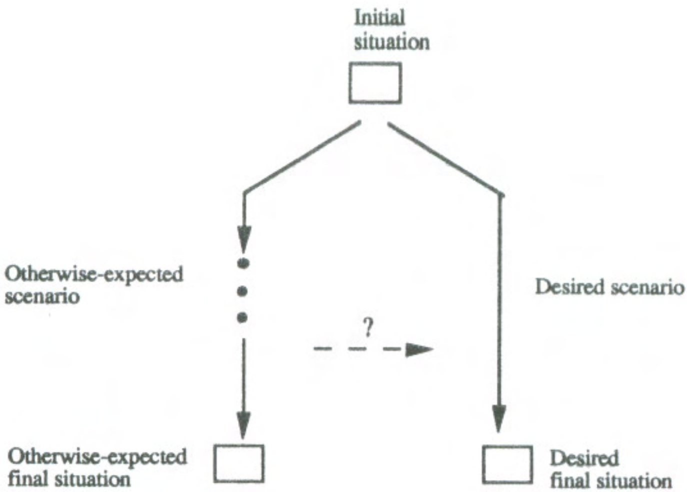

Fig.2 A proposed definition of a design problem

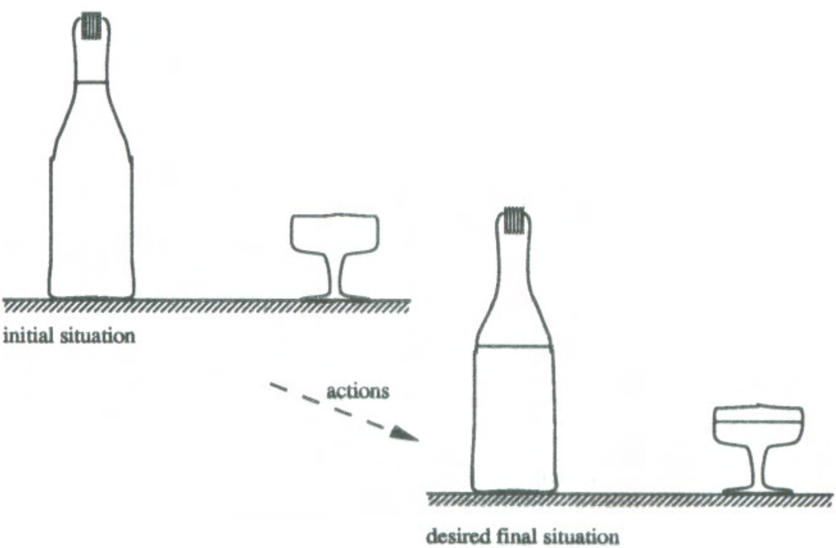

Fig.3 A desired scenario for the wine-transfer problem

The process of this comparison, however, is difficult. Firstly, the desired scenario is at most only incompletely known in terms of its objects and their relationships (particularly the design and how it relates to the rest of the situation, which we might call the context). Further, the actions intended to use the design in the context, and other actions such as "turning the bottle around" are design decisions and therefore assumptions*. Similarly, the otherwise-expected scenario also is at most incompletely defined, although less so than the desired scenario.

As we have seen above, figuring out possible actions, other than the ones a designer has started with, can sometimes be making assumptions. Another possible way is what was described in the problem discussed before regarding the clarification of the modes of "entering", by "unwanted people" into a specified area. It is interesting to note that although one could easily clarify the simplest modes of "entering" as walking and running (which might be the result of the knowledge of the existing modes of entering used in the given area), the modes such as "climbing over the wall", or "making holes in the wall" could come only in the context of a wall as the initial solution. In other words, without the assumption of a solution, some modes of action may not be visualised at all. These problems are termed here collectively as action identification problem, and is summarised below:

Action identification problem: Involves figuring out possible modes of action within the scenarios other than the given ones. Some of these involve making assumptions necessitating validation at later stages, while others can be conceived only in context of specific solutions.

As far as the above wine-flow problem is concerned, we are at a situation where wine flow is desired, and wine non-flow is the otherwise-expected scenario, given the assumption about the mode of action mentioned before. The next question to ask is perhaps what contributes to the non-flow. As flow requires non-equilibrium, and non-flow implies equilibrium, our desired state requires unbalanced forces on the fluid. What, then, are the forces, at present, contributing to the equilibrium? Can any of these be changed? These questions may lead to the understanding that the forces restricting are friction between, and structural containment by, the cork and the bottle; this in turn could lead to lower-level requirements: such as finding means for breaking (parts of) the bottle, breaking/penetrating the cork, or separating the cork from the bottle (overcoming friction)**. This mode of requirements identification is termed here as transformation problem, and summarised below:

---

* Each such assumption immediately restricts the kinds of solutions which could now be considered. For instance, the action of turning the bottle around restricts the possibility of considering breaking the bottle after solidifying the wine, whereafter taking the solid-wine out before heating it to the initial state. Although this solution is not a viable one, these assumptions do not necessarily restrict only the non-viable ones.

** One could go to a higher level of abstraction and consider the problem like this. The desired scenario is to establish a connection between wine and outside, while the otherwise-expected scenario is that these are now separated from each other by the cork and the bottle. Then looking at the possible ways of removing this separation, one could think of three principles: (i) removing (part of) the bottle, (ii) removing part of the cork, or (iii) removing the interface between cork and bottle. This, then could follow into the details described above.

Transformation problem: involves comparing the desired with the other-wise expected scenario, and identifying the reasons which might have led to the undesired, and whose change would bring about the desired scenario.

Now, assuming that the designer has decided to remove the cork from the bottle, that this means overcoming friction between the bottle-neck and the cork by either (i) adding extra force to the already existing gravity in terms of the weight of the cork and wine-mass combined, or (ii) reducing the friction by some means. Considering the first option as an example, the designer might look for ways by which this could be done. This can be at various levels, from the conceptual phase to the detailed stages where problems such as manufacturability is examined. This is what is termed an implementation problem here, and is summarised below:

Implementation problem: involves finding and detailing the potential solutions (in this case, how do you increase force?).

Designers often generate potential solutions which solve only part of the function; this partial solution is then modelled and simulated mentally, computationally or physically, against the full functional specification available at that stage, whereby it is checked whether or not the design is functionally satisfactory. Detection of these problems in a design is termed here as *functional problems*.

Four possible solutions to the problem of designing a means for overcoming friction between the bottle-neck and the cork are the use of: (i) a hydraulic/pneumatic force from inside, (ii) a hydraulic/pneumatic force from outside, (iii) a mechanical force from inside, or (iv) a mechanical force from outside. Supposing that increasing pressure inside the bottle is taken as the solution, two of its side effects might be that the (i) cork shoots off, or (ii) the bottle breaks, causing probable injuries. This could lead to problem identification of two kinds: (i) discarding this possibility at the higher level, or (ii) taking this as the new low-level requirement, leading to the re-definition of the problem at the higher level as removing the cork at a low velocity without injuring the bottle. This mode of problem identification is described here as side effects and additional difficulties, and summarised below:

Side effects and additional problems: involves identifying new problems which a solution to a previous problem might give rise to, as a side effect; this could also involve identifying additional problems, initially not part of the original problem, which the solution solves as a side effect.

The other two modes of problem identification considered here are (i) *analogous problems*, and (ii) *finding mistakes* in the previously carried out problem identification activities. The first one involves considering (part of) the problems associated with a previously known (similar) design for the present design; the other ones is self-evident.

It is worth mentioning, as a separate note, that simulation plays an essential role in the detection of most of the problem-types discussed above, as do the assumptions made about the constituents of the scenarios under consideration. It is the correctness of the simulation as well as that of the assumptions which contribute to the reliability of the problem identification process.

DISCUSSION

As can be seen from the above, most of the above problem-types depend on the assumption of a solution. More importantly,

- Requirements depend on solutions, solutions depend on requirements.
- Assumptions are to be made; their correctness could be checked only at later stages.
- Simulations are to be performed, their correctness could be checked only at later stages.

Therefore, all these lead to inevitable loops, and problem identification is inherently hard. Supports are thus required. What would their goals be? The single top level goal is that the problems identified should improve the likeliness that the tasks represent the need. Asking right questions as soon as possible is important. But what are those? Design research could provide some, while others have to be found iteratively. In the first case, guidelines regarding standard mechanisms [Altschuller, 1984] of problem identification should be useful. To support the identification of iterative ones, a framework which could allow explicit connection between high and low level requirements, render awareness of assumptions (action, situations, processes), and allow systematic documentation of all these, should be useful as an intermediate step; a long term possibility would be to develop an intelligent support which could help extensively with the simulation task. There are three kinds of possible support that have been identified here. One immediate possibility is to develop or compile guidelines for clarification of the task and for asking right questions. Possible sources of such information can be found in systematic design literature [see Pahl & Beitz, 1984, for a good compilation]; one specific source is Aguirre-Esponda's compilation work on design guidelines [Aguirre-Esponda, 1992]. Another, longer-term, possibility is to develop a framework for systematic identification and connecting together of requirements. It is still largely unexplored. The other long-term possibility is the development of intelligent support systems which should not only do the book keeping, but also provide simulation and diagnostic supports so that some of the processes of problem-identification discussed in the previous section could be done more reliably. Work in progress in qualitative reasoning, in AI and design, especially process oriented representations [Forbus, 1984] could be used in this direction.

The immediate future extension of the work reported in this paper would be an extensive validation of the proposed modes of problem identification using real-life protocol data, which has already been initiated.

ACKNOWLEDGEMENTS

Lucienne Blessing helped in correcting the manuscript and gave helpful suggestions for its improvement. The concept of wine transfer by the sequence of freezing-wine-and-then-breaking-glass-and-then-heating-wine-into-liquid-state came out of a discussion with Dr. Thomas Bligh.

REFERENCES

Aguirre-Esponda, G. (1992) Evaluation of Technical Systems at the Design stage. Cambridge University.

Altschuller, G. S. (1984). Erfinden: Wege zur Loesung technischer Probleme. VEB Verlag Technik Berlin.

Forbus, K. (1984). Qualitative Process Theory. Artificial Intelligence, 24, 85-168.

Gasparski, W. W. (1992). Design, Science, and Philosophy: The Praxiological Perspective. In NATO Advanced Workshop on Design Methodology, Eindhoven, The Netherlands.

Iwata, K., & Onosato, M. (1989). A computer aided conceptual design system of machine, Annals of the CIRP, Switzerland, Vol 38 , pp 137-140.

Iwata, M., & Morita, C. (1985). An available cad-technology for creative design. Iced 85, Hamburg, West Germany, Vol 1 , pp 644-660 .

Laudan, R. (1984). Cognitive Change in Technology and Science. In R. Laudan (Ed.), The Nature of Technical Knowledge: Are Models of Scientific Change Relevant?, (pp. 83-104). Dordrecht.

Pahl, G., and Beitz, W. (1984). Engineering Design (Wallace, K., Trans.). London: The Design Council.

Vincenti, W. G. (1990). What Engineers Know and How They Know It. The Johns Hopkins University Press.

* * *