Design
knowledge modeling - Knowledge modeling as design

Willemien Visser

Ce texte est un pre-print de
W. Visser (1993). Design & knowledge modeling - Knowledge modeling as design. *CCAI, Communication and Cognition - Artificial Intelligence* [today entitled "CC-AI, The journal for the integrated study of Artificial Intelligence, Cognitive Science and Applied Epistemology"], 10(3), 219-233.

Design & knowledge modeling - Knowledge modeling as design

Willemien Visser

Institut National de Recherche en Informatique et en Automatique
Projet de Psychologie Ergonomique pour l'Informatique
Rocquencourt B.P.105
78105 LE CHESNAY CEDEX (FRANCE)

email: willemien.visser@telecom-paristech.fr

September 1992

Abstract. The paper presents a cognitive analysis of design as a problem-solving activity. Design is characterized through results from empirical studies of various design tasks. Based on this characterization, it will be argued why and how knowledge modeling can be considered as a design task.
Knowledge modeling has been qualified as the "construction of a set of successive representations". Processing different types of representations at different levels and from different viewpoints plays an important role in design problem solving. It will be argued that other characteristics of design may also apply to knowledge modeling. As there are not yet data available on the cognitive activities involved in knowledge modeling, hypotheses for studying these activities may be inspired by what is known about design: design problems have no pre-existing, complete problem definition, and no unique, correct solution; the design activity does not follow a pre-existing plan, it is opportunistically organized.
Key-words. Design, Knowledge modeling, Problem solving, Ill-defined problem, Cognitive modeling, Representations, Empirical studies.

Résumé. L'article présente une analyse cognitive de la conception comme activité de résolution de problème. La conception est caractérisée à travers les résultats d'études empiriques conduites sur différentes tâches de conception. Sur la base sur cette caractérissation, on présente des arguments pour défendre pourquoi et comment la modélisation de connaissances peut être considérée comme une activité de conception.
La modélisation de connaissances a été qualifiée de "construction d'un ensemble de représentations successives". Le traitement de différents types de représentation de niveaux et de points de vue différents joue un rôle important dans la résolution de problèmes de conception. On défend que d'autres caractéristiques de l'activité de conception peuvent s'appliquer également à la modélisation de connaissances. Parce que l'on ne dispose pas encore de données empiriques sur les activités cognitives mises en oeuvre dans cette modélisation, on peut formuler des hypothèses à ce sujet à partir de ce que l'on sait sur la conception: l'absence d'une définition exhaustive, pré-existante du problème et d'une solution unique ou "correcte", et le fait que l'activité ne suit pas de plan pré-établi, mais est organisé de façon opportuniste.
Mots-clés. Conception, Modélisation de connaissances, Résolution de problème, Problèmes mal structurés, Modélisation cognitive, Représentation, Etudes empiriques.

Ce texte est un pre-print de
W. Visser (1993). Design & knowledge modeling - Knowledge modeling as design. *CCAI, Communication and Cognition - Artificial Intelligence* [today entitled "CC-AI, The journal for the integrated study of Artificial Intelligence, Cognitive Science and Applied Epistemology"], 10(3), 219-233.

1. Introduction

Knowledge modeling may be conceived as (a special case of) design. Not only because it involves design of a system, i.e. a knowledge system, but especially because modeling is an ill-defined problem-solving task -and one may suppose that this "ill-defined-ness" is not due to the novelty of the task domain, but is an intrinsic characteristic of the task.

Knowledge modeling has been qualified as the "construction of a set of successive representations" (Duverger, 1991). Processing different types of representations at different levels and from different viewpoints is considered to be an important activity involved in design. This paper presents a cognitive analysis of design as a problem-solving activity. Design will be characterized through the results from empirical studies conducted on various design tasks. Based on this characterization, it will be argued in the conclusion why and how knowledge modeling can be considered as a design task.

In what follows, we first present some approaches adopted to design through history (Section 2) and some theories and definitions proposed by various authors to characterize design (Section 3). Next, section 4 adopts the analysis of design as a problem-solving task and presents a characterization of design developed within this framework. In sections 5, 6 and 7, results from several empirical studies on design in different domains are presented, each one illustrating one of the three main characteristics of the design activity briefly presented in section 3. Section 8 then closes with a discussion of some consequences of these results for knowledge modeling as a design task.

2. Different approaches to design through history

In 1984, Cross has brought together a series of design studies from the period between 1962 (year of the first "Conference on Design Methods", the birthdate of design methodology) and 1982 (the "Design Policy Conference", the date of coming of age of design methodology) (see Cross, 1984a). The organization of his book reflects the progression of the "movement" in this domain through four stages:
• prescription of an ideal design process (Part One - The Management of design process; papers published between 1962 and 1967, the period of "systematic design");
• description of the intrinsic nature of the design activity (Part Two - The Structure of design problems; 1966-1973, the period when design problems were discovered to be not so amenable to systematization);
• observation of the reality of the design activity (Part Three - The Nature of design activity; a set of papers all, coincidentally, published in 1979);
• reflection on the fundamental concepts of design (Part Four - The Philosophy of design method; 1972-82).
The present paper focuses on studies describing and modeling "the intrinsic nature of the design activity", based on observations conducted on real design tasks (it covers of course more recent studies than those presented in Cross, 1984a).

Visser (1994), focusing on the research conducted on the organization of the activity in design tasks, also identified four stages in the history of research into this aspect of design: in the first stage, the terms “hierarchy” or “opportunism” are not relevant, as they had not yet been used; subsequently, several studies concluded that design was organized in a hierarchical way; in the third stage, the opportunistic character of design was shown; and recently, several researchers have started to

Ce texte est un pre-print de
W. Visser (1993). Design & knowledge modeling - Knowledge modeling as design. *CCAI, Communication and Cognition - Artificial Intelligence* [today entitled "CC-AI, The journal for the integrated study of Artificial Intelligence, Cognitive Science and Applied Epistemology"], 10(3), 219-233.

investigate more closely the “conflict” between “hierarchy” and “opportunism” (see, below, the section on the planning of the design activity). Visser (1994), staying within the domain of empirical design studies, analyzes this development through four stages as a "progression" from modeling design problem solving in restricted -generally experimental, laboratory- conditions towards models of design in real-life tasks.

3. Theories and definitions of design

Simon (1969/1981), one of the first authors to formulate a general theory of design, considers design as "the process of devising artifacts to attain goals". Reitman (1964), another "early" author theorizing on design problems, focuses on problems involving the "transformation or creation of states, objects or collections of objects" (represented as a three-component vector [A, B, ->]), and distinguishes six different types of these problems, according to the specification of A, B and ->: they can be given or not and, if given, more or less well specified. Design (or its form which Reitman calls a "problem of creation, invention, or discovering") is characterized by A and -> not being specified, and B being specified at an abstract level (function + constraints).

In the domain of knowledge-based systems, various authors have proposed categories of problem-solving tasks and define design through a confrontation with other types of problems or problem solving.
Generally, these categories have several levels. Hayes-Roth, Hayes-Roth, Rosenschein and Cammarata (1979), e.g., from a viewpoint of cognitive modeling and Artificial Intelligence, distinguish interpretation and generation problems. The famous Hearsay-II framework, originally developed for interpretation problems, is applied in Hayes-Roth et al. (1979) to a generation problem, i.e. planning (interpreted in Visser, 1994, as design of plans). Despite important differences between the two types of problems, the framework "appears robust enough to guide solution of both interpretation and generation problems" (Hayes-Roth et al., 1979, p. 382).
Generation problems are "problems which present the highest level representation (e.g., the goal) and require generation of the lowest level representation (e.g., the sequence of intended actions)" (ibid.). "Interpretation and generation problems differ in important ways. For example, interpretation problems lend themselves well to initial bottom-up strategies, while generation problems lend themselves well to initial top-down strategies. Interpretation problems generally permit only one (or a small number) of solutions, while generation problems permit an arbitrary number of different solutions. Further, interpretation problems typically have correct solutions, while the correctness of solutions to generation problems varies under different evaluation criteria." (ibid.)
Without reference to these two broad categories, Hayes-Roth, Waterman and Lenat (1983) distinguish the following types of tasks (for expert systems): interpretation, prediction, diagnosis, design, planning, monitoring, debugging, repair, instruction, and control. Design is defined as the configuration of objects under constraints. In the context of the interpretation - generation distinction, it surely would be considered a generation problem.

Several authors have formulated critiques on the list of tasks given by Hayes-Roth et al. (1983), especially because the different tasks are not all at the same level. Clancey (1985) introduces "system" as a central concept: systems can be characterized in terms of their inputs and outputs. Clancey proposes two large sets of "generic operations" "[doing things] to or with a system": they can construct

Ce texte est un pre-print de
W. Visser (1993). Design & knowledge modeling - Knowledge modeling as design. *CCAI, Communication and Cognition - Artificial Intelligence* [today entitled "CC-AI, The journal for the integrated study of Artificial Intelligence, Cognitive Science and Applied Epistemology"], 10(3), 219-233.

a new system (synthesis) or interpret an existing system (analysis). Design is a construction task and Clancey distinguishes two types of design task, configuration (characterizing a structure) and planning (characterizing a process).

In the KADS knowledge elicitation methodology (see Breuker, Wielinga, Someren, Hoog, Schreiber, Greef, Bredeweg, Wielemaker, Billeaut, Davoodi and Hayward, 1987), the same top-level distinction is used: system synthesis and system analysis are the major classes in problem-solving tasks. The authors propose modification tasks as a transition area between these two types of tasks. As in Clancey (1983), design is conceived as a synthesis task. Here, four types of design are distinguished: hierarchical design, transformational design, incremental design and multi-stream design. For all synthesis tasks, the authors adopt "a 2-stage model in which first the informal problem statement is analysed resulting in a formal specification of the structure to be synthesised, and a subsequent stage in which the structure is actually created" (p. 91). This decomposition in two distinct, consecutive stages will be commented on in the next section in a discussion of Simon's (1973/1984) approach to design (but notice the following remarks on generic task models).

As recalled by Wilson (1989), "task analysis techniques [adopted by the authors using generic task models for their data collection] divide tasks into components which are usually of pre-specified types. It is this pre-specification of types of components and the relationships those types can hold which differentiates descriptions resulting from the use of task analysis techniques from informal descriptions of tasks in natural language." (p. 202) The analysis presented below yields such "informal descriptions … in natural language" of design tasks. However, the aspect of design on which these descriptions differ most from the analyses based on task analysis techniques, i.e. the organization of the activity, has led to several formal descriptions and (proposals for) design models (see Davies, 1991; Rist, 1991; Visser, 1990). One of the causes of this difference is the use of interviews which is one of the most frequent knowledge elicitation techniques employed in task analysis (see also the section on Empirical design studies in Visser, 1992). Interviews are indeed known to be rather inappropriate for the acquisition of knowledge on (expert) strategies which is -relatively best- obtained by protocol analysis techniques (cf. Cordingley, 1989; Nwana, 1992).
One should remember, however, that the task performed by a knowledge-based system (KBS) is not the same as that performed by the expert whose knowledge has been used to design this system! The knowledge engineer indeed models the knowledge of the expert with a particular objective -the design of a KBS, he does not pretend to "simulate the mind of the expert" (Duverger, 1991).

4. Design and other types of problem solving

This paper presents a cognitive-psychology approach to design. In cognitive psychology, most authors have also characterized design by considering it as a problem-solving activity and comparing it with other types of problem solving.
Hoc (1988) presents a distinction between state transformation, structure induction and design problems: design consists in detailing a laconic, insufficiently detailed and inappropriate representation of the goal state into an appropriate representation.
Various authors have opposed "ill-defined" (or "ill-structured") and "well-defined" (or "well-structured") problems: design problems are of the first type, whereas the problems classically examined in psychological studies are of the second type.

Ce texte est un pre-print de
W. Visser (1993). Design & knowledge modeling - Knowledge modeling as design. *CCAI, Communication and Cognition - Artificial Intelligence* [today entitled "CC-AI, The journal for the integrated study of Artificial Intelligence, Cognitive Science and Applied Epistemology"], 10(3), 219-233.

Both Simon (1969/1981) and Reitman (1964) use this distinction (see also Eastman, 1969; Voss and Post, 1988). Reitman (1964) considers that there is a continuum ranging from well-defined formal problems to ill-defined empirical problems such as the design problem of composing a fugue, analyzed by him in great detail (cf. Reitman, 1965). He relates this continuum to the idea of ambiguity of a problem (i.e. the fact that, over a specified community of problem-solvers, there is little or no agreement regarding the referents of the problem attributes, permissible operations, and their consequences). A consequence of this is that solutions to ill-defined problems are not correct or incorrect, but more or less accepted.

Thomas and Carroll (1979/1984) propose to consider designing, not as a form of problem solving particular to "the class" of "ill-defined problems", but rather as a way of "looking at" a problem. A problem is not defined by its "objective" characteristics, but depends on its representation constructed by the problem solver. As put forward in Visser (1994), "rather than characterizing a particular task as a 'problem', independently of a subject (designer) or other cognitive system confronted with this task, we consider that a problem only exists relative to a subject (see Hoc, 1988; Mayer, 1989). Defining 'a situation [as] a functional system constituted by a task and a subject' (Leplat and Hoc, 1983, p. 49), a situation constitutes a 'problem' -or not- for a particular subject depending on the representation this subject constructs of her or his task."
This position could be judged similar to Simon's (1973/1984) analysis of the "elusiveness" of a problem's structure: "definiteness of problem structure is largely an illusion that arises when we systematically confound the idealized problem that is presented to an idealized (and unlimitedly powerful) problem-solver with the actual problem that is attacked by a problem-solver with limited (even if large) computational capacities" (p. 150). Simon considers, however, that -in a first stage- an ill-structured problem can be structured, and that -in a second stage- solving the resulting well-structured problem requires only a fraction of problem-solving effort. The idea defended here is that there are not two consecutive stages: first structuring the problem, and then solving it, but that -if two such processes can be distinguished- they are intertwined during the entire problem-solving process until a final solution is proposed.

The presentation given below of design problems and design problem solving should therefore not be interpreted as reflecting the position that design problems can be characterized independently of the activity of their solver. It simply focuses consecutively on only analytically distinguishable aspects: first on static, representational, next on dynamic, processing aspects.

A design problem, as it is considered by the designer,
• is not defined in the initial problem specifications, i.e. is not specified completely and unambiguously right from the start, when the designer receives the specifications (from the client) (cf. the definition given by Hoc, 1988);
• can be solved, more or less satisfactorily, by several solutions, rather than by just one "correct" solution: there is no "definite criterion for testing any proposed solution" (Simon, 1973/1984, p. 146), such as there typically exists for "well-defined" problems: design problem solutions are more or less "acceptable" or "satisfying", they are not either "correct" or "incorrect".
In an analysis of problems in the social sciences, Voss, Greene, Post and Penner (1983) attribute this characteristic to the fact that "the solvers of the community are not in agreement with respect to the

Ce texte est un pre-print de
W. Visser (1993). Design & knowledge modeling - Knowledge modeling as design. *CCAI, Communication and Cognition - Artificial Intelligence* [today entitled "CC-AI, The journal for the integrated study of Artificial Intelligence, Cognitive Science and Applied Epistemology"], 10(3), 219-233.

appropriate solutions" (p. 263) (cf. Reitman's ideas, presented above, on the varying ambiguity of problems). Voss et al. (1983) refer to Tweney (1981) for "evidence [suggesting] that the solving of [the physics problems in new research areas] highly resembles the solving of the political science problems" (ibid.). Due to this incomplete and ambiguous character of the problem requirements, the solution evaluation criteria are also ill-defined.

Designers, in order to solve a design problem,
• have to specify, and possibly modify, the problem constraints in the problem specifications; if -as is often the case- there are conflicting constraints in the initial problem specifications, these conflicts are to be resolved;
• need to construct and use representations of the "same" object at different abstraction levels and considered from different viewpoints (e.g., conceptual, functional, physical);
• need to possess -because they need to use- knowledge in different domains (e.g., design methods; knowledge in the "application" domain, such as mechanics or aerospace structures in composite material);
• cannot follow, in general, a pre-established plan or procedure (such as, to process the different levels or viewpoints in a particular order: for example, top-down, breadth-first, first the conceptual, then the functional, and finally the physical viewpoint).

To sum up: design tasks may be characterized by the following qualifications of the problem, the activity to solve it, and the solution to be developed:
• the problem to be solved is not completely and immutably specified from the outset: an important part of the engineer's activity consists of constraining this problem;
• this activity does not follow a pre-existing plan;
• the solution to be obtained is not unique: the "final" specifications arrived at by a designer may be modified, not because they are "incorrect", but because an alternative design may be chosen by another designer, using different criteria (cf., for these three characteristics, the Hayes-Roth et al., 1979, definition of generation problems quoted above).
In the last section of this paper, these characteristics of design will be used in the discussion of knowledge modeling as a design task. The following section will detail them through the examples of empirical studies conducted on different design tasks.

5. Evolution of the design representation throughout the problem-solving process
Solving a design problem consists in transforming one problem representation into another, generally via intermediate representations.

First, design always starts with "requirements" and produces "specifications" (see Visser, 1990). Differences of, at least, two types characterize these two problem representations:
• domain (or viewpoint): the representation domains of both representations differ;
• level: the first representation is at a more abstract level than the second (i.e. is less complete, less precise and less concrete than the second).
In software design, e.g., a specifications representation using concepts in the application domain is to be transformed into computer specifications represented in the software domain. In mechanical design,

Ce texte est un pre-print de
W. Visser (1993). Design & knowledge modeling - Knowledge modeling as design. *CCAI, Communication and Cognition - Artificial Intelligence* [today entitled "CC-AI, The journal for the integrated study of Artificial Intelligence, Cognitive Science and Applied Epistemology"], 10(3), 219-233.

going from the client's requirements to the working drawings, three consecutive representation stages may be distinguished, which are, however, not necessarily handled in this order (see below):
• representation of the goal to be attained by the artifact to be designed (or its function);
• representation of the mechanical operations which will allow this goal to be attained (to realize this function);
• representation of the machines or tools which are going to physically bring about these operations.
For each couple of consecutive representations, the first defines the requirements for the second (cf. Dasgupta, 1989, pp. 48-49).

The client's requirements, i.e. the specifications of a design problem presented at the outset, are rarely complete and/or stable. Generally client and designer(s) discuss them to arrive at a specification worked out in common (cf. Malhotra, Thomas, Carroll and Miller, 1980); but even the problem as defined in the legal documents resulting from this is subject to modifications: an important part of a designer's activity consists of developing, especially constraining, the problem.

Levels and/or viewpoints of representation. First, the design is developed in greater and greater detail: first defined in terms of a global goal to be achieved, it is progressively characterized to become sufficiently precise as to specify the realization of the target object.
Second, the design is developed from the viewpoint of different types of representation which are all to be taken into account in the design: e.g., functional, structural, topological, material / physical, financial. In a detailed analysis of protocols from computer network designers, Darses (1992) distinguishes five different viewpoints (types of constraints) from which the network problem is analyzed by the designers: initial required data, abstracted data, functional data, conceptual data and physical data. A designer does not necessarily analyze all problem components (or sub-problems) from all different viewpoints. Neither does the analysis of a problem pass in a fixed order through the different viewpoints -that is why a qualification in terms of "levels" does not seem appropriate here.
Darses (1992) describes the network design activity in terms of constraint satisfaction. Constraints are specified and developed. Darses presents several cognitive mechanisms used by designers for constraint management: constraints are propagated through the different viewpoints presented above; the propagation network which may become enormous can be reduced by compacting the constraints; plans, schemata and prototypes are proposed as knowledge representation formats which contain information used in marking out the direction adopted by constraint propagation.

6. The activity does not follow a pre-existing plan: design is opportunistically organized
Early design studies tended to "conflate prescriptive and descriptive remarks" on the activity, and, rather than to consider what the activity is really like, to focus on what it should be (see Carroll and Rosson, 1985; Cross 1984b). This led authors to describe design as being well structured and even as hierarchically organized. The design process as implemented by human designers, however, does not have a neatly organized structure of consecutive stages, such as advocated or laid down by design methodologies (cf., for engineering design, the ASE-paradigm proposed by Jones, 1963/1984, and its corresponding form in software engineering, i.e. the waterfall model). For a critical review of such methods, see Carroll and Rosson (1985), Cross (1984b), Dasgupta (1989, pp. 31-33), Visser and Hoc (1990).

Ce texte est un pre-print de
W. Visser (1993). Design & knowledge modeling - Knowledge modeling as design. *CCAI, Communication and Cognition - Artificial Intelligence* [today entitled "CC-AI, The journal for the integrated study of Artificial Intelligence, Cognitive Science and Applied Epistemology"], 10(3), 219-233.

Using data obtained for both individual designers and a group of designers working together on a design project, Visser (1988) argues why normative or prescriptive design models do not account for the human design activity (Visser, 1990, presents the detailed data for an individual mechanical designer working on a functional-specifications task).
As has been shown in empirical studies conducted on various types of design tasks, the design activity is quite different from the one presented in normative or descriptive theoretical models. It appears to be non-hierarchical: processing is neither completely top-down, nor totally bottom-up; it proceeds in a cyclical manner, introducing new goals as it goes along (see Visser, 1994).
Designers may describe their activity in the form of a hierarchically structured plan. This plan, however, does not represent the designer's actual activity which has, in fact, an opportunistic organization. Designers use such plans, but, as claimed by Visser (1994), even for experts involved in routine design, retrieval of preexisting plans does not appropriately characterize the organization of their activity. Underlying Visser's claim is the idea (based on data from various studies conducted on real design tasks reviewed in her paper) that designers may often find it profitable to deviate from these plans. Indeed,
• even if they can and, in fact, do retrieve a known solution plan for their design problem (which may be possible in routine design, as shown by several studies),
• yet if they evaluate the cost of their possible actions (“cognitive” and other costs), as they will do in real design,
• the action selected for execution will often be an action other than the one proposed by the plan.
The hierarchical plan is -temporarily- abandoned as
• the design of components is interrupted before it is completed,
• addenda and other modifications are made on components that have already been dealt with,
• components are anticipated, as they are dealt with before "their turn (according to the plan)".
Modifications may range from correcting errors to challenging design decisions (see Visser, 1988, 1990).

7. Not one correct solution, but various more or less acceptable solutions

In a study of traffic-signal setting technicians, Bisseret, Figeac-Létang and Falzon (1988) observed six expert designers solving the same problem. Each produced a different solution. Not only were the final results different, also the approaches used to achieve these solutions were different.

Bonnardel (1991) studied cognitive aspects of the evaluation process in design of aerospace structures. In an experimental study, she asked designers -experts and novices- to evaluate different problem-solution pairs and to select for each problem the solution they judged the best. Not only did different designers select different solutions; each designer also chose different solutions (two or three rather than one): the final choice between these two or three solutions would depend, they said, upon the final importance of a particular criterion. The level of expertise had no effect on the designers' choice. The differences in choice depended on the different priority which each accorded to certain criteria related to a particular characteristic of the aerospace structure (its material).

8. Conclusion

In the introduction, knowledge modeling was presented as a type of design. We have characterized it as an ill-defined problem-solving task. The present paper has detailed a certain number of characteristics

Ce texte est un pre-print de
W. Visser (1993). Design & knowledge modeling - Knowledge modeling as design. *CCAI, Communication and Cognition - Artificial Intelligence* [today entitled "CC-AI, The journal for the integrated study of Artificial Intelligence, Cognitive Science and Applied Epistemology"], 10(3), 219-233.

of design: solving a design problem consists in transforming one problem representation into another, design problems do not have one, correct solution, and design is opportunistically organized. These three characteristics apply indeed to knowledge modeling.

The expert's knowledge structures which the designers of knowledge systems (knowledge engineers) take as the input for their models are not given and well-defined at the start. Guided by hypotheses about these structures, their organization and their use, a knowledge engineer will have to make choices and decisions -which may be different from those adopted by a colleague, without one or the other being "incorrect" or even "less correct".

Knowledge modeling involves the construction of a set of successive representations. The development of methodologies such as KADS or KAKE may provide unified frameworks guiding the construction of these different representations, but the nature and type of the representations (models) constructed will still differ between different designers.

Even if there are no empirical data with respect to the organization of the cognitive activity involved in knowledge modeling, one may suppose that the conclusion based on empirical studies conducted on other types of design tasks also applies to this task: an ill-defined problem solving activity such as knowledge modeling does not follow a pre-existing plan, it is opportunistically organized as are other design tasks. This conclusion may be particularly valid for the middle stages of the task -accepting the abstraction adopted with respect to the real activity if using the "stages" approach. Data gathering -the first stage- and the design and implementation of the system -the last stage- might perhaps be rather well structured and organized -at least on paper or in the methodology handbook (cf. the remarks on the conflation between prescriptive and descriptive approaches). The main part of the task, however, i.e. the creation of different models of the expert's knowledge structures, will remain, in essence, an ill-defined task which cannot proceed according to a pre-established plan.

References

Bisseret, A., Figeac-Létang, C., & Falzon, P. Modeling opportunistic reasonings: the cognitive activity of traffic signal setting technicians (Research Report N° 898). Rocquencourt: INRIA, 1988.

Bonnardel, N. L'évaluation et la sélection de solutions dans la résolution de problèmes de conception (Rapport de recherche INRIA n° 1531). Rocquencourt: INRIA, 1991.

Breuker, J., Wielinga, B., Someren, M. van, Hoog, R. de, Schreiber, G., Greef, P. de, Bredeweg, B., Wielemaker, J., Billeaut, J.-P., Davoodi, M., & Hayward, S. Model-driven knowledge acquisition interpretation models (Deliverable task A1, Esprit Project 1098). Amsterdam: University of Amsterdam, 1987.

Carroll, J. M., & Rosson, M. B. Usability specifications as a tool in iterative development. In H. Rex Hartson (Ed.), Advances in human-computer interaction (Vol. 1). Norwood, N.J.: Ablex, 1985.

Clancey, W. Heuristic classification. Artificial Intelligence, 1985, 27, 289-350.

Cordingley, E. S. Knowledge elicitation techniques for knowledge-based systems. In D. Diaper (Ed.), Knowledge elicitation. Principles, techniques and applications. New York: Horwood, 1989.

Cross, N. (Ed.). Developments in design methodology. Chichester: Wiley, 1984a.

Cross, N. Introduction to Part One - The Management of design process. In N. Cross (Ed.), Developments in design methodology. Chichester: Wiley, 1984b.

Darses, F. Mécanismes cognitifs de gestion de contraintes dans la résolution de problèmes de conception. Actes d'ERGO-IA'92, Biarritz (France), 7-9 octobre 1992. (The following technical report could serve as an English version of the presented ideas:

Ce texte est un pre-print de
W. Visser (1993). Design & knowledge modeling - Knowledge modeling as design. *CCAI, Communication and Cognition - Artificial Intelligence* [today entitled "CC-AI, The journal for the integrated study of Artificial Intelligence, Cognitive Science and Applied Epistemology"], 10(3), 219-233.

Balfroid, F., Darses, F., & Jouve, C. Knowledge acquisition (Deliverable d43, Project Esprit MMI2 #2474). Rocquencourt, France: INRIA, 1990.
Dasgupta, S. The structure of design processes. Advances in Computers, 1989, 28, 1-67.
Davies, S. P. Characterizing the program design activity: neither strictly top-down nor globally opportunistic. Behaviour & Information Technology, 1991, 10, 173-190.
Duverger, L. Modeling knowledge. Unpublished document, 1991.
Eastman, C. M. Cognitive processes and ill-defined problems: A case study from design. Proceedings of the First Joint International Conference on Artificial Intelligence. Washington, D.C., 1969.
Hayes-Roth, B., & Hayes-Roth, F., Rosenschein, S., & Cammarata, S. Modeling planning as an incremental, opportunistic process. Proceedings of the 6th International Joint Conference on Artificial Intelligence, Tokyo, 20-08-1979.
Hayes-Roth, F., Waterman, D. A., & Lenat, D. B. (Eds.), Building expert systems. Reading, MA: Addison-Wesley, 1983.
Hoc, J.M. Cognitive psychology of planning. London: Academic Press, 1988.
Jones, J. C. A method of systematic design. In N. Cross (Ed.), Developments in design methodology. Chichester: Wiley, 1984. (Originally published in Jones, J. C., & Thornley, D. (Eds.), Conference on design methods. Oxford: Pergamon, 1963.)
Leplat, J., & Hoc, J.M. Tâche et activité dans l'analyse psychologique des situations. Cahiers de Psychologie Cognitive, 1983, 3, 49-63.
Malhotra, A., Thomas, J. C., Carroll, J. M. & Miller, L. A. Cognitive processes in design. International Journal of Man-Machine Studies, 1980, 12, 119-140.
Mayer, R. E. Human nonadversary problem solving. In K. J. Gilhooly (Ed.), Human and machine problem solving. New York: Plenum, 1989.
Nwana, H. Domain-driven knowledge modelling: mediating & intermediate representations for knowledge acquisition (Technical report TR92-06). Keele: Keele University, Department of Computer Science, 1992.
Reitman, W. Heuristic decision procedures, open constraints, and the structure of ill-defined problems. In M. W. Shelley & G. L. Bryan (Eds.), Human judgments and optimality. New York: Wiley, 1964.
Reitman, W. Cognition and thought. New York: Wiley, 1965.
Rist, R. Models of routine and non-routine design in the domain of programming. In J. S. Gero & F. Sudweeks (Eds.), Artificial Intelligence in Design (Preprints of the Workshop of the Twelfth International Joint Conference on Artificial Intelligence, Sydney, Australia, 25 August 1991). Sydney: University of Sydney, 1991.
Simon, H.A. The structure of ill-structured problems. In N. Cross (Ed.), Developments in design methodology. Chichester: Wiley, 1984. (Originally published in Artificial Intelligence, 1973, 4, 181-201.)
Simon, H. A. The sciences of the artificial. Second edition. Cambridge, MA: MIT Press, 1981. (Reedition, with important additions, of The sciences of the artificial. Cambridge, MA: MIT Press, 1969.)
Thomas, J. C., & Carroll, J. M. The psychological study of design. In N. Cross (Ed.), Developments in design methodology. Chichester: Wiley, 1984. (Originally published in Design Studies, 1979, 1, 5-11.)

Ce texte est un pre-print de
W. Visser (1993). Design & knowledge modeling - Knowledge modeling as design. *CCAI, Communication and Cognition - Artificial Intelligence* [today entitled "CC-AI, The journal for the integrated study of Artificial Intelligence, Cognitive Science and Applied Epistemology"], 10(3), 219-233.

Visser, W. Giving up a hierarchical plan in a design activity (Research Report N° 814). Rocquencourt: INRIA, 1988. Also accessible at http://hal.inria.fr/inria-00075737/en/

Visser, W. More or less following a plan during design: opportunistic deviations in specification. International Journal of Man-Machine Studies. Special issue: What programmers know, 1990, 33, 247-278. Also accessible at http://hal.inria.fr/inria-00633544/en/

Visser, W. Designers' activities examined at three levels: organization, strategies & problem-solving. Knowledge-Based Systems, 1992, 5 (1), 92-104. Also accessible at http://hal.inria.fr/hal-00653324/en/

Visser, W. Planning and organization in expert design activities. In D. Gilmore, R. Winder & F. Détienne (Eds.), User-centred requirements for software engineering environments (pp. 25-40). Berlin, Germany: Springer, 1994.

Visser, W., & Hoc, J.M. Expert software design strategies. In J.M. Hoc, T. Green, R. Samurçay & D. Gilmore (Eds.), Psychology of programming. London: Academic Press, 1990.
This chapter is also accessible in Dr Alan Blackwell's Course material 2010–11 for "Usability of Programming Languages" http://www.cl.cam.ac.uk/teaching/1011/R201/ with the following "Note on copyright: This book is currently not available in print. A selection of chapters has been provided, with the permission of the editors, for non-commercial educational and research use only. Members of the Psychology of Programming Interest Group (PPIG) continue to conduct research as outlined here, and it is intended that specific chapters will be replaced with updated versions when individual authors are able to provide them. Volunteer assistance with OCR of these chapters, to assist those authors who no longer have their original manuscripts, would be appreciated."

Voss, J. F., Greene, T. R., Post, T. A., & Penner, B. C. Problem-solving skill in the social sciences. In G. Bower (Ed.), The psychology of learning and motivation (Vol. 17). New York: Academic Press, 1983.

Voss, J. F., & Post, T. A. On the solving of ill-structured problems. In M. T. H. Chi, R. Glaser & M. J. Farr (Eds.), The nature of expertise. Hillsdale, N.J.: Laurence Erlbaum Associates, 1988.

Wilson, M. Task models for knowledge elicitation. In D. Diaper (Ed.), Knowledge elicitation. Principles, techniques and applications. New York: Horwood, 1989.