

NOTICE WARNING CONCERNING COPYRIGHT RESTRICTIONS:
The copyright law of the United States (title 17, U.S. Code) governs the making of photocopies or other reproductions of copyrighted material. Any copying of this document without permission of its author may be prohibited by law.

Architects' Reasoning with Structures and Functions

O. Akin

EDRC 48-24-91

Architects' Reasoning with Structures and Functions¹

Ömer Akin, Ph.D.
Professor of Architecture
Carnegie Mellon
Pittsburgh, Pennsylvania 15213
USA

September 26, 1991

Abstract

While there is a visible increase of publications in the area of design thinking and signs that it is becoming a serious area of research, there is a lack of theoretical approaches that directly address its domain specific characteristics. This study attempts to develop such a theory directed towards *design reasoning* based on a protocol study and formal notation, among other things. Three domains of reasoning are described: *construction*, *object*, and *representation*. Inference making within and between these domains is described in terms of operations called *functions* and states called *structures* in a state-space representation of design. Five types of *mapping* which are illustrated by the protocol study are described using structures and functions. Both shortcomings and strengths of the proposed theoretical formalisms are discussed. Future work is indicated.

Studies of Design Thinking

Descriptive studies of architectural design are relatively recent. Inspired by studies in management science, cognitive psychology and computer science,² Eastman conducted the first known protocol study of architectural design (1969, 1970). This was followed by a number of studies, both at Carnegie Mellon and elsewhere which resembled the initial work by Eastman in some sense: methodologically, substantively or both.³

These studies of the design process were concerned with characterizing the process in its most general form, identifying the operations and representations which were responsible for the development of designs, calibrating the operational aspects of the human cognitive system, describing design tasks in the context of a general taxonomy of tasks, and doing all of this within the paradigms of information processing theory originally developed for the study of human problem solving (Newell *et al.*, 1972). The results of these works indicate that, 1) the design process exhibits characteristics that are shared by other information processing phenomenon, 2) certain behaviors of designers can be basically described using various cognitive and problem solving models, and 3) some aspects of design behavior go beyond those that can be demonstrated by simple, algorithmic procedures.

Subsequently, researchers seasoned by this initial encounter with the often overwhelming scope of issues subsumed under the general heading of design as well as others entering the field from related areas of design, especially from engineering design, built upon this early

¹ This work was partially funded by the Engineering Design Research Center, Carnegie Mellon University, Pittsburgh, PA 15213 and presented at the Workshop on Design Thinking held at Faculty of Industrial Design Engineering, Delft University, May 29-31, 1991, under the title of "A Structure and Function Based Theory for Design Reasoning."

² This includes, in particular, work by some of the pioneers of these fields, such as Herbert Simon and Allen Newell. The several publications which are dated to coincide with this period are Simon's *The Sciences of the Artificial*, (1969) and "Style in design," (1970); and Newell's "Heuristic programming.." (1970). This is not to say that others did not study this subject in some connection, such as Reitman (1964), Miller, Galanter, and Pribram (1960), Simon and Feigenbaum (1964). Such an assumption would be in contradiction to all that we know and believe about the continuum of scientific discovery and development which involve large number of studies that both succeed and precede any given work.

³ Krauss and Myer, 1970; Foz, 1972; Goumain, 1973.

foundation. These studies represent the beginning of diversification in research agendas in the area of design thinking. Some of these studies deal with the internal and external representations of designed objects (Gobert, 1989; Akin, 1978), others with the issues of design generation (Cuomo, 1989; McDermott, 1982; Darke, 1979), others with the knowledge base of design thinking (Waldron, *et.al.*, 1989; Akin, 1986), others with the formulation of design problems (Akin, 1991; Carroll, *et.al.*, 1978), others with the thought processes that apply to learning (Schön, 1983; Goor, 1974) and yet others with refining the general descriptions of the design process offered by the initial group of studies (Chan, 1990; Eckersley, 1988).

In parallel and quite unrelated to these studies researchers and scholars have also dealt with *prescriptive* accounts of the design process. Perhaps the most influential work in this category, certainly the most frequently cited one, is Alexander's *Notes on the Synthesis of Form* (1964). This is a treatise for systematic design based on decomposition of complex design problems into quasi-independent sub-problems and their subsequent recomposition through synthesis. This approach bears a certain kinship to the previous areas of investigation, albeit circuitously. One of these areas is founded by the collection of works known as the "design methods" movement of 1960's.⁴ Another one is Freeman *et.al.*'s work (1972) on functional reasoning. This work proposes a formal reasoning model for assembling design components into complex designs on the basis of the logical "links" that are inherent in these parts. This approach to design thinking which is potentially fruitful both in its own right and as the corollary to early descriptive models of design, has not been pursued sufficiently.⁵

Another group of works in the category of prescriptive models deals with reasoning about objects and representations as the rational basis of design thinking. The roots of this work can be found in the works of Earl (1980), Mitchell *et.al.*, (1976), Steadman (1970) and Stiny (1976) on rectangular dissections and space planning methods based on geometric theories (Grattan-Guinness, 1970). In its more recent form, this work has evolved into a mathematical theory of shapes suitable for design generation, as a result of works by Stiny (1990), Flemming (1989), Krishnamurti (1980) and Mitchell (1990).

All of these studies dealing with descriptive and prescriptive accounts of design constitute the kernel of our knowledge in the area of design thinking. One inescapable conclusion that can be drawn from this brief review is that the area of design thinking⁶ is coming of age. It is even possible to consider it an "emerging" discipline, with its own body of independent works converging around a common domain of problems, methods and findings. The telltale signs of this emergence are, 1) the increased rate at which new research results are appearing in publication, 2) increased specialization in the subjects covered by these studies, 3) availability of specialized publication agents, such as *Design Studies*, 4) appearance of comparative and retrospective studies (Stauffer and Ullman, 1988), 5) maturation and refinement of methodology in the field (Ericsson and Simon, 1980; Eckersley, 1988), and 6) increased interest of existing institutions to underwrite conferences, symposia and workshops in the area, such as the present workshop.

In contrast to this rather optimistic point of view, there also are some disheartening signs. One of these is a lack of clarity in the subjects of these works. It is difficult to say whether the findings of these studies about building, designers, design, construction, or all of the above. Another one is the absence of shared tools, methods and theories. Each study seems to follow the methodological practices of the discipline or field it is most closely affiliated with. A third one is the lack of common purpose between these studies. It is not clear whether the

⁴There is a great deal of work that has been and continues to be done in this category. However, there isn't enough space in this study to do an adequate job of reviewing them.

⁵The exception to this can be found in the field of artificial intelligence, which in a very significant way generates new search and problem solving techniques that can be applied to design problems. While these studies can be considered the rightful heirs to the legacy of Freeman and Newell's work, they do not address the question of design reasoning as much as dealing with specific problem solving cases and finding solutions for them, albeit staying within the prescriptive mode, characteristic of by Freeman and Newell's contribution.

⁶This term implies a sufficiently broad concept so that it is representative of a number of important views. The issue of more specialized interpretations requires the invention of new terms which will be addressed in the following section.

investigation of the principles underlying design is done to improve design practice, to abandon it, to create alternatives to it, to teach it or for some other reason unspecified here.

It is a combination of these shortcomings and the opportunities listed that has motivated the present work.

Theories of Design Thinking

Currently, studies in this area which can claim to have any theoretical foundation either base this on theories from outside of the area of design thinking (i.e., information processing, management science) or on theories that can only be applied to limited aspects of the subject of study (i.e., geometry, operations research). The basic problems arising from this are that the premises and results produced contribute principally to the respective areas from which these theories have been gathered. Furthermore, their coverage of design issues is not sufficiently comprehensive.

Theories dealing with information processing for example can be used to model the designer as a processor of data, show that the process of design is influenced by the cognitive limitations and capabilities of the designer, and cast the process itself as a problem of search. But they do not necessarily tell us the resources and capabilities needed to solve these problems, indicate the alternative models of data processing that might be considered or if the formulation of the design problem in these terms is the best that can be done, in the first place. Similarly mathematical models that describe some aspect of design thinking, such as geometric and topological formulations of shapes, do not cover other aspects of design thinking, such as constructability of shapes into objects and the performance of objects in the real world.

The principal purpose of this paper is to develop theoretical ideas about design thinking that bridge some of the gaps cited above. There are two basic reasons for attempting this at this time: one is the general need for such theories, and the other is the new heights reached by our knowledge in this area, making the contemplation of a greater degree of inclusiveness plausible.

The remainder of this paper is devoted to developing ideas about a theoretical framework which:

- 1) is explicitly based on design phenomenon and involves the manipulation of design information by a designer during a design related task including some of the basic sub-tasks of design,
- 2) is operational, representing all aspects of information manipulation found under design thinking and allowing for formal descriptions of the phenomenon including computation, algorithms or formal proofs, and
- 3) extendible to all important aspects of design including generation, evaluation, and selection of designs, formulation of design problems, and so on.

It is important to note that this is not meant to be a treatise on design theory or the evolution of design knowledge, in the manner of Thomas Kuhn's work (1970). Rather, this is merely an effort to find some common ground which can be applied towards generalizable underpinnings of the area of design thinking.

Before going further, let us briefly attend to the business of definitions.⁷ *Design thinking* connotes a comprehensive concept: the totality of the cognitive activities that occur during design.⁸ *Design reasoning*, as opposed to thinking, distinguishes the conscious, predictable use of *rules of inference* for the purposes of manipulating design information, from intuition.

⁷ There is a very extensive literature on the subject of the psychology of thought and reasoning. The subjects covered include both normal (Piaget, 1947; Wason, *et.al.*, 1972), developmental (Piaget, 1947) and pathological cases (Wason, *et.al.*, 1972). Although, our subject is categorically related to all of these, here we have virtually no room to consider any of this literature.

⁸ As comprehensive as this sounds let us note that this meaning still excludes other meanings: all thinking that occurs while not designing, as we are doing right now; thinking about the design activity itself or a design not for the express purpose of using it in a design; or extra-cognitive activities such as sensory activities unrelated to the design being produced.

Design intuition, another facet of design thinking, implies manipulations of a sub-conscious kind, where the rules of manipulation are not explicable. Let us assume, for the purposes of this paper that design involves two kinds of thinking, reasoning and intuition, and the conscious manipulation of design information under reasoning is conducted through the application of known rules of inference.

In the following sections, ways of making these rules explicit and the theoretical constructs that underlie them obvious, are explored. In other words attention is focused on design reasoning.

An Episode of Design

In order to examine design reasoning empirically and illustrate some of the aspects of the theory to be proposed in this paper, a protocol experiment has been conducted. Four subjects were asked to perform a "What's wrong with the picture?"-type task with architectural drawings. This task, which will be called the *reasoning task* in this document, provided a suitable context for subjects to explain the reasoning behind architectural details.

First the subjects were given a set of working drawings including a sheet of details containing nine independent errors of architectural detailing. These errors were planted in the drawings by the experimenters. Three of these were commission errors; three were omission errors; and the remaining three were errors of "operation." The first category consists of redundant elements, such as door stops, structural braces, and insulation layers. The second category consists of elements of a similar kind, missing. The third category, having to do with elements impossible to install during construction or to operate during normal use of the building, included things such as clearances of window swings and trim work.

Four sets of construction drawings were used in the experiment. The sets were selected from the architectural archives of the Carnegie Mellon Library. They were of buildings designed and completed by noted local architects. All buildings were of a modest size, 3,000 - 5,000 sq.ft., residential, and of conventional construction. Two were of wood frame construction, and the other two were composite, i.e., masonry and timber.

Four subjects who participated in the experiment were senior designers from four medium to large firms, in the metropolitan Pittsburgh area. At the time of the experiment, all subjects were carrying the principal responsibility for design of building details and quality control in their respective offices. Each subject took between one to two hours to complete the reasoning task. This included the time of set up and removal of the video equipment used to record the protocols. All subjects volunteered their time.

All errors "planted" in the stimuli were identified by all four subjects. In addition, subjects found, on the average three other errors in each set. These included errors of "style," building maintenance, and exterior skin related errors, all of which must have been committed by the original designers. In addition, after finding these errors and discussing why they should be considered errors, the subjects proceeded to make suggestions for fixing them.⁹

Appendix 1 shows the first 16:06 minutes of one of the four protocols collected. Each statement of this protocol that constitutes a "complete" idea, not necessarily a sentence, is included as a separate line, numbered consecutively from 1 to 269, in the transcription. The statements that abbreviate or paraphrase the transcription are shown in parentheses. Statements by the experimenter and descriptions of motor behaviors of the subjects are shown in square brackets. Drawings done by the subjects and parts of the original drawings given to the subjects are shown on the right column of Appendix 1. Line drawings, consecutively numbered between 1.1 and 1.7, are drawings made by the subject. Facsimiles of blueprints shown are parts of the original drawings given to the Subject at the onset of the experiment.

⁹ While the error detection and recovery aspects of this experiment are of general interest, they are the subject of another investigation being conducted jointly with Professor Elizabeth Adelson, Computer Science, Tufts University, Boston, MA and will not be discussed in detail here. Other aspects of reasoning in design will, however, be the focus of our attention.

Domains of Design Reasoning

In the most global sense, the protocol can be divided into large segments characterized by gross attributes of the subjects' reasoning: that used in *evaluating* old designs (for example, lines 1-64), and in *generating* new ones (for example, lines 65-103). This distinction becomes clear in the protocol through the analysis of statements that explicate subject's intentions in either category. For example, statement "65. What I would do is..." clearly indicates that the subject is starting a generation segment, which is born out by the statements that follow, "66. You would have masonry.." as well as the drawing activity which accompanies them.

Whether in a generative or evaluative segment of the protocol, each statement manifests the generation of new information based on previous ones. For example, the presence of a door on the north wall (Appendix 1, line 21), coupled with the fact that the prevailing winter winds are due from Northwest (line 22), leads to the statement that there is a weather control problem on the north side of the house (lines 23-30). The Subject continues to build sequences of inferences of this kind, constructing ultimately a "reasoned" explanation for the faulty flashing/counter-flashing detail and how it should be fixed (Appendix 1, lines 66-103). While this process seems continuous and homogeneous in syntactic terms, semantically, there are important distinctions to be drawn, particularly, in terms of the domain of knowledge applied to the reasoning task.

Consider the segment of the protocol (lines 50-64) following the initial familiarization with the drawings and the task. This segment is dominated by descriptive statements and acts of observation: "see," "look," "not see," etc. These statements serve to describe and illustrate the expected performance of designs found in the stimulus: "58. What that's going to allow is.. the water coming down that wall.. (will) hit that... and penetrate back into the wall." The Subject is directly concerned with the behavior of physical objects constituting a design as well as attributes of these objects.

On the one hand, he is relying on knowledge about the objective world to study the behavior of water around the construction joint. This will be called the Subject's *objective domain* of knowledge. On the other hand he is using representations, such as drawings, gestures and words, to describe these to himself and to others. This will be called the *representation domain* of the Subject's knowledge. Both domains seem to be necessary for the task being carried out. Representations make the relevant properties of the objective world become apparent. The running of the water down the surface of the wall is simulated by lines on a drawing. Similarly, special knowledge from the physics of objects is brought to bear on these representations to explain things like capillary action, accumulation and melting of snow and their consequences (lines 140-144).

Once a problem is diagnosed, the Subject's domain of reference shifts completely to other things. Consider the segment of the protocol, immediately following the above segment (lines 66-103). Armed with an understanding of the problem from the previous segment, this segment begins with inferences on the construction process and its requirements. All of the operations described in the subsequent lines (66-75), "bring," "put," "take over," "taper," "bring over," "bend over," are acts of construction, or of assembly of materials. While it is obvious that these actions are being carried out only hypothetically and not in reality, they are very much in the center of the reasoning that is taking place. The designer is manipulating surrogates, words and lines on paper, in order to grasp the manipulation of real objects at the construction site. In this segment, there seem to be at least two domains of reasoning intimately linked that constitute the substance (semantics if you like) of design: the *construction domain* and the *representation domain*, this time of the construction process, which again serves as a substitute for the real thing.

Once a building design is "constructed," in the mind and on paper, the domain of reasoning used in making inferences shifts away from the construction once again to the object domain, testing the performance of this design against the forces of the real world and using the representation domain as a surrogate.

In this fashion, the Subjects goes back and forth between three domains, those of construction (C-domain), object (O-domain) and representation (R-domain). More often than not, the R-domain serves as a medium to express the former two. Furthermore, by enabling the

mapping of information from either domain onto itself, it enables the mapping of information between the C- and the O-domains. It is not entirely clear from the protocols if this is the only way of making this mapping. It is quite conceivable that, since the C-domain and the O-domain are governed by the same physical laws, they are also directly linked.

The protocol offers relatively little information in terms of the sources of the knowledge in each domain. Each statement appears to reflect, implicitly if not explicitly, specific knowledge brought to bear on the problem at hand by a subject from external memories; for example, Table 1, line 6: "In a residential project you tend to have less experiences less knowledgeable craftsmen." Alternatively, inferences can be based on knowledge provided by previous statements; for example, Table 1, line 7: "So you have to take that into account." In both cases, the generic act of reasoning is to infer new information based on current information, whether it comes from the Subject or previously generated statements.

Summarizing, several general observations can be made:

1. three domains of reasoning, construction, object, and representation, exist,
2. these domains correspond to the intuitively known aspects of the design delivery process: construction, occupancy and documentation,
3. reasoning can occur both within domains as well as between domains,
4. within-domain reasoning, by definition, infers information in one domain from information belonging to the same domain, and
5. between-domain reasoning, by definition, infers information in one domain from information belonging to a different domain.

Functions and Structures: Ingredients of a Formalism

So far, observing the data in an aggregate manner, important distinctions about the use of human knowledge during design reasoning have been made. Such a distant view of our target, while is instrumental in making our considerations more comprehensive, cannot provide for us an operational understanding of the reasoning activity and its internal mechanisms. This goal requires a closer view of the reasoning task.

Analysis of the protocol at a greater level of detail requires methodical and previously tested approaches. Eckersley in his recent study (1988) developed a method for reliably encoding designers' protocols. He defined nine *a priori* categories of cognitive activity relevant to his protocols.¹⁰ The validity of these categories were tested through independent codifications of each protocol by multiple encoders and through the comparison of these results for consistency, afterwards.¹¹

Eckersley's study establishes a standard of empirical veridicality which does not exist in previous protocol studies of design. The present study is motivated by this approach and the desire to introduce rigor and consistency into the codification of subjects' responses. Since our purpose here is to develop an operational model of design reasoning, Eckersley's method does not suffice. Instead, it is necessary to develop another way of encoding the protocols. Here, towards this end, direct use of the linguistic categories represented by the individual words contained in the protocols will be made.

There are two general categories of linguistic elements in our protocols: descriptions and actions. Descriptions generally specify or assign value to some entity: "26. where you have windows, entrance." Actions speak of either intended or hypothetical activities of various sorts: "30. I would look for a heavy overhang here;" or "81. (The cleat) will receive the end of the flashing." Each category of statements requires a different notation. In the case of the descriptive statements, objects and relationships of objects are indicated: for example, a window, a window near the door, a window to the right of the door, a window across from the door, and so on. Thus, a flexible form of chaining an unspecified number of nouns, adjectives and other descriptive phrases normally found in natural language is needed. Without trying to

¹⁰ These are: 1) literal copy, 2) paraphrased copy, 3) inference, 4) intention/plan, 5) move, 6) search, 7) specific assessment, 8) general assessment, 9) none of the above.

¹¹ All but one subjects' protocols were codified into the nine categories with 95% or better agreement between independent coders.

Table 1: Transcription of lines 1-105 of the Protocol of Subject 1

| | | | |
|---|--|--|--|
| <p>1. I get the idea. 2. This is like the "what's wrong with this picture" game. 3. [E: Have you seen the building before?] 4. [The first question is: "what is it?"] 5. [A residential project.] 6. In a residential project you tend to have less experienced less knowledgeable craftsmen. 7. So you have to take that into account. 8. You can be as technical and complicated as you want in a building where you have higher skilled workmen. 9. I would also like to know where it is. 10. What the environment is. 11. North, South? 12. Is it on a hill? 13. Concern about water table.. 14. Slope? 15. Soil problem? 16. Underground foundation, horizontally.. orientation, sun, strong winds. 17. [E: makes assumptions, North arrow.] 18. I recognise that heavier winds, rain are coming from the back of the house (W, NW). 19. I see problems there [sheet #1] 20. That's an entrance. 21. Weather coming in from W, NW 22. So we're looking at a concern over water penetration.. 23. especially on grade areas 24. I would look for water penetration in these areas.. 25. where you have windows, entrance.. 26. I would hope there is an overhang to protect that 27. I'd be concerned about wind blowing here.. 28. at the door [sheet #1, back door area] 29. I would look for a heavy overhang here [points to wall along the back of kitchen] 30. It follows the contours. 31. It does not alter the elevation drastically. 32. It sits very well on the contours 33. So it is all on one level 34. The other thing I'd be concerned about.. the materials 35. Wood, plaster siding work.. 36. some concrete foundations [drwg #2, points to unexcavated area] 37. Concrete here 38. Brick cavity wall [points to front of bldg.] 39. You're looking at details to be generated for the contractor to build.. 40. but allow him enough flexibility to adapt to conditions at the site. 41. Accessibility to tools 42. [Room for hammer and screw driver] 43. There's a question about surface detail 44. What would he need to allow that [flashing and counter-flashing] to be fastened to that brick? looking at flashing and counter flashing detail over the roof 45. [E: what are you looking at?]</p> | <p>type (building) = ? type (building) = residential craftsmanship (residential (building)) = little consider (you) [crafts. (resid. (bldg.)) = little] can-be (you) [technical] IFF [craftsmanship (not (residential (building))) = lot] like-to-know (I) [location (building) = ?] environment (building) = ? orientation (building) = ? topography (site (building)) = ? watertable (site (building)) = ? slope (site (building)) = ? problem (soil (site (building))) = ? undergrd. (foundation (building)) = horizontal sun, wind (orientation (building)) = ? [E] recognize (I) [coming (heavy(wind)) [back (house)]] see (I) [problems (sheet #1)] entrance coming (weather) [direction = NW, W] look (we) [concern = (- (penetration (water))] -- = on (grade-area) look (I) [penetration (water) [areas = -]] -- = (windows, entrance) hope (I) [protect (overhang) (that)] concern (I) [blowing (wind) (-)] -- = back (door) look (I) [heavy (overhang) [areas = -]] -- = back (wall (kitchen)) follows (it) [contours] not (drastically (alter)) (it) [elevations] very-well (sitt) (it) [on (contours)] exists (it) [on (level) = one] concern (I) [other (things = -)] -- = (materials = -) -- = wood , plaster, siding (work) -- = concrete (foundations) exists (concrete) = here exists (brick (cavity (wall))) = front (building) look (you) [generate (build (contractor) [details]]] allow(flexibility) [adapt (contractor) (building, conditions (site))] access (tools) room (hammer, screwdriver) exists (question (surface (detail))] [need (contractor) [allow (fasten (flashing-counterflashing, brick))] = ? [E]</p> | <p>48. [E: What do you see?] 49. I'm looking at flashing-counter-flashing details over the roof above the entry . 50. I see.. 51. I don't see any flashing in detail 52. The counter-flashing apparently being recessed into the joint 53. That's typical 54. However, what they've done is.. 55. they've taken the top of that flashing.. 56. and brought it back out 57. What that's going to allow is.. 58. the water coming down that wall.. 59. hit that..[shows flashing] 60. and penetrate back into the wall. 61. I'm concerned about that 62. I would change that flashing.. 63. so that you would have.. 64. What I would do is.. 65. You would have masonry [drwg #1.1] 66. Then you can bring flashing in like this. 67. A lot of times what they do is 68. They'll put a piece of wood on here 69. And will take this over 70. Sometimes they will taper the top of the wood 71. So that you can bring this over 72. .. and bend this over 73. (no sharp edges 74. And then bring this down 75. with a continuous cleat inside 76. so that the flashing comes up to this point, whatever.. 77. and on down the roof. 78. And then this continuous cleat.. 79. which is a piece fastened to this wood.. 80. will receive the end of the flashing. 81. This then goes back ... 82. and it keeps from blowing up in the wind.. 83. the rain.. 84. as these things get older they will flop out.. 85. If they are not attached. 86. If you nail through.. 87. you automatically put a hole through .. 88. which water (whatever) will go through. 89. So you try to keep that thing as watertight as possible.. 90. You hide your nail behind. 91. You hide your cleat under it. 92. You bring this back.. 93. and you take this back into your joint 94. A lot of time.. 95. they use what they call lead wedge.. 96. which will wedge it back in there. 97. You try to take it back far enough.. 98. so that you have a raked joint.. 99. and they will put caulking in there. 100. When the water comes down 101. it will hit that and run back down. 102. It won't get back in there. 103. [E: what are you looking at?] 104. [E: What do you see?]</p> | <p>[E] look (I) [over (roof, above (entry) (details) (flashing, counter-flashing))] see (I) not (see) (I) [detail (flashing)] recessed [counterflashing, joint] typical (that) do (they) = ? taken (they) [top (flashing)] and back(bring)(they) (it) allow (that) = ? down(comes) (water) (that (wall)) hit (water) (that = flashing)] and penetrate (water) (into (wall)) concerned (I) (that) change (I) (flashing) have (you) do (I) = ? have (you) [masonry] draw (masonry) in(bring) (you) [flashing, masonry] draw (flashing, masonry) do (they) = ? draw put (they) [here (wood)] draw (wood) and take (over (this)) draw (flashing) taper (they) [top (wood)] draw (wood) over (bring) (you) (this=flashing) draw (flashing) over (bend) [this=flashing; wood] draw (flashing) sharp (edges) = none and bring-down [this=flashing] draw (flashing) inside (continuous (cleat, masonry,)) up (comes) (flashing) [here, -] draw (flashing) -- = down (roof) draw (flashing) and continuous (cleat) = - draw (cleat) fastened [- = piece, wood] draw (cleat) receive (cleat, end (flashing,)) draw (cleat) back (go) (this) draw (cleat) and keep-from (blow (wind) (it)) draw (flashing) rain out (flop) (things) IFF get-older (things) - -- = IFF not (attach) [they = masonry, wood] - -- = in (put) (you) [hole, flashing] draw (hole) through (go) (water) [hole] draw (water) try (you) [keep-watertight (things)] hide (you) [behind (nail, flashing)] draw (nail) hide (you) [behind (cleat, flashing)] draw (cleat) back(bring) (you) (this) and back(take) (you) [this, joint] draw (joint) lot (time) use (they) [lead (wedge)] back (wedge) (it, in (there)) try (you) [take (it, enough (far (back))] have (you) [raked (joint)] put (they) [caulking, in (there)] draw (caulking) down (come) (water) hit (water) (that) and back, down (run) (water) not (back (in (get))) (it, there) [E] [E]</p> |
|---|--|--|--|

solve the larger problems of structural linguistics, a simple notation is assumed:

(modifier₁ (modifier₂ (... (modifier_n (object)...))), [1]

where the symbol "modifier" stands for any combination of descriptors that specify something about one of the modifiers of the "object" or the "object" itself. Of course to show parallel modify relationships rather than nested ones one can simply use the form:

(modifier₁, modifier₂ ... modifier_m (... (modifier_n (object)...))), [2]

In the case of actions, there are three ingredients: the act, the agent and the subject. The act is a verb. The agent is the thing or person performing the act, either figuratively or actually. The subject, is a description of the kind shown above indicating the entity acted upon. The overall syntax for this is:

act (agent) [description] [3]

The syntax for the act itself, in some instances, can get as complex as a description:

(action-modifier₁ ... (... (action-modifier_n (act)...))), [4]

where the action-modifiers are adverbs and other compound clauses that further describe the act. The real actions of the Subject, such as drawing lines and shapes on paper, are also included in this category.

Table 2

Functional and Structural Constructs Used for
Transcribing Protocol Statements
in BNF Notation

| | |
|-----------------|--|
| statement | ::= abstraction application conditional valuation question |
| application | ::= abstraction [application] |
| conditional | ::= abstraction IFF conditional |
| valuation | ::= application = abstraction |
| question | ::= application = ? |
| abstraction | ::= structure function |
| structure | ::= existential-atr unary-str binary-str multiple-str |
| unary-str | ::= existential-atr existential-atr (unary-str) |
| binary-str | ::= existential-atr existential-atr (binary-str, binary-str) |
| multiple-str | ::= existential-atr existential-atr (multiple-str) multiple-str, multiple-str |
| existential-atr | ::= <attribute> <object> existential-atr (existential-atr) |
| function | ::= existential-act unary-func binary-func multiple-func |
| unary-func | ::= existential-act existential-act (unary-func) |
| binary-func | ::= existential-act existential-act (binary-func, binary-func) |
| multiple-func | ::= existential-act existential-act (multiple-func) multiple-func, multiple-func |
| existential-act | ::= <action> <action> (existential-act) <action> (existential-act) (<agent>) |

Table 2 shows, in BNF notation,¹² the entire syntax for, 1) actions, which will be called *functions*, 2) descriptions, which will be called *structures*, and 3) their compound statement forms. Since the statements in the protocol come in a variety of forms, including questions (line 10), conditional statements (line 85), value assigning statements (line 6), and compound statements (lines 23, 24), a variety of statement forms are included. Table 2 shows the first 105 lines of the protocol transcribed in this notation. On the left hand column, the actual statements of the protocol and, on the right hand column, the transcribed form of each statement are shown.

In terms of reasoning, these two forms, structures [1][2] and functions [3][4], have a critical relationship. Structures define typical design situations and the functions show how they can be modified which gives rise to new structures and the evolution of a design, during the course of the overall design process. Thus, they constitute a set of necessary ingredients for design reasoning.

Furthermore, the formal representation of these two entities, provides for us the primitive elements of an operational theory of design reasoning. The critical aspects of this formalism, so far, are, 1) comprehensiveness, i.e., accounting for the entire protocol including both its generative and evaluative segments, and 2) operability, i.e., enabling the representation of the protocol in algorithmic terms.¹³

Functional and Structural Mapping of Design Information

So far, three basic domains of knowledge that in one sense constitute the semantics of the functional reasoning task studied here have been identified; and a syntax for codifying the statements included in one of the protocols has been proposed. In this section, some of the logical and operational consequences of these proposals will be shown. Hopefully, from all of this emerges the beginnings of a theoretical framework equally applicable to other instances of the same phenomenon.

What precisely is meant when one says that a subject reasons within a certain domain, say, the construction domain? What precisely is manipulated when an inference occurs, in that domain? Does this impact the knowledge present in the other two domains? In order to answer these questions, and others like it, the basic process of reasoning in design has to be further penetrated.

Consider the first 17 lines in Table 1 (lines 66- 82). Statement 66 indicates, from the inspection of the drawings of the house in review, that the detail being designed is attached to a masonry wall. The first drawing in Figure 1 illustrates the sketch generated by the Subject at this time. This inference is made on the basis of the examination of the drawings, thus is one that generates information for the C-domain based on information obtained in the R-domain. Statement 67 indicates, based on statement 66, that you can bring the flashing around the block in the manner shown in Figure 1.2. This inference while is carried out in the R-domain is paralleled with information being generated in the C-domain. This new information includes the flashing and its physical configuration, based on previous information which included only the masonry block. Both of these states (Figures 1.1 and 1.2), while represented on paper, that is, simultaneously reflected in the R-domain, are reasoned through in the C-domain at the moment of the inference. The reader should not have any difficulty extending this interpretation to the remaining sketches shown in Figure 1 on their own.

Even in the case of such a minor design move, a great deal of knowledge may be necessary to carry out the inference shown, such as the condition of the block wall, its relationship to the roof, the location and orientation of the building.¹⁴ Alternatively, the information may have

¹² A formal notation that is widely used in computer science and structural linguistics to define syntaxes with accuracy and rigor (Bachus, 1960).

¹³ Ideally, this would lead to representations in mathematical form, as well. There isn't enough space here to treat this aspect of the present theory. We intend to address this issue in the near future.

¹⁴ Incidentally, some of these consideration are illustrated in the first segment of the protocol.

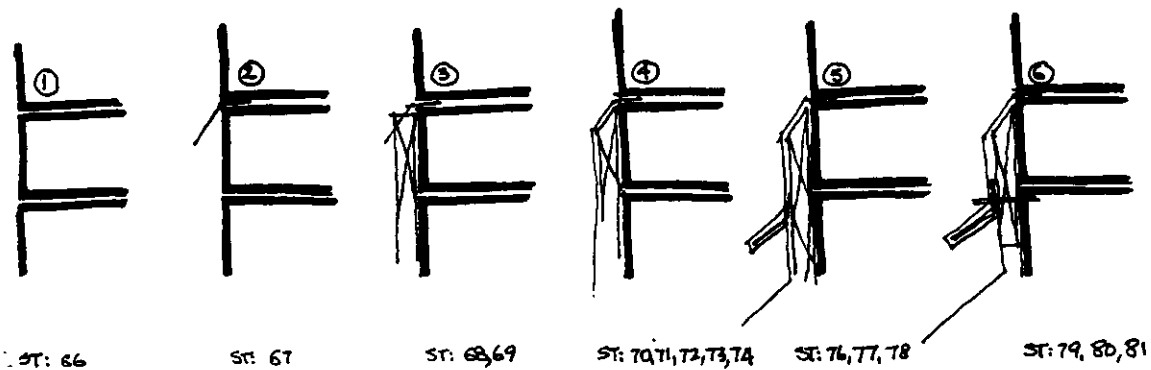


Figure 1. Six sketches generated by the Subject during the interaction on lines 66- 82 of Table 1

been pulled out of the subject's knowledge of "patterns" or prototype details, which requires relatively little reasoning.¹⁵ Whether from a prototype or from scratch, the basic reasoning operation that generates new information¹⁶ can also be seen as the transformation of a given state into the next one. This can be modeled using a *state-space* representation (Hunt, 1975).

In a state-space representation there are static states, akin to the *structures* defined in the previous section, and operations that manipulate these states in some way, similar to the *functions*, also previously defined. By formulating a set of functions and initial structures that are allowed to exist in a state-space, one can implicitly define the entire set of new structures that can be generated:

$f(s_1) \rightarrow s_2$, such that s_1 and s_2 are proper sub-sets of S , where S is the set of states defining a domain for all t for all f and s_i , f is a function, and s_i is any given state.

By defining sets of pairs of variables (Q), functions and initial structures (f, s_i); any sub-set of the reasoning operations necessary in the reasoning task can be formally defined. s_i and f consist of sets of values defined by [1][2] and [3][4], respectively.

Take the sequence shown in Figure 1. The initial structure in this case is the set of blocks shown in 1.1. The function applied to this structure is "in (bring) (you) [flashing, *masonry*]¹⁷". (Table 1, line 67, right column). In other words, the action contained in this *binary*-function is one of "bringing in" the "flashing" between the "masonry" elements. Subsequently, other actions are taken, in a particular sequence, to install the tapered block under the flashing, install the cleat and the couter-flashing and secure the assembly to the wall, as shown in Figure 1. The important issues to note in this sequence of events are:

- 1) the sequence of functions applied to existing structures help generate new structures by modifying them.
- 2) all new structures generated remain in the C-domain, with their representations remaining in the R-domain.
- 3) the final structure generated (Figure 1.6) is a potentially constructible object.

¹⁵ In addition, the distinction between these may be important for explaining expert and novice behavior and ultimately would be important in understanding different reasoning modes for education. While this is a possible extension of the theoretical work proposed here, it will not be expanded further in the present work.

¹⁶ That is, regardless of whether it is within or between domains, and regardless of whether the sources of the domain knowledge is found in prototypes or other didactic knowledge.

¹⁷ Italics indicate implicit reference, not direct, to the word italicized.

- 4) the sequence of steps that lead to this structure mimic in some form the construction process that might produce it in reality.
- 5) subsequent to generating a final structure the Subject carries out inferences that mimic the expected behavior of real world conditions surrounding the design (i.e., rain around this joint detail, as shown in the segment in Appendix 1, lines 129-131), which constitutes reasoning in the O-domain.
- 6) testing of the structures generated in the C-domain is accomplished by applying the functions (i.e., rain, wind) in the O-domain.
- 7) the resulting behavior found at the end of the inferences carried out in this manner help predict what might happen in reality when the construction is complete.
- 8) the steps that lead to the final inference in the O-domain also remain in that domain, and
- 9) the R-domain is used to simulate the O-domain inferences on paper.

Thus it can be asserted that within-domain reasoning can be represented as pairs of variables, f and s , which by definition, would guarantee that all new structures inferred would remain within a given domain. For example, in the construction domain, Q^C would consist of f^C, s_1^C , such that all s^C resulting from the application of f^C would be a proper subset of S^C , or all possible C-domain structures. In this context, the reasoning task of the Subject can be defined by determining Q^C , and carrying out the possible inferences within this set until a likely solution is generated. At that point, if not earlier, it is necessary to discover the consequences of this construction in the O-domain. That is between domain inferences are also necessary in order to complete the reasoning task.

Table 3

Mappings in the C-domain episode of the protocol in Appendix 1, lines 66-82.

s = structure; s_1 = initial structure; s_t = terminal structure; f = function; c = C-domain; o = O-domain; r = R-domain

| mapp'g | parameters | result | type | domain | range | explanation |
|----------------|------------------------|-------------|----------------|--------|-------|---|
| m ₁ | $s_1^{c,o}, f^{c,o}$ | $s_t^{c,o}$ | within-domain | C,O | C,O | design from scratch: mapping from any C- or O-structure to any other C- or O-structure within the same domain |
| m ₂ | s_1^r, f^r | s_t^r | within-domain | R | R | design from prototype: mapping from any R-structure to any other R-structure using prototypical designs patterns |
| m ₃ | $f_1^{c,o}, s_1^{c,o}$ | $f_2^{c,o}$ | within-domain | C, O | C, O | performance or constructability evaluation: mapping from C- or O-structure to C- or O- function to verify performance of structure |
| m ₄ | $s_1^{c,o}, f^{cr,or}$ | s_t^r | between-domain | C,O | R | representation: mapping from C- or O-structure to R-structure |
| m ₅ | $s_1^r, f^{rc,ro}$ | $s_t^{c,o}$ | between-domain | R | C,O | drawing reading: mapping from R-structure to C- or O-structure |

In the case of between-domain inferences, the resulting inferences would be guaranteed to fall in the category of structures included in a domain other than that of the initial structure, s_1 . Thus, given an initial structure in the R-domain and the pair Q^{CO} (where the double superscript indicates the direction of the mapping between domains) the initial structure, s_1^C , will be in the C-domain, and all s^O resulting from the application of the functions, f^{CO} , will be a proper subset of S^O , or in the O-domain. Take the example provided above. The function included on line 129, namely, "allow (you) [hit (water) [this]]," is applied to the initial structure shown in Figure 1.6. A few inferences later, lines 130, 131, the conclusion that water will go down and around the joint, is reached.

Viewed in these terms, design reasoning, becomes a series of mappings of information within and between knowledge domains. Table 3 shows five different forms of mapping that exist in the analyzed data.

The first mapping, m_1 , is the one that is illustrated in Figure 1. This represents the task of transforming a given structure, such as the masonry wall, into a desired one in the same domain, such as the masonry wall with a flashing detail. The operational difficulty here is to insure that all mappings remain in the same domain and the structure generated at the end is the one that is desired, in other words, controlling the range of the functions applied ($f^{C,O}$). It would be convenient to assume that it is entirely possible to accomplishing this through the careful definition of s_1 and $f^{C,O}$, that is the specific state-space of the reasoning task in question.¹⁸

The second one, m_2 , serves a similar purpose as m_1 . The difference is m_2 relies on prototype designs which are already mapped in their entirety into the R-domain. The Subject does not have to bother with the internal consistency of the reasoning which is so crucial in m_1 , so long as the reasoning within the R-domain is consistent. The problem of knowledge representation is reduced manifold to the problem of knowledge about representations, such as drawing parts or other views of given drawings. The entire reasoning operation is carried out in the R-domain.

The third kind of mapping, m_3 , which also remains within-domain (in the C- and O-domains), serves a different purpose than the first two. m_3 enables evaluation of designs that are already generated while the previous two mappings help in generating them. The evaluation is done to discover if the structures obtained during generation perform in the desired manner. This performance, water running down into the gutter, sun penetrating through the glass, and so on, is represented in functional terms. This mapping is from structures to functions, subject to other functions, such as falling of the rain, blowing of the wind, which belong to the C- or the O-domain.

The last two forms of mapping are between domains; typically, in the case of m_4 , from the C- or O-domain to the R-domain; and, in the case of m_5 , from the R-domain to the C- or O-domain. These are mappings that are governed by the rules and techniques of generating orthogonal drawings, for example, in the first case; and those of reading orthogonal drawings in the second case.

Theoretical Implications

In the previous sections, the domains of knowledge used in design reasoning, the mechanisms of inferences made, and the mapping of information within and between these domains using these mechanisms have been described. This is aimed at meeting all three goals

¹⁸ However, there is more to it than that. Issues of representing very large knowledge bases may be applicable here, since both f^{CO} and s depend on domain specific knowledge. In which case the problem of formally defining Q may go beyond the scope of this work.

of this work specified at the outset: 1) being founded in the design task, 2) being formulated in operational terms, and 3) being applicable to all major aspects of the design continuum. In this final section, the comprehensiveness of this proposal, its significance, and the issues that remain unexplored will be considered.

Presently, the ideas outlined in this work are far from providing a fully fledged theory of design reasoning. Many of the conclusions presented are tentative; the generalizations untested, and their implications speculative. The bulk of the formal work, definition of axioms and proofs, has not been started. A wider sample of applications has not been attempted. All of this remains to be done.

This work provides the foundation, however, upon which such advances can be constructed. In doing so, it accomplishes several other important things. The first one of these is providing a vocabulary for functional descriptions. Previously, models of reasoning in physical object domains have been proposed by others.¹⁸ These approaches, by virtue of their geometric basis enable the treatment of the subject rigorously and formally. However, functional considerations in design that deal with the behaviors resulting from construction or the principles of physics have not been directly included in these theoretical constructs. Functionally related objects often do not have contiguity relations or shape resemblances. The connections that exist between the steel rods and the weight of the concrete or that of the flashing detail and the skylight on the roof do not logically derive from a theory of shapes. To include functional constructs in design, it is mandatory to include the logic of functions in design reasoning formalisms.

Another aspect of the logic of functions as opposed to the logic of forms is the idea of performance. Performance, with respect to some criteria related to the behavior of rain drops, occupants or carpenters, not to mention carpenter ants, are all inherent in the definition of functions as proposed in this work. It is inconceivable to propose a reasoning formalism for design, which is comprehensive of both generation and evaluation of designs, without incorporating the representation and manipulation of variables that stand for performance values.

The second worthwhile aspect of the theoretical constructs proposed here is their ability to represent specific aspects of the design process. The various forms of mapping between domains, particularly, m_1 and m_2 and the distinctions between them, accommodate the distinctions that exist between design from scratch *versus* design from prototypes. Similarly, the distinctions between m_4 and m_5 , parallel the production and interpretation of architectural representations, respectively; while m_3 corresponds to performance evaluation in design. In short, the theoretical basis of this work is one which, at least at the level of aggregation suggested by these examples, lends itself to many aspects of the design process.

Another important dimension of this consideration is the ability to represent the structuring and re-structuring of design problems (Akin, 1991). By defining a sub-domain, that is Q^* in the present notation, it is possible to precisely define the domain of a set of design transactions, all possible structures that can be generated in that domain, and furthermore the precise form in which this domain can be redefined through the re-specification of Q^* .

The third useful aspect of the present formalism is its ability to show the special relationship that drawings, models, and so on, have to design reasoning. Representations are used to document a design for the use of a contractor who is responsible for building a building and for the use of the client to understand and manage the physical infrastructure provided by the building. These purposes usually a part and parcel of professional conventions are directly reflected by the formalism proposed here.

A fourth attribute, which has been a goal of this work from the beginning, is the operability of the proposed theory. What is meant by this is the possibility of translating the mappings of information described by the theory into algorithmic or mathematical forms. There are important questions that need to be investigated before a clear conclusion can be reached in this regard.

The branch of mathematics that deals with functions, combinatorics or lambda-calculus,

¹⁸ For example, Flemming, Krishnamurti, Mitchell and Stiny's works.

provides a framework for the questions that need to be asked. In order to have computable functions (f) the domains, |D|, of these functions have to have closure. This represents a difficult problem even if we consider small sub-domains of the over all design problem. For example, in the case of the function that bends flashing over wood block, there are many degrees of freedom to contend with: the material, its thickness, availability of tools, skill, etc.

Another aspect of closure is the range of $Q_1(f, s)$, which will be shown as $|R|_1$. This must match one of a large number of possible ranges, $|R|^*$, where the superscript indicates set of all ranges, as a function of whether the mapping is within- or between-domains. Many of the mappings that are defined here are likely to be polyadic, that is, contain an arbitrary number of functions, which further compounds the closure problem at higher levels provided by the theory.

One way of approaching this problem is to find sub-domains that for reasons that cannot yet be articulated, have properties that lend themselves to computability, such as Turing machines, Markov chains or recursive functions (Revesz, 1988). Ultimately, the challenge that underlies this issue is the ability to close the types and values of the function and structure variables we have defined. This may be possible through the limits which can be imposed on types of data admissible in the state-space or through the definition of new forms of inference making. A new logic for design may have to be invented before the process of reasoning in design will become a viable research area (Zeng, *et al.*, no date).

There are undoubtedly other issues that can be considered under this section. However, the issues already identified are both challenging and hopefully provocative enough so that this search for issues can be presently terminated.

Bibliography

- Akin, Ö. (1991) "Heuristic generation of layouts (HeGeL): based on a paradigm of problem structuring," *Environment and Planning B: Planning and Design*, (in print).
- Akin, Ö. (1986) *Psychology of Architectural Design*, Pion Limited, London.
- Akin, Ö. (1978) "How do architects design?" in *Artificial Intelligence and Pattern Recognition in Computer-Aided Design*, ed. by J-C Latombe, North Holland, New York.
- Alexander, C. (1964) *Notes on the Synthesis of Form*, Harvard University Press, Cambridge, MA.
- Bachus, J. W. (1960) "The syntax and semantics of the proposed international algebraic language for the Zurich ACM-GAMM Conference," in *Information Processing*, UNESCO, Paris.
- Caroll, J. M., J. C. Thomas and A. Malhotra (1978) "Presentation and representation in design problem solving," *British Journal of Psychology*, 71, 143-153.
- Chan, C. (1990) "Cognitive processes in architectural design problem solving," *Design Studies*, 11, 2, 60-80.
- Cuomo, D. L. (1988) "A study of human performance in computer-aided architectural design: methods and analysis," unpublished PhD thesis, State University of New York at Buffalo.
- Darke, J. (1979) "The primary generator and the design process," *Design Studies*, 1, 1, 36-44.
- Eckersley, M. (1988) "The form of design process: a protocol analysis study," *Design Studies*, 9, 2, 86-84.
- Earl, C. F. (1980) "Rectangular shapes," *Environment and Planning B: Planning and Design*, 7, 311-342.
- Eastman, C. M. (1970) "On the analysis of intuitive design processes," in *Emerging Methods in Environmental Design and Planning*, ed. by G. T. Moore, The MIT Press, Cambridge, MA., pp. 21-37.
- Eastman, C. M. (1969) "Cognitive processes and ill-defined problems: a case study from design," *Proceedings of the First Joint International Conference on Artificial Intelligence*, Washington, D. C., Joint International Conference on Artificial Intelligence.

- pp. 675-699.
- Ericsson, K. A. and H. A. Simon (1980) *Protocol Analysts: Verbal Reports as Data*, The MIT Press, Cambridge, MA.
- Flemming, U. (1989) "More on the representation and generation of loosely packed arrangements of rectangles," *Environment and Planning B: Planning and Design*, 16, 327-359.
- Foz, A. (1973) "Observation on the designer behavior in the parti," in *The Design Activity International Conference, Volume I*, Printing Unit, University of Strathclyde, Glasgow, pp. 19.1-19.4.
- Freeman, P. A. and Newell A. (1971) "A model for functional reasoning in design, in *Proceedings of the Second International Joint Computer Conference on Artificial Intelligence*, British Computer Society, 13 Mansfield street, London W1M 0BP, pp. 621-640.
- Gobert, J. D. (1989) "The interpretation of architectural plans by expert and sub-expert architects," unpublished Master of Arts thesis, Department of Educational Psychology and Counseling, McGill University, Montreal, Quebec.
- Goor, A. (1974) "Problem Solving Processes of Creative and Non-creative Students," *Dissertation Abstracts International, University of North Carolina*, 74, 3, 3517A.
- Goumain, P. G. (1973) "Design methods and designer's methods," in *The Design Activity International Conference, Volume I*, Printing Unit, University of Strathclyde, Glasgow, pp. 23.1-23.8.
- Grattan-Guinness, I (1970) *The Development of Foundations of Mathematical Analysis from Euler to Riemann*, The MIT Press, Cambridge, MA.
- Hunt, E. B. (1975) *Artificial Intelligence*, Academic Press, New York.
- Krauss R. I. and R. M. Myer (1970) "Design: a case history," in *Emerging Methods in Environmental Design and Planning*, ed. by G. T. Moore, The MIT Press, Cambridge, MA., pp. 11-20..
- Krishnamurti, R. (1980) "The arithmetic of shapes, *Environment and Planning B*, 7, 463-484.
- Kuhn, T. (1970) *The Structure of Scientific Revolutions*, second edition, International Encyclopedia of Unified Science, Volume 2, Number 2, University of Chicago Press, Chicago.
- McDermott, J. (1982) "Domain knowledge and the design process," *Design Studies*, 3, 1, 31-36.
- Miller, G. A. E., E. Galanter, and K. H. Pribram (1960) *Plans and the Structure of Behavior*, Henry Holt, New York.
- Mitchell, W. J. (1990) *The Logic of Architecture*, The MIT Press, Cambridge, MA.
- Mitchell, W. J., P. Steadman and R. S. Liggett (1976) "Synthesis and optimization of small rectangular floor plans," *Environment and Planning B: Planning and Design*, 3, 1, 37-70.
- Newell, A. and H. A. Simon, (1972) *Human Problem Solving*, Prentice-Hall, Englewood Cliffs, New Jersey.
- Newell, A. (1970) "Heuristic programming: ill-structured problems," in *Progress in Operations Research, Volume 3*, ed. by J. A. Arnofsky, John Wiley, New York, pp. 360-414.
- Piaget, J. (1947) *The Psychology of Intelligence*, translated from French by Malcolm Piercy, Routledge & Kegan Paul, London.
- Reitman, W. R. (1964) "Heuristic decision procedures, open constraints and structure of ill-defined problems," in *Human Judgments and Optimality*, eds. M. W. Shelly and G. L. Bryan, John Wiley, New York, pp. 282-315.
- Revesz, G. E. (1988) *Lambda-Calculus, Combinatorics and Functional Programming*, Cambridge University Press, Cambridge.
- Schön, D. A. (1983) *The Reflective Practitioner: How Professionals Think in Action*, New York, Basic Books.
- Simon, H. A. (1970) "Style in design," in *Proceedings of the Environmental Design Research Association Conference II*, Department of Architecture, Carnegie-Mellon University, Pittsburgh, PA, pp. 1-10.
- Simon, H. A. (1969) *The Sciences of the Artificial*, The MIT Press, Cambridge MA.
- Simon, H. A. and E. A. Feigenbaum (1964) "An information-processing theory of some effects of similarity, familiarization, and meaningfulness in verbal learning," *Journal of Verbal Learning and Verbal Behavior*, 3, 385-396.

- Steadman J. P. (1970) "The automatic generation of minimum standard house plans," in *Proceedings of the Environmental Design Research Association Conference II*, Department of Architecture, Carnegie-Mellon University, Pittsburgh, PA.
- Stauffer, L. A. and D. C. Ullman (1988) "A comparison of the results of empirical studies into the mechanical design process," *Design Studies*, 9, 2, 107-114.
- Stiny, G. (1990) "What is design?" in *Environment and Planning B: Planning and Design*, 17, 97-103.
- Stiny, G. (1976) "Two exercises in formal composition," *Environment and Planning B: Planning and Design*, 3, 187-210.
- Waldron, M. B. and K. J. Waldron (1989) "A time sequence study of a complex mechanical system design," *Design Studies*, 9, 2, 95-106.
- Wason, P.C. and P. N. Johnson-Laird (1972) *Psychology of Reasoning, Structure and Content*, Harvard University Press, Cambridge MA.
- Zeng, Y. and G. D. Cheng (no date) "On the logic of design," unpublished report, Research Institute of Engineering mechanics, Dalian University of Technology, Dalian 116023, China.

Appendix I: Protocol of Subject 1

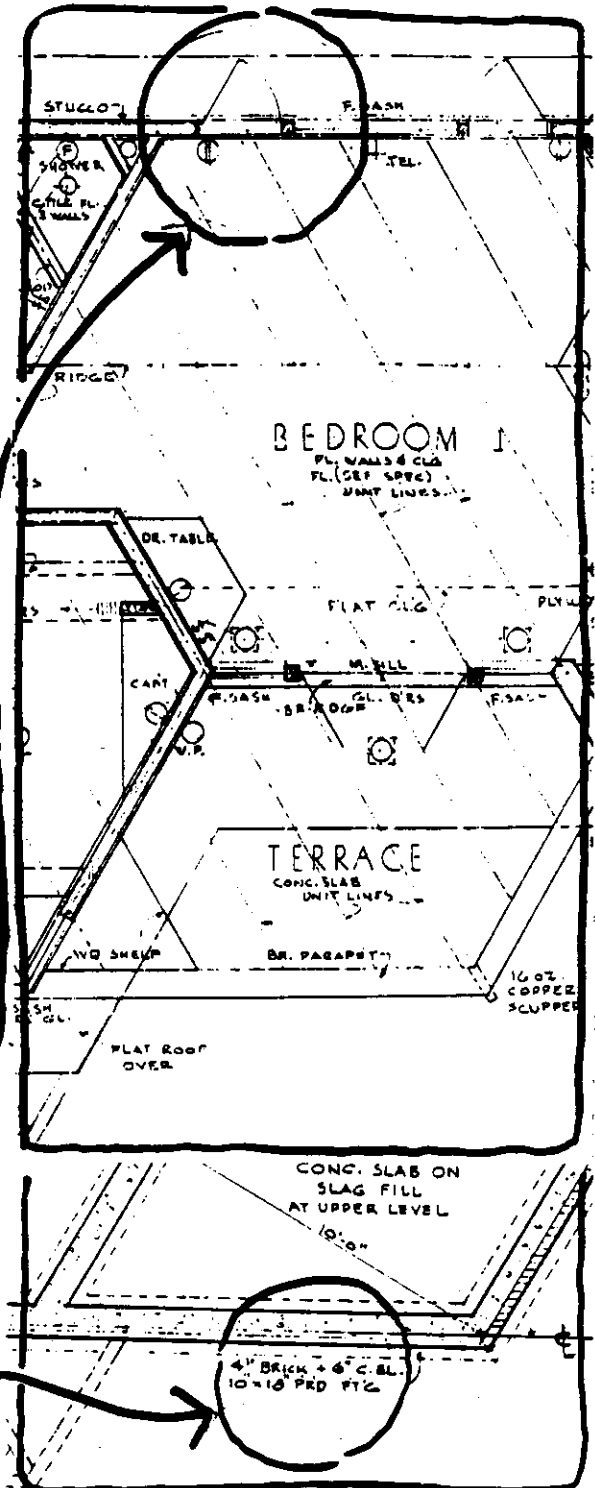
Subject 1.

00:00

...

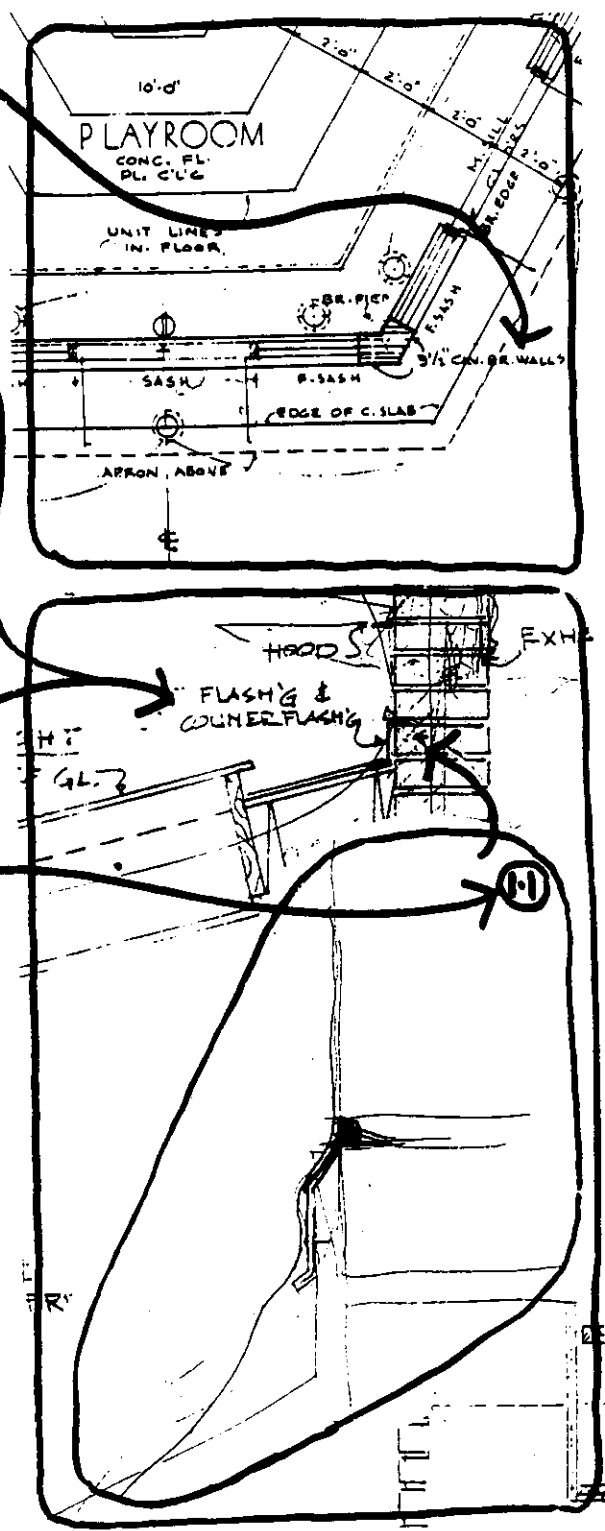
10:26

1. (I get the idea.
2. This is like the "what's wrong with this picture" game.)
3. [E: Have you seen the building before?]
4. (The first question is: "what is it?")
5. (A residential project.)
6. In a residential project you tend to have less experienced less knowledgeable craftsmen.
7. So you have to take that into account.
8. You can be as technical and complicated as you want in a building where you have higher skilled workmen.
9. I would also like to know where it is.
10. What the environment is.
11. North, South?
12. Is it on a hill?
13. Concern about water table..
14. Slope?
15. Soil problem?
16. Underground foundation, horizontally..
17. orientation, sun, strong winds.
18. [E: make assumptions. North arrow.]
19. I recognize that heavier winds, rain are coming from the back of the house (W, NW).
20. I see problems there [sheet #1]
21. That's an entrance.
22. Weather coming in from W, NW
23. So we're looking at a concern over water penetration..
24. especially on grade areas
25. I would look for water penetration in these areas..
26. where you have windows, entrance..
27. I would hope there is an overhang to protect that
28. I'd be concerned about wind blowing here..
29. at the door [sheet #1, back door area]
30. I would look for a heavy overhang here [points to wall along the back of kitchen]
31. It follows the contours.
32. It does not alter the elevation drastically.
33. It sits very well on the contours
34. So it is all on one level
35. The other thing I'd be concerned about..
36. the materials
37. Wood, plaster siding work..
38. some concrete foundations [drwg #2, points to unexcavated area]



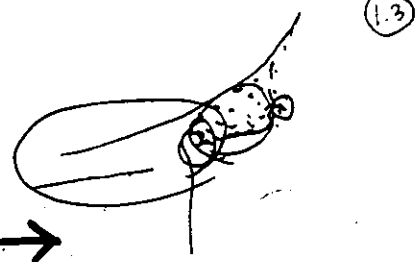
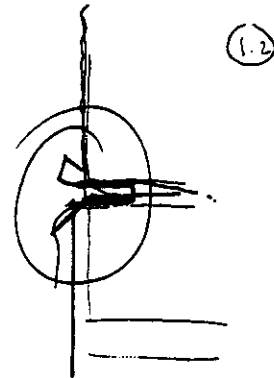
Appendix I: Protocol of Subject 1, continued, p.2

39. Concrete here
40. Brick cavity wall [points to front of bldg.]
41. You're looking at details to be generated for the contractor to build..
42. but allow him enough flexibility to adapt to conditions at the site.
43. Accessibility to tools
44. (Room for hammer and screw driver)
45. There's a question about surface detail
46. What would he need to allow that (flashing and counter-flashing) to be fastened to that brick? [looking at flashing and counter flashing detail over the roof]
47. [E: what are you looking at?]
48. [E: What do you see?]
50. I'm looking at flashing-counter-flashing details over the roof above the entry .
51. I see..
52. I don't see any flashing in detail
53. The counter-flashing apparently being recessed into the joint
54. That's typical
55. However, what they've done is..
56. they've taken the top of that flashing..
57. and bought it back out
58. What that's going to allow is ..
59. the water coming down that wall..
60. hit that..[shows flashing]
61. and penetrate back into the wall.
62. I'm concerned about that
63. I would change that flashing..
64. so that you would have..
65. What I would do is..
66. You would have masonry [drwg #1.1]
67. Then you can bring flashing in like this.
68. A lot of times what they do is
69. They'll put a piece of wood on here
70. And will take this over
71. Sometimes they will taper the top of the wood
72. So that you can bring this over
73. .. and bend this over
74. (no) sharp edges
75. And then bring this down
76. with a continuous cleat inside
77. so that the flashing comes up to this point, what ever..
78. and on down the roof.
79. And then this continuous cleat..
80. which is a piece fastened to this wood..
81. will receive the end of the flashing.
82. This then goes back ...
83. and it keeps from blowing up in the wind..
84. the rain..
85. as these things get older they will flop out..



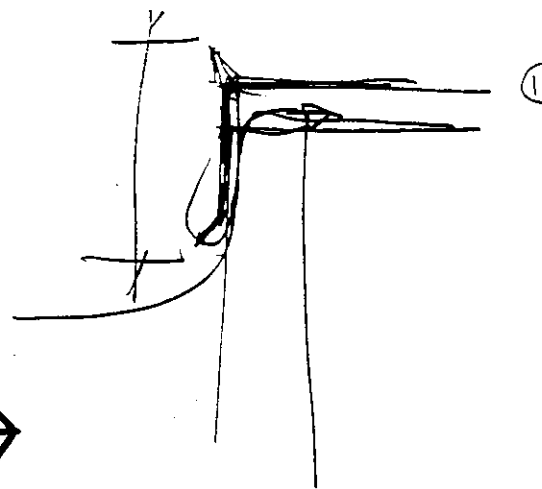
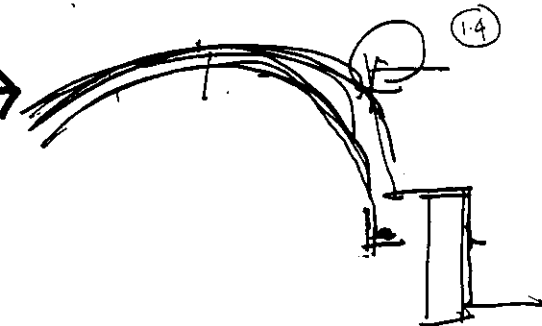
Appendix I: Protocol of Subject 1, continued, p.3

- 86. If they are not attached.
- 87. If you nail through..
- 88. you automatically put a hole through ..
- 89. which water (what ever) will go through.
- 90. So you try to keep that thing as watertight as possible..
- 91. You hide your nail behind.
- 92. You hide your cleat under it.
- 93. You bring this back.
- 94. and you take this back into your joint
- 95. A lot of time..
- 96. they use what they call lead wedge..
- 97. which will wedge it back in there.
- 98. You try to take it back far enough..
- 99. so that you have a raked joint..
- 100. and they will put caulking in there.
- 101. When the water comes down
- 102. it will hit that and run back down.
- 103. It won't get back in there.
- 104. [E: what are you looking at?]
- 105. [E What do you see?]
- 106. What was wrong is..
- 107. something at much smaller detail..
- 108. that (I can't see the flashing.
- 109. I see the counter-flashing)
- 110. What I think I see there [drwg #1.2] →
- 111. looks like its gone back in..
- 112. and then comes back out.
- 113. There is a piece up like this..
- 114. and then down the wall.
- 115. Now that will anchor here. in the wall
- 116. and then you can put a lead wedge back in your anchor.
- 117. But you can see water coming down in here..
- 118. will work its way back in there..
- 119. and create a problem.
- 120. So, what I've done by this is I've taken..
- 121. sloped that off..
- 122. so it goes down.
- 123. Water will get in anywhere.
- 124. My biggest concerns with detailing is water
- 125. By doing this,
- 126. we're putting caulking in this raked joint
- 127. and bringing this in here.
- 128. Tapering the top of this
- 129. you allow the water to hit this
- 130. and down
- 131. and around
- 132. [E: Why do you say that?]
- 133. (the problem here with water is it's combined with wind..
- 134. wind blows it against the wall..
- 135. and all the joints)
- 136. Snow..
- 137. I see another problem, right away..
- 138. right here [points to skylight, drwg #1.3] →



Appendix I: Protocol of Subject 1, continued, p.4

139. This is not flashed properly.
140. You'll get a certain amount of drifting snow here [drawing # 1.3]
141. And when in Spring it melts,
142. you get super saturated snow and slush
143. And it will work itself in here .
144. So you wan'a make this joint and this joint as watertight as possible.
145. In fact, what I'd do is
146. I would suggest to the architect
147. to try to move this [skylight] down a bit
148. to give more room for clearance.
149. Obviously, it will affect the design
150. You can't move it all the way.
151. So, I'll move it there [draws] →
152. Right here I would do the same thing.
153. Here.. [pointing to the end of the skylight]
154. there is no flashing there.
155. [E: How would you do it?]
156. I would say..
157. I would cop out on that..
158. and buy a prefabricated skylight.
159. I don't know if I can duplicate it.
160. What they'll have generically is a curve..
161. that comes down and flashes over..
162. and up under..
163. which is integral with the dome..
164. or whatever.
165. They'll be double for insulation..
166. whatever.
167. Inside, they will have a gutter..
168. which will take whatever condensation that occurs on the inside..
169. and will lay there..
170. and later will evaporate.
171. This detail will be much better to put a prefabricated skylight in there.
172. They are not that expensive.
173. Going on down..
174. All the details could be ok but when you blow them up..
175. for example, this facia detail with the flashing coming down here.
176. Again, you can get water blowing up underneath that..
177. so, I would bring that down more [drwg. 1.5] →
178. Facia piece here.
179. Sheating here.
180. I will bring down my flashing down more like this..
190. so that water would not..
191. There is tar and gravel..
192. so, you have a gravel stop up here.
193. I'll bring that down further.
194. You don't want it down too far.
195. You want to see the wood.
196. That's an architectural detail.



Appendix I: Protocol of Subject 1, continued, p.5

- 197. This dimension would have to be worked out with the architect...
- 198. as to how deep that would be.
- 199. But you wan'a have it far enough so that..
- 200. you don't have water blowing up in there..
- 201. or you don't have water migrate up in there.
- 202. Right now, it is not down far enough to protect the other side of that gravel stop.
- 203. [E: What do you see?]
- 204. I'm looking at the wall section..
- 205. insulation..
- 206. pretty typical..
- 207. assuming this is gip-board.
- 208. The top of your wall..
- 209. there is a problem right here, perhaps.
- 210. This gets into the fine line.
- 211. You need to ventilate this.
- 212. They've got a continuous screen here [pointing to under the fascia area]
- 213. But you're not gon'a get any ventilation..
- 214. unless you have it vented right here [points to sides of skylight]
- 215. Right now, I don't know how I could solve it..
- 216. but you have to get the ventilation to go out and around, up to this point.
- 217. Maybe we use a prefabricated, premanufactured, continuous vent.
- 218. Looks like a ridge vent put up here.
- 219. You have to work it into the back here..
- 220. so this would vent itself [drwg 1.6] →
- 221. [E: what do you see?]
- 222. I saw the continuous screen vent.
- 223. You got the insulation here
- 224. You got what appears to be a kitchen down here
- 225. So you will get a lot of humidity.
- 226. You're gon'a get the cold air out here.
- 227. You're gon'a get the condensation here. [points to the skylight]
- 228. With moisture here.
- 229. So you wan'a get that back out..
- 230. so you you wan'a ventilate it..
- 231. so I would change this detail up here.
- 232. Vent it..
- 233. with a continuous prefab vent..
- 234. a ridge vent..
- 235. a wall vent..
- 236. to get that vented out.
- 237. [E: why there?]
- 238. The reason why I wan'a put it here is..
- 239. obviously this appears to be not a continuous skylight
- 240. In plan you will have [drwg. 1.7]
- 241. And I wan'a get this whole thing ↓

16
↑
○

Appendix I: Protocol of Subject 1, continued, p.6

- ventilated ..
- 242. and you don't want to stop it there [points to the skylight]
 - 243. I also look for a vapor barrier..
 - 244. up in here [points to roof cavity]
 - 245. I don't see it here..
 - 246. unless the specification calls for it.
 - 247. The insulation is here.
 - 248. The water barrier could be on one side..
 - 249. aluminum or something.
 - 250. Going on down..[points to wall section]
 - 251. This is stucco.
 - 252. I have to look at the stucco detail
 - 253. Looking for joints..
 - 254. for expansion and contraction.
 - 255. Elevations.
 - 256. I want to make sure that we have some joints here [points to wall]
 - 257. Or otherwise that's gon'a crack..
 - 258. and I'm not sure what the detail looks like..
 - 259. as far as the stucco..
 - 260. or with reinforcement in that..
 - 261. or that its typical stucco..
 - 262. or ... wall..
 - 263. or drivit..
 - 264. one of those.
 - 265. But I'm more concerned with this wood shoe down here.
 - 266. I see another problem with water.
 - 267. Water can come down here and go through.
 - 268. So I would change this.
 - 269. I would bring..

26:32 ...

