The Science of Design: Creating the Artificial
Author(s): Herbert A. Simon
Source: *Design Issues*, Vol. 4, No. 1/2, Designing the Immaterial Society (1988), pp. 67-82
Published by: The MIT Press
Stable URL: http://www.jstor.org/stable/1511391
Accessed: 15-11-2016 12:38 UTC

Herbert A. Simon

The Science of Design: Creating the Artificial

between design and artificiality, a topic that is even more alive today than a quarter of a century ago, as the essays in this special edition of *Design Issues* show.

Historically and traditionally, it has been the task of the science disciplines to teach about natural things: how they are and how they work. It has been the task of engineering schools to teach about artificial things: how to make artifacts that have desired properties and how to design.

Engineers are not the only professional designers. Everyone designs who devises courses of action aimed at changing existing situations into preferred ones. The intellectual activity that produces material artifacts is no different fundamentally from the one that precribes remedies for a sick patient or the one that devises a new sales plan for a company or a social welfare policy for a state. Design, so construed, is the core of all professional training: it is the principal mark that distinguishes the professions from the sciences. Schools of engineering, as well as schools of architecture, business, education, law, and medicine, are all centrally concerned with the process of design.

In view of the key role of design in professional activity, it is ironic that in this century the natural sciences have almost driven the sciences of the artificial from professional school curricula. Engineering schools have become schools of physics and mathematics; medical schools have become schools of biological science; business schools have become schools of finite mathematics. The use of adjectives like "applied" conceals, but does not change, the fact. It simply means that in the professional schools those topics are selected from mathematics and the natural sciences for emphasis which are thought to be most nearly relevant to professional practice. It does not mean that design is taught, as distinguished from analysis.

The movement toward natural science and away from the sciences of the artificial has proceeded further and faster in engineering, business, and medicine than in the other professional fields I have mentioned, though it has by no means been absent from schools of law, journalism, and library science. The stronger universities are more deeply affected than the weaker, and the graduate programs more than the undergraduate. Few doctoral dissertations in first-rate professional schools today deal with genuine design problems, as distinguished from problems in solid-state physics or stochastic processes. I have to make partial exceptions — for reasons I shall mention — of dissertations in computer science and management science, and there are undoubtedly some others, for example, in chemical engineering.

Such a universal phenomenon must have a basic cause. It does have a very obvious one. As professional schools, including the independent engineering schools, are more and more absorbed into the general culture of the university, they hanker after academic respectability. In terms of the prevailing norms, academic respectability calls for subject matter that is intellectually tough, analytic, formalizable, and teachable. In the past much, if not most, of what we knew about design and about the artificial sciences was intellectually soft, intuitive, informal, and cookbooky. Why would anyone in a university stoop to teach or learn about designing machines or planning market strategies when he could concern himself with solid-state physics? The answer has been clear: he usually wouldn't.

The problem is widely recognized in engineering and medicine today and to a lesser extent in business. Some do not think it a problem, because they regard schools of applied science as a superior alternative to the trade schools of the past. If that were the choice, we could agree.[1] But neither alternative is satisfactory. The older kind of professional school did not know how to educate for professional design at an intellectual level appropriate to a university; the newer kind of school has nearly abdicated responsibility for training in the core professional skill. Thus we are faced with a problem of devising a professional school that can attain two objectives simultaneously: education in both artificial and natural science at a high intellectual level. This too is a problem of design — organizational design.

The kernel of the problem lies in the phrase "artificial science." In my previous chapters I have shown that a science of artificial phenomena is always in imminent danger of dissolving and vanishing. The peculiar properties of the artifact lie on the thin interface between the natural laws within it and the natural laws without. What can we say about it? What is there to study besides the boundary sciences — those that govern the means and the task environment?

The artificial world is centered precisely on this interface between the inner and outer environments; it is concerned with attaining goals by adapting the former to the latter. The proper study of those who are concerned with the artificial is the way in which that adaptation of means to environments is brought about — and central to that is the process of design itself. The professional schools will reassume their professional responsibilities just to the degree that they can discover a science of design, a body of intellectually tough, analytic, partly formalizable, partly

---

1) That was in fact the choice in our engineering schools a generation ago. The schools needed to be purged of vocationalism; and a genuine science of design did not exist even in a rudimentary form as an alternative. Hence the road forward was the road toward introducing more fundamental science. Karl Taylor Compton was one of the prominent leaders in this reform, which was a main theme in his presidential inaugural address at MIT in 1930: " I hope . . . that increasing attention in the Institute may be given to the fundamental sciences; that they may achieve as never before the spirit and results of research; that all courses of instruction may be examined carefully to see where training in details has been unduly emphasized at the expense of the more powerful training in all-embracing fundamental principles."

Notice that President Compton's emphasis was on "fundamental," an emphasis as sound today as it was in 1930. What I am urging in this essay is not a departure from the fundamental but an inclusion in the curriculum of the fundamental in engineering along with the fundamental in natural science. That was not possible in 1930; but it is possible today.

empirical, teachable doctrine about the design process.

It is the thesis of this chapter that such a science of design not only is possible but is actually emerging at the present time. It has already begun to penetrate the engineering schools, particularly through programs in computer science and "systems engineering," and business schools through management science. Perhaps it also has beach-heads in other professional curricula, but these are the two with which I am most familiar. We can already see enough of its shape to predict some of the important ways in which engineering schools tomorrow will differ from departments of physics, and business schools from departments of economics and psychology. Let me now turn from questions of university organization to the substance of the matter.

THE LOGIC OF DESIGN: FIXED ALTERNATIVES

We must start with some question of logic.[2] The natural sciences are concerned with how things are. Ordinary systems of logic —the standard propositional and predicate calculi, say — serve these sciences well. Since the concern of standard logic is with declarative statements, it is well suited for assertions about the world and for inferences from those assertions.

Design, on the other hand, is concerned with how things ought to be, with devising artifacts to attain goals. We might question whether the forms of reasoning that are appropriate to natural science are suitable also for design. One might well suppose that introduction of the verb "should" may require additional rules of inference, or modification of the rules already imbedded in declarative logic.

Paradoxes of Imperative Logic

Various "paradoxes" have been constructed to demonstrate the need for a distinct logic of imperatives, or a normative, deontic logic. In ordinary logic from "Dogs are pets" and "Cats are pets," one can infer "Dogs and cats are pets." But from "Dogs are pets," "Cats are pets," and "You should keep pets," one can infer "You should keep cats and dogs?" And from "Give me needle and thread!" can one deduce, in analogy with declarative logic, "Give me needle or thread!"? Easily frustrated people would perhaps rather have neither needle nor thread than one without the other, and peaceloving people, neither cats nor dogs, rather than both.

As a response to these challenges of apparent paradox, there have been developed a number of constructions of modal logic for handling "should," "shalts," and "oughts" of various kinds. I think it is fair to say that none of these systems has been sufficiently developed or sufficiently widely applied to demonstrate that it is adequate to handle the logical requirements of the process of design.

---

2) I have treated the question of logical formalism for design at greater length in two earlier papers: "The Logic of Rational Decision," *British Journal for the Philosophy of Science*, 16 (1965): 169-186; and "The Logic of Heuristic Decision Making," in Nicholas Rescher (ed.), *The Logic of Decision and Action* (Pittsburgh: University of Pittsburgh Press, 1967), pp. 1-35. The present discussion is based on these two papers, which have been reprinted as chapters 3.1 and 3.2 in my *Models of Discovery* (Dordrecht: D. Reidel Pub. Co., 1977).

Fortunately, such a demonstration is really not essential, for it can be shown that the requirements of design can be met fully by a modest adaptation of ordinary declarative logic. Thus a special logic of imperatives is unnecessary.[3]

Reduction to Declarative Logic

The easiest way to discover what kinds of logic are needed for design is to examine what kinds of logic designers use when they are being careful about their reasoning. Now there would be no point in doing this if designers were always sloppy fellows who reasoned loosely, vaguely, and intuitively. Then we might say that whatever logic they used was not the logic they *should* use.

However, there exists a considerable area of design practice where standards of rigor in inference are as high as one could wish. I refer to the domain of so-called "optimization methods," most highly developed in statistical decision theory and management science but acquiring growing importance also in engineering design theory. The theories of probability and utility, and their intersection, have received the painstaking attention not only of practical designers and decision makers but also of a considerable number of the most distinguished logicians and mathematicians of the present and recent past generations. F. P. Ramsey, B. de Finetti, A. Wald, J. von Neumann, J. Neyman, K. Arrow, and L. J. Savage are examples.

The logic of optimization methods can be sketched as follows: The "inner environment" of the design problem is represented by a set of given alternatives of action. The alternatives may be given *in extenso*: more commonly they are specified in terms of *command variables* that have defined domains. The "outer environment" is represented by a set of parameters, which may be known with certainty or only in terms of a probability distribution. The goals for adaptation of inner to outer environment are defined by a utility function — a function, usually scalar, of the command variables and environmental parameters — perhaps supplemented by a number of constraints (inequalities, say, between functions of the command variables and environmental parameters). The optimization problem is to find an admissible set of values of the command variables, compatible with the constraints, that maximize the utility function for the given values of the environmental parameters. (In the probabilistic case we might say, "maximize the expected value of the utility function," for instance, instead of "maximize the utility function.")

A stock application of this paradigm is the so-called "diet problem" shown in figure 6. A list of foods is provided, the command variables being quantities of the various foods to be included in the diet. The environmental parameters are the prices and nutritional contents (calories, vitamins, minerals, and so on) of each of the foods. The utility function is the cost (with a minus

---

3) I should like to underline the word "unnecessary." When I said something like this in another place (the second paper mentioned in the previous footnote), an able logician, who had specialized in modal logics, accused me of asserting that modal logics were "impossible." Now this is patently false: modal logics can be shown to exist in the same way that giraffes can — namely, by exhibiting some of them. The question is not whether they exist but what they are good for. A modal logician should have no difficulty in distinguishing "non-necessity" from "impossibility."

Example:

Logical Terms		**The diet problem**
Command variables	("Means")	Quantities of foods
Fixed parameters	("Laws")	Prices of foods Nutritional contents
Constraints Utility function	("Ends")	Nutritional requirements — Cost of diet

Constraints characterize the inner environment; parameters characterize the outer environment.

Problem: Given the constraints and fixed parameters, find values of the command variables that maximize utility.

Fig. 6. The paradigm for imperative logic.

sign attached) of the diet, subject to the constraints, say, that it not contain more than 2,000 calories per day, that it meet specified minimum needs for vitamins and minerals, and that rutabaga not be eaten more than once a week. The constraints may be viewed as characterizing the inner environment. The problem is to select the quantities of foods that will meet the nutritional requirements and side conditions at the given prices for the lowest cost.

The diet problem is a simple example of a class of problems that are readily handled, even when the number of variables is exceedingly large, by the mathematical formalism known as linear programming. I shall come back to the technique a little later. My present concern is with the logic of the matter.

Since the optimization problem, once formalized, is a standard mathematical problem — to maximize a function subject to constraints — it is evident that the logic used to deduce the answer is the standard logic of the predicate calculus on which mathematics rests. How does the formalism avoid making use of a special logic of imperatives? It does so by dealing with sets of *possible worlds*: First consider all the possible worlds that meet the constraints of the outer environment; then find the particular world in the set that meets the remaining constraints of the goal and maximizes the utility function. The logic is exactly the same as if we were to adjoin the goal constraints and the maximation requirement, as new "natural laws," to the existing natural laws embodied in the environmental conditions.[4] We simply ask what values the command variables *would* have in a world meeting all these conditions and conclude that these are the values the command variables *should* have.

Computing the Optimum

Our discussion thus far has already provided us with two central topics for the curriculum in the science of design:

---

[4] The use of the notion of "possible worlds" to embed the logic of imperatives in declarative logic goes back at least to Jürgen Jürgensen, "Imperatives and Logic," *Erkenntnis*, 7 (1937-1938): 288-296. See also my *Administrative Behavior* (New York: Macmillan, 1947), chapter 3. More recently this same idea has been used by several logicians to construct a formal bridge between the predicate calculus and modal logic by means of so- called semantic or model-

1. *Utility theory and statistical decision theory as a logical framework for rational choice among given alternatives.*
2. *The body of techniques for actually deducing which of the available alternatives is the optimum.*

Only in trivial cases is the computation of the optimum alternative an easy matter. If utility theory is to have application to real-life design problems, it must be accompanied by tools for actually making the computations. The dilemma of the rational chess player is familiar to all. The optimal strategy in chess is easily demonstrated: simply assign a value of +1 to a win, 0 to a draw, -1 to a loss; consider all possible courses of play; minimax backward from the outcome of each, assuming each player will take the most favorable move at any given point. This procedure will determine what move to make now. The only trouble is that the computations required are astronomical (the number 10^{120} is often mentioned in this context) and hence cannot be carried out — not by humans, not by existing computers, not by prospective computers.

A theory of design as applied to the game of chess would encompass not only the utopian minimax principle but also some practicable procedures for finding good moves in actual board positions in real time, within the computational capacities of real human beings or real computers. No exceptionally good procedures of this kind exist today, other than those stored in the memories of grandmasters, but there is at least one computer program that plays at the level of an expert or a weak master —that is, better than all save a few hundred human players.

The second topic then for the curriculum in the science of design consists in the efficient computational techniques that are available for actually finding optimum courses of action in real situations, or reasonable approximations to real situations. As I mentioned in chapter 2, that topic has a number of important components today, most of them developed — at least to the level of practical application — within the past 25 years. These include linear programming theory, dynamic programming, geometric programming, queuing theory, and control theory.

THE LOGIC OF DESIGN; FINDING ALTERNATIVES

When we take up the case where the design alternatives are not given in any constructive sense but must be synthesized, we must ask once more whether any new forms of reasoning are involved in the synthesis, or whether again the standard logic of declarative statements is all we need.

In the case of optimization we asked: "Of all possible worlds (those attainable for some admissible values of the action variables), which is the best (yields the highest value of the criterion function)?" As we saw, this is a purely empirical question, calling

theoretic methods. See, for example, Richard Montague, "Logical Necessity, Physical Necessity, Ethics, and Quantifiers," *Inquiry*, 4 (1960): 259-269, where references are also given to work of Stig Kanger and Saul Kripke; and Jaakko Hintikka, "Modality and Quantification," *Theoria*, 27 (1961): 119-128. While these model-theoretic proposals are basically sound, none of them seems yet to have given adequate attention to the special role played in the theory by command variables and criterial constraints.

only for facts and ordinary declarative reasoning to answer it.

In this case, where we are seeking a satisfactory alternative, once we have found a candidate we can ask: "Does this alternative satisfy all the design criteria?" Clearly this is also a factual question and raises no new issues of logic. But how about the process of *searching* for candidates? What kind of logic is needed for the search?

Means-Ends Analysis

The condition of any goal-seeking system is that it is connected to the outside environment through two kinds of channels: the afferent, or sensory, channels through which it receives information about the environment and the efferent, or motor, channels through which it acts on the environment.[5] The system must have some means of storing in its memory information about states of the world — afferent, or sensory, information —and information about actions — efferent, or motor, information. Ability to attain goals depends on building up associations, which may be simple or very complex, between particular changes in states of the world and particular actions that will (reliably or not) bring these changes about. In chapter 4 we described these associations as productions.

Except for a few built-in reflexes, an infant has no basis for correlating his sensory information with his actions. A very important part of his early learning is that particular actions or sequences of actions will bring about particular changes in the state of the world as he senses it. Until he builds up this knowledge, the world of sense and the motor world are two entirely separate, entirely unrelated worlds. Only as he begins to acquire experience as to how elements of the one relate to elements of the other can he act purposefully on the world.

The computer problem-solving program called GPS, designed to model some of the main features of human problem solving, exhibits in stark form how goal-directed action depends on building this kind of bridge between the afferent and the efferent worlds. On the afferent, or sensory, side, GPS must be able to represent desired situations or desired objects as well as the present situation. It must be able also to represent *differences* between the desired and the present. On the efferent side, GPS must be able to represent *actions* that change objects or situations. To behave purposefully, GPS must be able to select from time to time those particular actions that are likely to remove the particular differences between desired and present states that the system detects. In the machinery of GPS, this selection is achieved through a *table of connections*, which associates with each kind of detectable difference those actions that are relevant to reducing that difference. These are its associations, in the form of

5) Notice that we are not saying that the two kinds of channels operate independently of each other, since they surely do not in living organisms, but that we can distinguish conceptually, and to some extent neurologically, between the incoming and outgoing flows.

productions, which relate the afferent to the efferent world. Since reaching a goal generally requires a sequence of actions, and since some attempts may be ineffective, GPS must also have means for detecting the progress it is making (the changes in the differences between the actual and the desired) and for trying alternate paths.

Now the real worlds to which problem solvers and designers address themselves are seldom completely additive in this sense. Actions have side consequences (may create new differences) and sometimes can only be taken when certain side conditions are satisfied (call for removal of other differences before they become applicable). Under these circumstances one can never be certain that a partial sequence of actions that accomplishes *certain* goals can be augmented to provide a solution that satisfied *all* the conditions and attains *all* the goals (even though they be satisficing goals) of the problem.

For this reason problem-solving systems and design procedures in the real world do not merely *assemble* problem solutions from components but must *search* for appropriate assemblies. In carrying out such a search, it is often efficient to divide one's eggs among a number of baskets — that is, not to follow out one line until it succeeds completely or fails definitely but to begin to explore several tentative paths, continuing to pursue a few that look most promising at a given moment. If one of the active paths begins to look less promising, it may be replaced by another that had previously been assigned a lower priority.

Our discussion of design when the alternatives are not given has yielded at least three additional topics for instruction in the science of design:

3. *Adaptation of standard logic to the search for alternatives.* Design solutions are sequences of actions that lead to possible worlds satisfying specified constraints. With satisficing goals the sought-for possible worlds are seldom unique; the search is for *sufficient*, not *necessary*, actions for attaining goals.
4. *The exploitation of parallel, or near-parallel, factorizations of differences.* Means-ends analysis is an example of a broadly applicable problem-solving technique that exploits this factorization.
5. *The allocation of resources for search to alternative, partly explored action sequences.* I should like to elaborate somewhat on this last-mentioned topic.

DESIGN AS RESOURCE ALLOCATION

There are two ways in which design processes are concerned with the allocation of resources. First, conservation of scarce resources may be one of the criteria for a satisfactory design. Second, the design process itself involves management of the resources of the designer, so that his efforts will not be dissipated unnecessarily in following lines of inquiry that prove fruitless.

There is nothing special that needs to be said here about resource conservation — cost minimization, for example, as a design criterion. Cost minimization has always been an implicit consideration in the design of engineering structures, but until a few years ago it generally *was* only implicit, rather than explicit. More and more cost calculations have been brought explicitly into the design procedure, and a strong case can be made today for training design engineers in that body of technique and theory that economists know as "cost-benefit analysis."

THE SHAPE OF DESIGN: HIERARCHY

In my first chapter I gave some reasons why complex systems might be expected to be constructed in a hierarchy of levels, or in a boxes-within-boxes form. The basic idea is that the several components in any complex system will perform particular subfunctions that contribute to the over-all function. Just as the "inner environment" of the whole system may be defined by describing its functions, without detailed specification of its mechanisms, so the "inner environment" of each of the subsystems may be defined by describing the functions of that subsystem, without detailed specification of *its* submechanisms.[6]

To design such a complex structure, one powerful technique is to discover viable ways of decomposing it into semi-independent components corresponding to its many functional parts. The design of each component can then be carried out with some degree of independence of the design of others, since each will affect the others largely through its function and independently of the details of the mechanisms that accomplish the function.[7]

There is no reason to expect that the decomposition of the complete design into functional components will be unique. In important instances there may exist alternative feasible decompositions of radically different kinds. This possibility is well known to designers of administrative organizations, where work can be divided up by subfunctions, by subprocesses, by subareas, and in other ways. Much of classical organization theory in fact was concerned precisely with this issue of alternative decompositions of a collection of interrelated tasks.

The Generator-Test Cycle

One way of considering the decomposition, but acknowledging that the interrelations among the components cannot be ignored completely, is to think of the design process as involving, first, the generation of alternatives and, then, the testing of these alterna-

[6] I have developed this argument at greater length in my essay The Architecture of Complexity," chapter 7.

[7] This approach to the design of complex structures has been explored by Christopher Alexander in *Notes on the Synthesis of Form* (Cambridge: Harvard University Press, 1967). He has also presented in his book some automated procedures for finding plausible decompositions once the matrix of interconnections of component functions has been specified.

tives against a whole array of requirements and constraints. There need not be merely a single generate-test cycle, but there can be a whole nested series of such cycles. The generators implicitly define the decomposition of the design problem, and the tests guarantee that important indirect consequences will be noticed and weighed. Alternative decompositions correspond to different ways of dividing the responsibilities for the final design between generators and tests.

To take a greatly oversimplified example, a series of generators may generate one or more possible outlines and schemes of fenestration for a building, while tests may be applied to determine whether needs for particular kinds of rooms can be met within the outlines generated. Alternatively the generators may be used to evolve the structure of rooms, while tests are applied to see whether they are consistent with an acceptable over-all shape and design. The house can be designed from the outside in or from the inside out.[8]

Alternatives are also open, in organizing the design process, as to how far development of possible subsystems will be carried before the over-all coordinating design is developed in detail, or vice-versa, how far the over-all design should be carried before various components, or possible components, are developed. These alternatives of design are familiar to architects. They are familiar also to composers, who must decide how far the architectonics of a musical structure will be evolved before some of the component musical themes and other elements have been invented. Computer programmers face the same choices, between working downward from executive routines to subroutines or upward from component subroutines to a coordinating executive.

A theory of design will include principles — most of which do not yet exist — for deciding such questions of precedence and sequence in the design process.

Process as a Determinant of Style

When we recall that the process will generally be concerned with finding a satisfactory design, rather than an optimum design, we see that sequence and the division of labor between generators and tests can affect not only the efficiency with which resources for designing are used but also the nature of the final design as well. What we ordinarily call "style" may stem just as much from these decisions about the design process as from alternative emphases on the goals to be realized through the final design. An architect who designs buildings from the outside in will arrive at quite different buildings from one who designs from the inside out, even though both of them might agree on the characteristics that a satisfactory building should possess.

When we come to the design of systems as complex as cities, or

---

8) I am indebted to John Grason for many ideas on the topic of this section. J. Grason, "Fundamental Description of a Floor Plan Design Program," EDRAl, *Proceedings of the First Environmental Design Association Conference*, H. Sanoff and S. Cohn (eds.), North Carolina State University, 1970.

buildings, or economies, we must give up the aim of creating systems that will optimize some hypothesized utility function, and we must consider whether differences in style of the sort I have just been describing do not represent highly desirable variants in the design process rather than alternatives to be evaluated as "better" or "worse." Variety, within the limits of satisfactory constraints, may be a desirable end in itself, among other reasons, because it permits us to attach value to the search as well as its outcome — to regard the design process as itself a valued activity for those who participate in it.

We have usually thought of city planning as a means whereby the planner's creative activity could build a system that would satisfy the needs of a populace. Perhaps we should think of city planning as a valuable creative activity in which many members of a community can have the opportunity of participating — if we have wits to organize the process that way. I shall have more to say on these topics in the next chapter.

However that may be, I hope I have illustrated sufficiently that both the shape of the design and the shape and organization of the design process are essential components of a theory of design. These topics constitute the sixth item in my proposed curriculum in design:

6. *The organization of complex structures and its implication for the organization of design processes.*

REPRESENTATION OF THE DESIGN

I have by no means surveyed all facets of the emerging science of design. In particular I have said little about the influence of problem representation on design. Although the importance of the question is recognized today, we have little systematic knowledge about it. I shall cite one example, to make clear what I mean by "representation."

Here are the rules of a game, which I shall call number scrabble. The game is played by two people with nine cards — let us say the ace through the nine of hearts. The cards are placed in a row, face up, between the two players. The players draw alternately, one at a time, selecting any one of the cards that remain in the center. The aim of the game is for a player to make up a "book," that is, a set of exactly three cards whose spots add to 15, before his opponent can do so. The first player who makes a book wins; if all nine cards have been drawn without either player making a book, the game is a draw.

What is a good strategy in this game? How would you go about finding one? If the reader has not already discovered it for himself, let me show how a change in representation will make it easy to play the game well. The magic square here, which I introduced in the third chapter, is made up of the numerals from 1 through 9.

4	9	2
3	5	7
8	1	6

Each row, column, or diagonal adds to 15, and every triple of these numerals that add to 15 is a row, column, or diagonal of the magic square. From this, it is obvious that "making a book" in number scrabble is equivalent to getting "three in a row" in a game of tic-tac-toe. But most people know how to play tic-tac-toe well, hence can simply transfer their usual strategy to number scrabble.[9]

Problem Solving as Change in Representation

That representation makes a difference is a long-familiar point. We all believe that arithmetic has become easier since Arabic numerals and place notation replaced Roman numerals, although I know of no theoretic treatment that explains why.[10]

That representation makes a difference is evident for a different reason. All mathematics exhibits in its conclusions only what is already implicit in its premises, as I mentioned in a previous chapter. Hence all mathematical derivation can be viewed simply as change in representation, making evident what was previously true but obscure.

This view can be extended to all of problem solving — solving a problem simply means representing it so as to make the solution transparent.[11] If the problem solving could actually be organized in these terms, the issue of representation would indeed become central. But even if it cannot — if this is too exaggerated a view —a deeper understanding of how representations are created and how they contribute to the solution of problems will become an essential component in the future theory of design.

Spatial Representation

Since much of design, particularly architectural and engineering design, is concerned with objects or arrangements in real Euclidean two-dimensional or three-dimensional space, the representation of space and of things in space will necessarily be a central topic in a science of design. From our previous discussion of visual perception, it should be clear that "space" inside the head of the designer of the memory of a computer may have very different properties from a picture on a paper or a three-dimensional model.

These representational issues have already attracted the attention of those concerned with computer-aided design — the cooperation

9) Number scrabble is not the only isomorph of tic-tac-toe. John A. Michon has described another, JAM, which is the dual of tic-tac-toe in the sense of projective geometry. That is, the rows, columns, and diagonals of tic-tac-toe become points in JAM, and the squares of the former become line segments joining the points. The game is won by "jamming" all the segments through a point — a move consists of seizing or jamming a single segment. Other isomorphs of tic-tac-toe are known as well.

10) My colleague, Allen Newell, has been investigating this question. I shall not try to anticipate his answer.

11) Saul Amarel, "On the Mechanization of Creative Processes, IEEE *Spectrum* 3 (April 1966): 112-114.

of human and computer in the design process. As a single example, I may mention Ivan Sutherland's SKETCHPAD program, which allows geometric shapes in terms of constraints, to which they then conform.[12]

Geometric considerations are also prominent in the attempts to automate completely the design, say, of printed or etched circuits, or of buildings. Grason, for example, in a system for designing house floor plans, constructs an internal representation of the layout that helps one decide whether a proposed set of connections among rooms, selected to meet design criteria for communication, and so on, can be realized in a plane.[13]

The Taxonomy of Representation

An early step toward understanding any set of phenomena is to learn what kinds of things there are in the set — to develop a taxonomy. This step has not yet been taken with respect to representations. We have only a sketchy and incomplete knowledge of the different ways in which problems can be represented and much less knowledge of the significance of the differences.

In a completely pragmatic vein we know that problems can be described verbally, in natural language. They often can be described mathematically, using standard formalisms of algebra, geometry, set theory, analysis, or topology. If the problems relate to physical objects, they (or their solutions) can be represented by floor plans, engineering drawings, renderings, or three-dimensional models. Problems that have to do with actions can be attacked with flow charts and programs.

Other items most likely will need to be added to the list, and there may exist more fundamental and significant ways of classifying its members. But even though our classification is incomplete, and perhaps superficial, we can begin to build a theory of the properties of these representations. A number of topics in the growing theories of machines and of programming languages may give us some notion of the directions that a theory of representations — at least on its more formal side — may take.[14] These topics can also provide, at the beginning, some of the substance for the final subject in our program on the theory of design:

7. *Alternative representations for design problems.*

SUMMARY — TOPICS IN THE THEORY OF DESIGN

My main goal in this chapter has been to show that there already exist today a number of components of a theory of design and a substantial body of knowledge, theoretical and empirical, relating to each. As we draw up our curriculum in design — in the science of the artificial — to take its place by the side of natural science in

---

12) I. E. Sutherland, "SKETCHPAD, A Man-Machine Graphical Communication System," *Proceedings, AFIPS Spring Joint Computer Conference, 1963* (Baltimore: Spartan Books), pp. 329-346.

13) See also C. E. Pfefferkorn, "The Design Problem Solver: A System for Designing Equipment or Furniture Layouts," in C. M. Eastman (ed.), *Spatial Synthesis in Computer-Aided Building Design* (London: Applied Science Publishers, 1975).

14) By way of example, see Marvin L. Minsky, *Computation: Finite and Infinite Machines* (Englewood Cliffs, NJ: Prentice-Hall, 1967), and Kenneth E. Iverson, *A Programming Language* (New York: Wiley, 1962).

the whole engineering curriculum, it includes at least the following topics:

The Evaluation of Designs

1. Theory of evaluation: utility theory, statistical decision theory
2. Computational methods:
   a. Algorithms for choosing *optimal* alternatives such as linear programming computations, control theory, dynamic programming
   b. Algorithms and heuristics for choosing *satisfactory* alternatives
3. **The Formal Logic of Design:** imperative and declarative logics

The Search For Alternatives

4. Heuristic search: factorization and means-ends analysis
5. Allocation of resources for search
6. **Theory of Structure and Design Organization:** hierarchic systems
7. **Representation of Design Problems**

In small segments of the curriculum — the theory of evaluation, for example, and the formal logic of design — it is already possible to organize the instruction within a framework of systematic, formal theory. In many other segments the treatment would be more pragmatic, more empirical.

But nowhere do we need to return or retreat to the methods of the cookbook that originally put design into disrepute and drove it from the engineering curriculum. For there exist today a considerable number of examples of actual design processes, of many different kinds, that have been defined fully and cast in the metal, so to speak, in the form of running computer programs: optimizing algorithms, search procedures, and special-purpose programs for designing motors, balancing assembly lines, selecting investment portfolios, locating warehouses, designing highways, diagnosing and treating diseases, and so forth.

Because these computer programs describe complex design processes in complete, painstaking detail, they are open to full inspection and analysis, or to trial by simulation. They constitute a body of empirical phenomena to which the student of design can address himself and which he can seek to understand. There is no question, since these programs exist, of the design process hiding behind the cloak of "judgment" or "experience." Whatever judgment or experience was used in creating the programs must now be incorporated in them and hence be observable. The programs are the tangible record of the variety of schemes that man has devised to explore his complex outer environment and to discover in that environment the paths to his goals.

ROLE OF DESIGN IN THE LIFE OF THE MIND

I have called my topic "the theory of design" and my curriculum a "program in design." I have emphasized its role as complement to the natural science curriculum in the total training of a professional engineer — or of any professional whose task is to solve problems, to choose, to synthesize, to decide.

But there is another way in which the theory of design may be viewed in relation to other knowledge. My third and fourth chapters were chapters on psychology — specifically on man's relation to his biological inner environment. The present chapter may also be construed as a chapter on psychology: on man's relation to the complex outer environment in which he seeks to survive and achieve.

All three chapters, so construed, have import that goes beyond the professional work of the man we have called the "designer." Many of us have been unhappy about the fragmentation of our society into two cultures. Some of us even think there are not just two cultures but a large number of cultures. If we regret that fragmentation, then we must look for a common core of knowledge that can be shared by the members of all cultures — a core that includes more significant topics than the weather, sports, automobiles, the care and feeding of children, or perhaps even politics. A common understanding of our relation to the inner and outer environments that define the space in which we live and choose can provide at least part of that significant core.

This may seem an extravagant claim. Let me use the realm of music to illustrate what I mean. Music is one of the most ancient of the sciences of the artificial, and was so recognized by the Greeks. Anything I have said about the artificial would apply as well to music, its composition or its enjoyment, as to the engineering topics I have used for most of my illustrations.

Music involves a formal pattern. It has few (but important) contacts with the inner environment; that is, it is capable of evoking strong emotions, its patterns are detectable by human listeners, and some of its harmonic relations can be given physical and physiological interpretations (though the esthetic import of these is debatable). As for the outer environment, when we view composition as a problem in design, we encounter just the same tasks of evaluation, of search for alternatives, and of representation that we do in any other design problem. If it pleases us, we can even apply to music some of the same techniques of automatic design by computer that have been used in other fields of design. If computer-composed music has not yet reached notable heights of esthetic excellence, it deserves, and has already received, serious attention from professional composers and analysts, who do not find it written in tongues alien to them.[15]

[15] L. A. Hillier and L. M. Isaacson's *Experi-*

mental Music (New York: McGraw-Hill, 1959), reporting experiments begun more than two decades ago, still provides a good introduction to the subject of musical composition, viewed as design. See also Walter R. Reitman, *Cognition and Thought* (New York: Wiley, 1965), chapter 6, "Creative Problem Solving: Notes from the Autobiography of a Fugue."

Undoubtedly there are tone-deaf engineers, just as there are mathematically ignorant composers. Few engineers and composers, whether deaf, ignorant, or not, can carry on a mutually rewarding conversation about the content of each other's professional work. What I am suggesting is that they *can* carry on such a conversation about design, can begin to perceive the common creative activity in which they are both engaged, can begin to share their experiences of the creative, professional design process.

Those of us who have lived close to the development of the modern computer through gestation and infancy have been drawn from a wide variety of professional fields, music being one of them. We have noticed the growing communication among intellectual disciplines that takes place around the computer. We have welcomed it, because it has brought us into contact with new worlds of knowledge — has helped us combat our own multiple-cultures isolation. This breakdown of old disciplinary boundaries has been much commented upon, and its connection with computers and the information sciences often noted.

But surely the computer, as a piece of hardware, or even as a piece of programmed software, has nothing to do directly with the matter. I have already suggested a different explanation. The ability to communicate across fields — the common ground — comes from the fact that all who use computers in complex ways are using computers to design or to participate in the process of design. Consequently we as designers, or as designers of design processes, have had to be explicit as never before about what is involved in creating a design and what takes place while the creation is going on.

The real subjects of the new intellectual free trade among the many cultures are our own thought processes, our processes of judging, deciding, choosing, and creating. We are importing and exporting from one intellectual discipline to another ideas about how a serially organized information-processing system like a human being — or a computer, or a complex of men and women and computers in organized cooperation — solves problems and achieves goals in outer environments of great complexity.

The proper study of mankind has been said to be man. But I have argued that man — or at least the intellective component of man — may be relatively simple, that most of the complexity of his behavior may be drawn from man's environment, from man's search for good designs. If I have made my case, then we can conclude that, in large part, the proper study of mankind is the science of design, not only as the professional component of a technical education but as a core discipline for every liberally educated person.

We gladly acknowledge Professor Simon and the MIT Press for granting the permission to reprint these pages