Chapter 5
Constraining Functional Architecture

> The [benefit] which rhyme has over blank verse . . . is, that it bounds and circumscribes the fancy. For imagination in a poet is a faculty so wild and lawless, that, like an high-ranging spaniel, it must have clogs tied to it, lest it outrun the judgment.
>
> John Dryden, dedication of *The Rival-Ladies*

The Level-of-Aggregation Problem

Given the central importance of functional architecture, the question we need to ask next is, How can one empirically constrain the functional architecture adopted in constructing models of cognitive processes? In approaching this question I shall concentrate on a particular aspect of the architecture, namely, the "level of aggregation" of its primitive operations. Beyond the sketch presented in chapter 3, no other aspects are discussed, such aspects as disciplining the control of processes (Is it serial or parallel? What constraints exist on the availability of resources, for example, workspace buffers? What initiates different processing? What controls are there over their sequencing? How transparent and modular are various aspects of the control structure?). Although those issues are also crucial to cognitive modeling, there is much less to be said about them at the moment, except that various alternatives are under investigation in computer science.

The *level of aggregation* defining the strong equivalence of programs in the Scott and Strachey sense already outlined was dictated by the abstract semantic model, particularly, what were considered the significant functions performed in the virtual machine. In the case of cognition, choosing the appropriate level of aggregation at which to model the process is one of the most difficult problems confronting theorists. Newell and Simon (1972) discuss this problem in relation to a study of human problem-solving. Their goal here was to account for the regularities in a problem-solving trace called a "problem behavior graph" constructed from an analysis of a "thinking out loud" protocol recorded from a subject. Specifically, the problem was to posit a set of operators that would account for the transitions among problem states or *states of knowledge* about the problem. Newell and Simon (1972, p. 186) cite several pitfalls accompanying a too global or too microscopic choice of operator. In the too-global case:

> It may be that the essential problem solving is done "inside" one or more of these [operators]. If this were so, the problem graph would have to be termed *superficial*, since, although a true enough description, it would not explicate the important processing steps. Note that the basic issue is not how much selection is performed within the components . . . but whether the selection requires problem solving: either search in another space, or some other as-yet-unspecified intellectual process.

In the too-microscopic case:

> It may be that the analysis has gone too far—is too *disaggregated*. . . . Excessive disaggregation would reveal itself in the capriciousness of various selections, viewed in terms of the local context, whenever the next action was in fact (but not explicitly) determined by the structure of the higher-level plan or method.

These cases are quoted because they suggest several issues that concern us in this book. First is the suggestion of an appropriate level at which the greatest regularity is to be found (see the beginning of chapter 1). Second, the claim is made that a too-fine-grained analysis loses regularities because the explanation (or determination) of each component is found at a higher level; or if, as in verbal protocols, the hierarchy itself is flattened (so that higher-and lower-level descriptors are mentioned in a linear narration), the determiner of a particular behavior may be far away in the linearized sequence. This case also poses the problem "we will be led to analyze in detail subprocesses that are not problematic for the subject, hence amount simply to doing what the task environment demands . . . " (p. 189). The notion that processes which are not difficult or problematical for the subject need not themselves be explicated in detail also appears in the quotation for the first case. With it is introduced my third and most important point. Newell and Simon claim that an elementary operation can be as selective or complex as is appropriate, providing it is not an instance of further problem-solving. This sounds like Dan Dennett's requirement (Dennett, 1971) that the work of each homunculus be done by "committees of stupider homunculi." It is stronger than that, however, for *any* problem-solving activity hidden in an operator—even if the problem is substantially easier than the original one—would lead to superficiality in the account (and perhaps to loss of systematicity). We want to encapsulate as a basic operation nothing which itself involves nontrivial processing and which thus, in turn, must be explained.

Exactly what kinds of processes qualify as primitive (hence, explanatory) operations is a question to which the remainder of this chapter is devoted. Clearly, the criterion of being an instance of problem-solving is insufficiently precise; at least, it provides no methodological criterion, unless we already know when problem-solving is occurring or unless we know in advance what are the basic operators or states of the "problem space." It is equally clear that a number of fundamental distinctions hinge on there being such criteria, among these the ability to distinguish between an ad hoc model (with numerous free, empirical parameters) and a principled model (in which independently motivated constraints are imposed to decrease the number of free parameters), between a computational or representation-governed process and a process whose operation can be explained physically, and between weak and strong equivalence. Thus it becomes crucial to make the criteria for primitiveness as precise and as principled as possible.

Several criteria have already been discussed in various contexts, particularly in chapter 4. One criterion might be that no cognitive operator is considered primitive unless it can be realized on a computer. Surely this criterion, or condition, is, in principle, necessary. As I have argued, mechanical realizability is the sine qua non of noncircular, mechanistic explanation. Without a demonstration of the constructibility of the behavior in question by a mechanistic process, we have not, as Dennett puts it, discharged all the homunculi and repaid our explanatory debt. Yet the condition clearly is not *sufficient,* since it relies on the realizability of the process on a functional architecture we have no reason to believe is the correct one, a functional architecture responsive primarily to commercial and technological considerations rather than empirical ones.[1]

[1] This is somewhat of an oversimplification. Even in work apparently unmotivated by cognitive science concerns, several empirical constraints are implicit. For example, there are general constraints on the material realizability of certain complex, information-processing functions. While these constraints are not clearly understood, they appear to include the need for autonomous levels of organization. No complex information processor (especially no "universal" processor in the sense discussed in chapter 3) has yet been designed that does not have several distinct levels of organization, one of which is a "symbolic level" (for more on this claim see Newell, 1982), where tokens of physical properties of states function as codes for something else. Another way that empirical constraints are implicit even in work concerned purely with technical computer implementation problems, is that empirical facts about the structure of tasks, as well as human understanding of classes of tasks, is smuggled in whenever machines are designed to do "intelligent" things. I discuss this topic in some detail in Pylyshyn (1978a).

Unless the functional architecture has been independently constrained, the program or model cannot be viewed as strongly equivalent to the cognitive process.

Another criterion implicit in the discussion of informal examples in chapters 1 and 2 is that of capturing generalizations. One might want simply to maintain that the appropriate level of aggregation depends on how useful generalizations turn out to cluster. It might be that for a particular set of generalizations one would adopt a certain level of aggregation, whereas for another set, another level is more appropriate. This much is true: Behavior can and should be described over a variety of levels, each of which captures some generalizations. We cannot be content, however, merely to set our sights on what seems the most convenient level of abstraction available at the moment for dealing with each phenomenon—for several reasons, which arise primarily from our desire to achieve explanatory adequacy.

In cognitive science, as in all theory-motivated sciences, the goal is not merely to describe the regularities of behavior but to relate these regularities to causal mechanisms in an organism. Thus generalizations of such a nature that no one has any idea (however sketchy) how they can possibly be realized by some mechanism are interpreted as *descriptions* of phenomena, not *explanations*. Among the examples of such generalizations are those that seem to require nothing less than a full-blooded, intelligent homunculus inside the model to make it run (the old telephone-switchboard model comes to mind). Also included are generalizations stated over properties of the represented domain, with no indication of the functional (that is, symbol or physical-level) properties of the organism that can give rise to them (several examples are mentioned in chapter 8).

Once one accepts the need for an explanation that is couched at least partly in terms of properties of some mechanisms, the questions arise: Which principles or which aspects of the observed regularity can be attributed to intrinsic functional properties of these mechanisms? Which ones reflect the rules and representations symbolically encoded in the system? I have argued that these different aspects not only involve two fundamentally different forms of explanation, but the assumptions made about the available functional architecture severely constrain the algorithms that can be realized. Furthermore, as I have been at pains to point out, there is no neutral way to describe a cognitive process: every algorithm implicitly presumes some functional architecture. Thus, even though the primary concern is to express generalizations at various levels of the abstraction hierarchy, the need to relate these generalizations to an explanatory causal mechanism makes essential a principled way to choose an empirically constrained *basic level* that is realizable by the available functions of the architecture.

Some Methodological Proposals

How can we distinguish between regularities that are attributable to properties of the functional architecture and those that are attributable to the nature of the cognitive process and its representations? No "simple and sovereign" method is available that will ensure the correct, basic architectural functions have been hypothesized. Not only that, there are no necessary and sufficient conditions for a function qualifying as primitive. Primitiveness is a theory-relative notion. Although we have a sketch of the theory—or, more accurately, some metatheoretical conditions on a theory, together with a set of fundamental, empirical hypotheses—there is still plenty of room for maneuver. Twenty-five years ago many of the techniques (for example, "mental chronometry") for assessing strong equivalence were not available, or, rather, their use in this context was not understood. If, at that time, someone had undertaken to analyze the notion of strong equivalence, much of what we now believe is germane would not have been included. Similarly, it would be foolhardy today to lay down a set of necessary and sufficient conditions to be met by a strongly equivalent model (and, in particular, by the functional architecture). Nonetheless, I shall develop a few provisional ideas because they are already implicit in the work of information-processing psychologists (even if some might not agree with my way of putting it; compare Anderson, 1978; Wasserman and Kong, 1979), whereas others are simply entailed by the theoretical position outlined here.

As an example of the latter idea, recall that strong equivalence requires that a model be expressed at a level of aggregation such that all basic representational states are revealed, since each of these states is essential in the representational story; that is, each cognitive state plays a role in the explanation of behavioral regularities. Thus the transition from one representational state to another must itself involve no representational states; it must be instantiated in the functional architecture. Hence, any evidence of the existence of such intermediate representational states is evidence of the nonprimitiveness of the subprocess in question. Various methods for obtaining such evidence are available. One of the earliest methods for discovering intermediate states in problem-solving involves recording subjects' expressed thoughts while solving the problem (Duncker, 1935). Newell and Simon (1972) developed this technique, which they call "protocol analysis," to a high level of precision (parts of it have been automated in a system called PAS-II; Waterman and Newell, 1971). Although the method can be used only with certain slow, deliberate types of problem-solving tasks (including problems involving visual imagery; see, for example, Baylor, 1972; Farley, 1974; Moran, 1973), it provides evidence of intermediate states that otherwise might not be available for constraining the model. When combined with additional, intermediate observations—for example, protocols of hand movements obtained from video recordings (Young, 1973) and records of eye movements (Just and Carpenter, 1976)—the method can yield extremely useful data.

Possession of intermediate representational states is sufficient reason for the operation not being a primitive one. Protocol analysis has the usual caveat about methodology: Subjects cannot be relied on to provide evidence of only authentic intermediate states; they may provide retrospective rationalizations as well. Furthermore, subjects are highly prone to miss some states; thus the protocol's failure to indicate intermediate states in a certain subprocess is insufficient evidence that such a subprocess is primitive. In the quotation near the beginning of this chapter, Newell and Simon indicate the general strategy for inferring the best set of operations from a summary of the protocol, called the "problem behavior graph". This strategy consists of searching for the smallest set of hypothetical operators to account for the largest number of transitions in the problem-behavior graph.

The existence of intermediate representational states sometimes can be inferred in more indirect ways. A good example occurs in psycholinguistics, in the study of real-time sentence processing. Some indirect evidence exists of certain components of syntactic analysis becoming available during sentence comprehension (Frazier and Fodor, 1978; Marslen-Wilson and Tyler, 1980; Forster, 1979). Any evidence of the availability of intermediate states of a process to any other process (that is, evidence that the workings of the process are "transparent" to another part of the system) can be taken as evidence that such a process is not primitive but has a further cognitive decomposition.

Occasionally the argument that an operation is not primitive must be extremely indirect, because intermediate states are not observable and other relatively direct methods (to be discussed next) are not applicable. In such cases we can resort to the oldest, most venerable method of all: the hypothetico-deductive strategy. If hypothesizing a particular theoretical construct allows us to account for a greater range of phenomena, with fewer assumptions than some other alternative, then we can conclude—always provisionally—that the hypothesis is true. Thus some interesting work has been done that establishes elaborate, detailed models of apparently simple processes such as subtraction without using protocols, reaction time, or many other more common measures of strong equivalence. One example is the BUGGY model described by Brown and Burton (1978), and Brown and Van Lehn (1980), and further developed by Young and O'Shea (1981). This and similar research is based entirely on observation of errors children make in doing arithmetic problems, primarily subtraction. In a task that might otherwise appear quite straightforward, these authors have found it necessary to postulate a large number of extremely detailed subprocesses and rules in order to account in a systematic way for what might otherwise appear to be a random scatter of "silly mistakes" children make. Thus, such an indirect analysis, which involves a much longer deductive chain than other methods, provides evidence of more elementary operations than might otherwise have been hypothesized, as well as providing evidence of the role of various individual rules in explaining regularities in the process.

The preceding brief discussion should serve as a reminder that our ability to discern whether a certain process goes through intermediate representational states is limited only by the theorist's imagination. Whereas there are a number of techniques that, when properly used and independently verified, are sufficient to demonstrate that a process is not primitive but instead involves more microscopic *cognitive* steps, the ability to demonstrate that even smaller steps exist is largely a matter of being clever.

In the remainder of this chapter we consider two empirically based criteria for deciding whether certain aspects of behavioral regularities should be attributed to properties of mechanisms—that is, to the functional architecture—or to the representations and processes operating on them. As I have already suggested, both criteria ideally can tell us when a function requires a more complex cognitive analysis, though they cannot tell us that we have gone far enough, since, as was pointed out, there may be various sources of indirect evidence of the need for further decomposition. The first criterion derived from computational considerations, defines a notion of strong equivalence of processes which I refer to as *complexity equivalence*. This notion of equivalence—while it appears to be similar, though perhaps somewhat weaker, than the intuitive notion of the "same algorithm"—has the advantage of being related to a set of empirical indicators that have been widely investigated in recent cognitive-psychology studies, for example, reaction time and attention-demand measures.

The second criterion is more subtle. It assumes that what I have been calling cognitive phenomena are a "natural kind" explainable entirely in terms of the nature of the representations and the structure of programs running on the cognitive functional architecture, a claim we have already considered informally. If that assumption is found to be true, then the functional architecture itself must not vary in ways that demand a *cognitive* explanation. In other words, the architecture must form a cognitive "fixed point" so that differences in cognitive phenomena can be explained by appeal to arrangements (sequences of expressions and basic operations) among the fixed set of operations and to the basic resources provided by the architecture. Although the architecture might vary as a function of physical or biochemical conditions, it should not vary directly or in logically coherent or rule-governed ways with changes in the content of the organism's goals and beliefs. If the functional architecture were to change in ways requiring a cognitive rule-governed explanation, the architecture could no longer serve as the basis for explaining how changes in rules and representations produce changes in behavior. Consequently, the input-output behavior of the hypothesized, primitive operations of the functional architecture must not depend in certain, specific ways on goals and beliefs, hence, on conditions which, there is independent reason to think, change the organism's goals and beliefs; the behavior must be what I refer to as *cognitively impenetrable.*

Both criteria are developed in the following sections of this chapter. It should be pointed out, however, that there is no way to guarantee in advance that both criteria pick out the *same* level of functional architecture. In fact, it is an interesting empirical question whether they do this. Since we are interested in the strongest possible sense of the psychological reality of programs—hence, of the strong equivalence of processes—we should screen our models by all available criteria, together with such general scientific principles as maximizing the range of generalizations that can be captured by the theory.

Complexity Equivalence

In discussing the dependence of possible algorithms on the functional architecture of the underlying virtual machine, I have presented some examples of algorithms I claim cannot be executed directly on certain types of architectures. For example, I claim that such algorithms as the hash-coding table lookup algorithm, which relies on the primitive capacity of the underlying architecture to retrieve a symbol when given another symbol (called its "name" or "address"), cannot be executed on a primitive machine of the type originally described by Turing (1937), that is, a machine that stores symbols as a linear string on a tape. Similarly, I claim that a register machine that can retrieve symbols by name cannot execute a binary-search algorithm of the kind involved in playing "Twenty Questions" unless it has a way to primitively determine something like an interval measure over the set of names, as would be the case if the names were numerals and the functional architecture contained primitive operations corresponding to the operations of arithmetic.

If we know the architecture—that is, if we are given the set of primitive functions—we can determine whether a particular algorithm can be made to run on it *directly*. For an algorithm to run directly on a certain architecture, the architecture must contain primitive operations whose behavior is formally isomorphic to each elementary step required by the algorithm. In other words, for each elementary operation in the algorithm there must already exist some operation in the functional architecture whose input-output behavior is isomorphic to it. If, to get the algorithm to execute, we must first mimic the input-output behavior of each elementary step in the algorithm, using a combination of different, available operations, we would not say the algorithm is executed *directly* by the available operations, that is, by that virtual machine. We would say that it is the emulated functional architecture rather than the originally available one that directly executes the algorithm in question. The reason I insist on examining the direct execution of algorithms by the relevant functional architecture is that the whole point of specifying a functional architecture is, the architecture is supposed to pick out the correct level of aggregation for the purpose of defining the notion of *same algorithm*, hence, of defining the strong equivalence of programs.

My goal in this section is to work toward a notion of strong equivalence that will serve as a methodological tool for deciding whether a proposed mental function is at the correct level of aggregation, so the program can be viewed as an explanatory model. I shall do this in two stages. In the first stage I suggest a number of properties that are shared by distinct realizations of what intuitively would seem to be instances of the same algorithm. Based on the preceding discussion, it then appears that such properties probably are not preserved if the algorithm's input-output behavior is simulated on the "wrong" functional architecture—even if that is done by emulating each step of the algorithm. On the other hand, these properties should allow for quite different implementations of the same algorithm, so long as the differences are ones that seem inessential to the algorithm's identity. Since, as I have indicated, there is no well-developed theory of algorithmic equivalence in computer science, these ideas must be developed without benefit of an existing body of analysis.

In the second stage I discuss some additional assumptions needed to make these general properties or conditions serve as methodological tools. As an example of the property I have in mind, recall that I have already suggested at least one property of the hash-coding algorithm that must be preserved by any strongly equivalent process, which would not be preserved if the same function were realized on a traditional Turing machine. That property is the relation between (or the form of the function that *characterizes* the relation between) the number of steps it would take to look up a symbol in a table and the total number of symbols stored there. The hash-coding algorithm, implemented on a machine with a primitive facility to retrieve symbols by name (commonly referred to as a random-access or register architecture), can look up symbols with a number of steps that, to a first approximation, is independent of the number of entries in the table. By contrast, if this algorithm were emulated on a Turing machine, the number of steps required would increase as the square of the number of strings stored on the tape (so that the function relating the number of steps and the number of items stored would be a polynomial of order 2). Whereas the exact number of steps required depends on what one decides in advance to count as individual steps, the shape (or order) of the function relating the number of such steps to the number of entries in the table (subject to a qualification concerning what is allowed to count as a single step) does not.

The relation between the number of primitive steps taken and certain properties of the symbolic input (where, in the example above, the entire stored table counts as an input) is generally considered an essential invariant property of what one intuitively regards as different realizations of the same algorithm. For example, clearly we would not count two processes as realizing the same algorithm if one of them computes a function in some fixed time, regardless of its input, whereas the other is combinatorially explosive, so that the time it requires increases without bound as some property of the input (for example, length) is varied. The total time or total number of steps taken is not important in assessing the equivalence of algorithms, since these depend on the particular machine the algorithm is running on. What is important is the nature of the relation between such aspects as time or number of steps taken, and properties of the input, such as its length. Thus, certain differences among programs do not matter for purposes of what I call their *complexity equivalence*. For example, two different programs are viewed as instantiating complexity-equivalent (CE) algorithms if there exists a certain kind of topological relation between them. If every linear series of nonbranching operations in a program can be mapped into a single operation with the same input-output function in the second program, then the programs are complexity equivalent. The second program thus has more powerful primitive operations; but the operations lead to the same complexity profiles, to the same systematic variation in the number of steps or the time taken as the input is varied, *provided only that the number of steps or time taken by each operation is independent of the inputs.* If this provision is the case, then the two programs are indiscernible from the viewpoint of the complexity-profile criterion. Thus, if a program contains the sequence of operations illustrated in figure 1a, it counts as complexity equivalent to a part of another program with only one operation, O_1', which computes the same input-output function.

In this example I am speaking of the length of time or number of steps as a measure of relative execution complexity. A more general notion is one of "amount of resources used," the general idea of computational resources being that of some independently measurable index of the amount of processing being done. In the case of computers, the notion includes the number of basic machine cycles, length of time, amount of memory used, and so on. There are several reasons for considering *amount of resources used* an index of computational complexity. First, in the computer case, it comports well with intuitions concerning *how much computation* a process does. Furthermore, some interesting results have been achieved in computer science, using "time complexity" and "space complexity" (that is, the amount of memory used for intermediate results), though these have been too coarse-grained for distinguishing the kinds of differences we are interested in when we speak of strong equivalence. Second, as we shall see, there is some hope of finding empirical measures of on-line resource use in human cognitive processing, hence, of using these ideas in the empirical enterprise.

Figure 1b, in contrast, illustrates the case of a program *not* considered complexity equivalent to a program with only one operation that computes the same input-output function. In the figure, the diamond-shaped box indicates a branch operation. The loop back to the previous operation, O_1, is taken whenever P(n), which is a predicate of, say, the length of input x, evaluates to "true". The reason this program segment does not count as equivalent to the one-operation subprogram O'_1, *even if O'_1 had the same input-output function as the segment above,* is that the number of O_1 steps it takes in the above program would be a function of n, whereas the corresponding O'_1 has complexity independent of n. Thus any function that can be represented nontrivially with the same flowchart topology as the above segment does not qualify as a computational primitive. Similarly, a subprocess called recursively (for example, on the PUSH arcs of an augmented recursive transition network, or ATN, parser) does not qualify as a primitive operation,

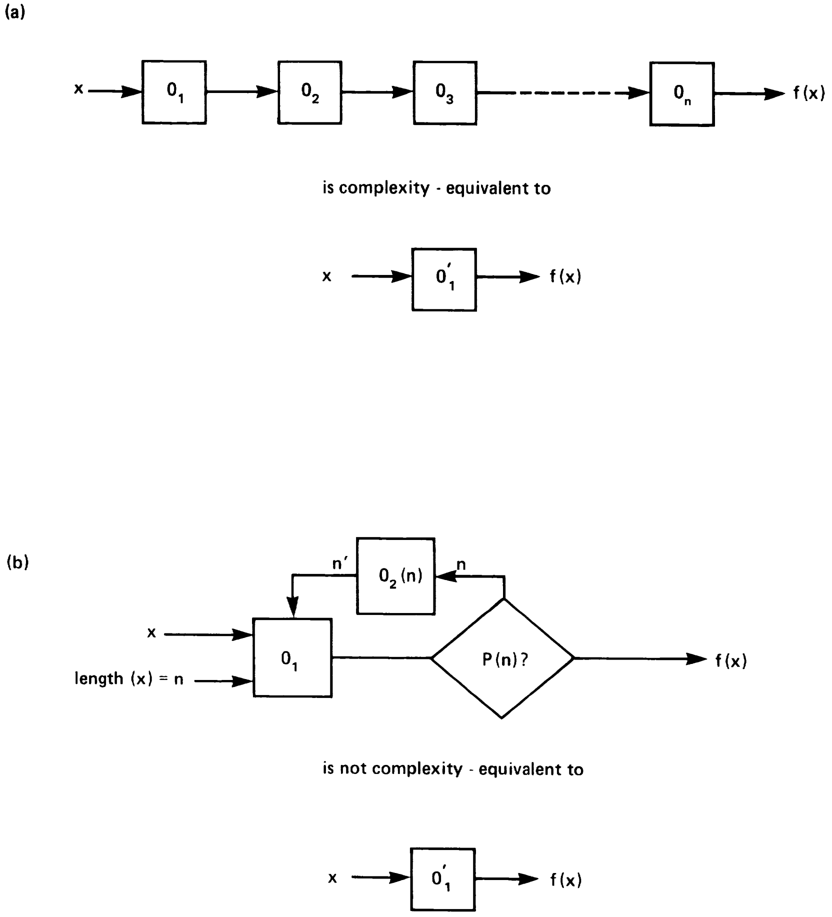

Figure 1. Graphical illustration of equivalent and nonequivalent program segments, using the *complexity-equivalence* criterion.

since its resource use, measured in terms of both time and memory, vary with input. Such formal methods as the graph-minimization methods described by Fosdick and Osterweil (1976) are available for deriving something like a canonical flowchart representation of an algorithm based on topological properties of flowcharts. Indeed, complexity equivalence can be viewed as a special topological relation between two flowcharts, in which the nodes in the flowchart are restricted to those that correspond to fixed-resource processes.

The set of programs that are equivalent, with respect to the way their resource use varies with properties of their input, clearly represents a refinement of the class of programs that compute the same input-output function, and hence, a restriction of the weak-equivalence relation. Although complexity equivalence captures an important aspect of the intuitive notion of *same algorithm*, it alone is not sufficient to define strong equivalence. In other words, it is a necessary but not sufficient condition for strong equivalence. It may not even be strong enough to correspond precisely to the intuitive notion of algorithmic equivalence, for reasons that will soon become clear.

Although the study of computational complexity is an active area in computer science, interest has been confined to study of the way in which the need for such resources as amount of memory and number of steps varies systematically with certain, special properties of the input. For example, the efficiency of sorting algorithms (which in some way order a set of symbolic expressions) is often expressed in terms of the order of the polynomial function that, under worst-case conditions, relates the number of items to be sorted to the maximum number of elementary steps required. There are, however, numerous, different algorithms that can sort n items using a number of elementary steps proportional to $n\log n$. Whereas all such algorithms are equivalent from the point of view of their *time complexity*, they are not complexity equivalent in our sense, because their resource use varies in different ways with change in *other* properties of the input—for example, with the degree of dispersion (along the ordering dimension) of the elements of the set being sorted or even with the addition of certain items to the list being sorted. From the viewpoint of the particular complexity-equivalence relation we are interested in, we can examine the function relating *any* resource-use parameter to *any* property of the input symbols, down to the properties of individual symbols. If two algorithms yield a different resource-use function for any pair of inputs (for example, if they have an interactive effect on a resource-use index), then the two algorithms do not count as equivalent.

Note that at least one pair of inputs is required, since the absolute resource usage means nothing in the case where different machines are involved; we can assess resource use meaningfully only up to a linear transform. Another way of stating this is to recall that collapsing over arbitrarily complex but constant-complexity operations is permitted; thus comparisons can be made only for the form—and, particularly, the polynomial order—of the complexity function.

Because there has been little study of such fine-grained notions of complexity equivalence in computer science, the precise equivalence class thus picked out is not known in detail. We can see already, however, that this criterion is much stronger than the one captured by such notions as the time complexity of algorithms already discussed. For example, both *merge-* and *insertion*-sorting algorithms have time complexity of the order of *n*log*n*; yet the number of steps required for a merge algorithm to sort an item is independent of the exact interval value (on the scale along which the elements are ordered) of the element being sorted, whereas the complexity of insertion algorithms depends on the interval values of the elements. Conversely, our strong complexity-equivalence criterion distinguishes between two cases of a linear-search algorithm that differs only in the order in which the two linear-search algorithms scan a list, whereas many theorists count these as instances of the same algorithm (thus treating the order of the stored list as a nonessential property that can vary with different implementations of what would otherwise count as the same algorithm).

The notion of complexity equivalence is important in cognitive science. Whether it corresponds exactly to the intuitive notion of *same algorithm* is of relatively minor concern; what is important is that the notion provides a way to approach the relation of strong equivalence of programs in terms of functions from properties of inputs to a certain, uniform, aggregate property of the process (which, I argue, we stand a chance of being able to measure empirically). Note that, in this form, the relation of strong equivalence is expressed without the requirement that we know in advance what the functional architecture is. From our point of view, this is exactly the right direction to take, for eventually we want to determine the functional architecture from observed behavior. Thus the next question to ask is: In view of this discussion of complexity equivalence, is there any conceivable, observable behavior that can help fix the functional architecture, and hence, establish strong equivalence?

Strong-Equivalence and Reaction-Time Measures

In this section we need to make a short digression, approaching this question by way of dealing with a frequently heard lament that strong equivalence is, in principle, not possible in psychology, or that it can be achieved only by appealing to the facts of biology or by the gratuitous importation of esthetic or other forms of subjective judgment. The lament might begin with a remark such as: Because weak equivalence is, by definition, equivalence with respect to input-output behavior, and because all we have to go by in psychology (if we exclude physiological psychology) is observed behavior, it must be the case that, leaving aside neurophysiological data, the best we can do is construct a model that accounts for the behavior, hence, that is weakly equivalent to the real mental, or brain, process. After all, how could we ever tell whether our model reflects the "true" process at a finer grain of detail than is exhibited as a difference in overt behavior? Because our notion of strong equivalence explicitly sets out to define a level of correspondence more refined than what appears to be behavioral equivalence, this objection merits discussion, especially in view of the widespread acceptance of the view that there is an inherent indeterminacy of information-processing theories arising from their commitment to behavioral data alone, hence, according to this view, to a weak-equivalence criterion (see, for example, the claims in Anderson, 1978; Townsend, 1974).

The apparent paradox of the behavioral indeterminacy of information-processing models has led to responses from various schools of theoretical psychology. Behaviorists frequently advert to this point in justifying their empty-organism view. If all we have to go by in building *functional* theories of behavior is a record of observations of behavioral movements, together with the environmental contingencies of their occurrence, how can we distinguish among behaviorally equivalent theories except in terms of preconceived, mentalistic ideas? In the case of information-processing theories, Anderson (1978) has developed this conundrum into an "indeterminacy of representation" thesis. The claim of this thesis is that, as long as we attend only to behavior we cannot, *in principle*, tell which of two behaviorally equivalent models (which might differ, for example, in the form of representation they use) is the correct one.

Before looking into these claims in detail, we should note that, as far as this thesis is concerned, there is nothing special about psychology. It can be argued with equal validity that all we have in *any* science is behavior of one kind or another; yet no one has ever suggested that physical theorizing is, in principle, indeterminate, that we cannot, in principle, discover, for instance, the earth's true shape. After all, it *could* be argued that the earth actually is flat and that the physical laws we have discovered are incorrect. In principle, one could have an entirely different set of laws that are compatible with every empirical observation made up to this time, together with the assumption that the earth is flat. Indeed, an unlimited number of models are compatible with any finite set of observations; that is why we have the well-known problem of explaining how induction works. On the face of it, though, the claim is absurd. It is absurd because merely matching a body of observations is not the goal of science; the purpose of theories is to cast light on what seems like chaos by finding the most general and revealing, *lawlike generalizations* that lie behind the observations. To be explanatory, a theory cannot have as many free parameters as there are data points. Put another way, we would not be content with a theory that must be changed each time a new observation is made even if, at any given time, the theory accounts for all available observations (that is why the *pre* appears in the word *prediction*).

The requirement of explanatory adequacy—that theories should capture the most general and revealing, lawlike (counterfactual supporting) generalizations—is itself enough to ensure that the criterion of matching a set of behaviors does not allow proliferation of weakly equivalent theories, among which science should be indifferent. There is a more specific way to approach the indeterminacy thesis in information-processing theories. I have argued (Pylyshyn, 1979c) that the answer to the sort of counsel of indifference one finds in Anderson (1978) and elsewhere is this: While, in a sense, all we have is behavior, not all behavior is of the same kind, from the point of view of theory construction. By imposing an independently motivated partition on a set of behaviors, and by interpreting the partitioned behaviors in different ways, we can do much better than weak equivalence.

As an example, consider modern linguistics. It is true that all we have are the linguistic utterances of speakers. It does not follow, however, that the best we can do is analyze a corpus of utterances and, perhaps, develop a taxonomy and tables of co-occurrence relations. Even within the narrow goal of describing the syntactic structure of a language (as opposed to, say, explaining when and why certain utterances occur in relation to other factors), we can do much better—as Chomsky showed in the early 1950s (see, for example, an early monograph reprinted as Chomsky, 1975). To do that, we must distinguish two classes of utterances (ignoring, for now, the question of developing appropriate idealizations rather than taking actual observed behavior). The utterances in one class are those sentences whose structure we wish to describe. Together with their structural descriptions, they constitute the output of our grammatical theory.

The utterances in the other class, by contrast, are *judgments*; they are not part of the output of any theory. Rather, they are interpreted as judgments reflecting the speaker's perception, or tacit knowledge, of the structure of sentences in the first class. In other words, it is the semantic content of the latter group of sentences, or what they *assert*, that matters. In contrast with the primary data sentences, these sentences are taken as asserting something the theorist chooses to believe or not to believe, depending on the theorist's methodological bent. The sentences provide secondary data of a metalinguistic sort (in the sense of their not being among the outputs of the model, not in the sense of their being less important), from which the theorist typically infers the structural properties of the primary data sentences.

A parallel distinction is made in experimental cognitive psychology. Here, the investigator collects primary observations in a certain domain—say, those concerning the behavior of a person solving a problem. These are observations which a constructive theory of the domain might be expected to generate as output. In addition, the investigator typically collects secondary observations, or what might, without overly distorting the terminology, be called metabehavioral observations, from which certain properties of the intervening process are inferred. Sometimes such observations are taken to be truth-valuable assertions, or judgments about the primary behavior, similar to the linguistics case. This is the case, for example, when subjects provide "thinking-out-loud protocols" of the kind studied extensively by Newell and Simon (1972). At other times, such secondary observations are interpreted as *indices* of, for example, processing complexity or what I have referred to as "resource use." In this case, it is not expected that a theory or a model actually generate such behavior as part of its output. Rather, the model should generate the primary (output) behavior in a manner that reflects certain real-time processing properties assumed to be indexed by observations in the secondary class.

Consider the following example in which the developing methodology of cognitive science has led to a gradual shift in the way an important aspect of observed behavior is interpreted. The example involves what may be the most widely used dependent measure in cognitive psychology: reaction time. On occasion this measure has been interpreted as merely another response to be accounted for by a cognitive model just as the model accounts for such response properties as the record of buttons pressed. Since Donders' pioneering work (in the 1860s; reprinted as Donders, 1969), the measure has been widely interpreted as a more or less direct measure of the duration of mental processes (Wasserman and Kong, 1979). In commenting on the Wasserman and Kong paper, I argue (Pylyshyn 1979a) that neither interpretation essentially is correct, that, in general, reaction time can be viewed as neither the computed output of a cognitive process itself nor a measure of the duration of a mental-event type.

If reaction time were viewed as simply another response, it would be sufficient for our computational model to calculate a predicted value for this reaction time, given the appropriate input. That would not be sufficient, however, if the computation is viewed as modeling the cognitive *process*. Contemporary cognitive theorists would not view a system that generated pairs of outputs, interpreted as the response and the time taken, as an adequate model of the underlying process, no matter how well these outputs fit the observed data. Instead, they wish to interpret the model as computing the output *in the same way* as the subject that is, by using the same algorithm.

It has become customary in cognitive science to view reaction time in the same way that measures such as galvanic skin response, or plethysmograph records, and distractibility (as measured, say, by Brown, 1962) are viewed, as an *index* or an observable correlate of some aggregate property of the process. Frequently, reaction time is viewed as an index of what I call "computational complexity", usually considered as corresponding to such properties of the model as the number of operations performed. A process that merely computes time as a parameter value does not account for reaction time viewed in this particular way, since the parameter does not express the process's computational complexity.

This view of the role of reaction-time measures takes it for granted that such measures are not interpreted as a direct observation of the duration of a certain mental type. True, reaction time may be a measure of the duration of a particular episode of some mental event; to interpret it as the duration of a mental-event type, however, is to assume that, in other circumstances (on another occasion, in another subject, in another part of the brain), the same event type would always require the same length of time. If we were considering a type of physical event, we would know that if something which counts as the identical physical event occurred on a different occasion, it would indeed require the same length of time—because taking a particular length of time is part of being a particular physical event; that is, it is an intrinsic property of a physical event (something that took a different length of time is, by definition, a different physical event). It is helpful to remind ourselves that a mental process does not possess the intrinsic physical property of duration any more than it possesses the property of location, size, mass, electrical resistance, or concentration of sodium ions. Since this statement often causes eyebrows to be raised, it might be useful to pause and consider why that must be true, as well as why we find it so natural to ascribe duration to mental events even when we are not similarly tempted to ascribe volume or some other physical property to them.

The distinction that must be kept in mind is the one I devote considerable attention to in chapter 1, namely, that between types and tokens of events. Every mental-event type (for example, thinking that 2 + 3 = 5) is realized by a corresponding, token brain event, or so goes the "supervenient," or conservative, version of the materialist story. Naturally, this token brain event has various physical properties, including mass, volume, temperature, and duration. As I have argued though, there is no a priori reason for assuming that the entire class of brain events that could ever conceivably constitute the same mental event (identified by whatever criteria for sameness of mental-event type we are disposed to adopt—for example, functional or semantic) has any physical property in common. Clearly, in a machine, all tokens of the same computational event type (such as adding two numbers) need have no common physical property—at least, not in the ordinary sense of physical property, such as those found in physics texts. Thus there is no reason to assume, a priori, that all instances of brain events corresponding to a certain cognitive event have certain physical properties in common. We would certainly not expect all brain events corresponding to thinking 2 + 3 = 5 to have a common temperature or volume.

It does not seem at all implausible, however, that every occurrence of a brain event corresponding to certain elementary cognitive operations might turn out to require the same length of time, to a first approximation. If that contingent empirical fact turns out to be true, then we will have discovered an important feature of the cognitive system, in which case we could make important inferences from measurements of the duration of token brain events. Strictly speaking, we would still not be measuring the duration of the mental event—that is, of an independently defined class of biological events—only a property of a brain event we have discovered to be invariant over the class. In other words, we would be measuring an empirically valid physical correlate of the mental event.

Although the use of reaction time and other, similar measures is crucial in the empirical assessment of complexity equivalence, it is equally important to understand the way in which these measures serve this methodological goal. Let us suppose we have an observation (for example, duration) of a property of the algorithm's physical instantiation. I have already argued that this observation cannot be interpreted as the direct measurement of a property of some mental or computational event. Nonetheless, the question remains: Under what conditions can such observations provide evidence in favor of one proposed algorithm over another?

We have already considered several cases in which it is possible to decide which of two radically different algorithms is executed by examining the relative number of primitive steps (or single operations in the functional architecture) they took when given different inputs. Now, if we have some reason to believe that the amount of real time required is proportional to, or at least is a monotonically increasing function of, the number of such primitive steps of the algorithm, then measures of relative time taken might provide the evidence needed for deciding between the putative algorithms. In this case, however, we need an independent reason for believing that reaction time is a valid index of an algorithmic property—namely, the number of primitive steps in the functional architecture. Such independent reasons as these frequently are available, for example, when regularities inferred on the basis of the assumption that reaction time is a reliable indicator of processing complexity are corroborated by other methods. When such patterns of consistency keep showing up under converging methodologies, we have a prima facie reason for expecting the methods to be valid, all else being equal (for examples of much convergence, see Posner, 1978).

Nonetheless, it should be kept in mind that inferences drawn about the nature of the algorithm from reaction-time data (or any other *physical* measurement) always depend on the validity of an ancillary hypothesis. Such a hypothesis could, in principle, be false. There are many situations in which measurements of properties of the underlying physical events tell us little about an algorithm. Instead, they might tell us (1) about the way a process is physically (that is, neurophysiologically) instantiated on some occasion, in some individual, or (2) about subjects' tacit knowledge, or about the nature of the task itself. Here, I digress briefly and consider these two cases, since they reveal an important, general point concerning the relationship among cognitive phenomena, the task being carried out, the method a person uses, the fixed, functional properties (or functional architecture) of the cognitive system, and the biological or physical properties of a particular, token instantiation of the solution process.

The possibility that such measurements as reaction time tell us little about the underlying biological mechanism is of special concern in "higher-level cognitive functions," where processing is not tied as closely to anatomical structures as it is, say, in certain areas of perception or motor coordination. In such cases, we are not as likely to find physical properties that are invariant over all instances, or token occasions, of cognitive events. Whereas, often there is a correlation between the duration of an observable physical event and such purely algorithmic properties as the number of steps taken, the particular steps taken, whether parts of the algorithm are performed serially or in parallel, and so on, that is not always the case. There are certainly cases in which time differences arise from properties of the physical realization that are unique to the particular occasion or instantiation (for example, a particular individual) and therefore are, in general, irrelevant to the algorithmic, or process, explanation. Such duration data may not validly discriminate among putative algorithms.

Using a computer as an example, we can see that some time differences might arise because a signal has farther to travel on a particular (or token) occasion because of the way the machine is wired or the way the algorithm is implemented in it, or that some differences might arise from variable-delay effects unique to specific occasions. An example of the latter case is the delays caused by the distance a movable arm must travel in making a disk access in some implementation and on certain occasions, unrelated to the content of the memory or the algorithm used. Other delays may depend on physical properties of the noncomputational environment, as would be the case if real-time interrupts were to occur. None of these observations bears on the nature of the algorithm, since they could differ considerably on another occasion or for a different realization of the same algorithm. Consequently, in this case, measuring the times involved does not help us distinguish different candidate algorithms. That is why, in the computer case, time measurement alone cannot be taken as measurement of a property of the algorithmic process. For precisely the same reasons, time measurement cannot be taken literally as a measurement of mental duration—only as indirect (and, possibly, false) indicators of such things as processing complexity, to be used judiciously with other indirect sources of evidence in inferring underlying mental processes.

The other case in which observations may tell us little about the cognitive process itself arises when the primary determinant of the behavior in question is what Newell and Simon (1972) call the "task demands". Consider the various observations associated with certain operations on mental images. Many such investigations (for example, Shepard and Feng, 1972; Kosslyn, 1980) have measured the time it takes to imagine a certain mental action, for instance, mentally folding a piece of paper according to certain instructions, or scanning your attention between two points on a mental image. What these experiments consistently fail to do is distinguish between two different tasks demanded by the same general instructions to a subject. The first task is simply to use a certain form of representation to solve the problem; the second is to imagine *actually seeing* certain problem-solving events take place. In the latter case, we would, of course, expect the subject to attempt to duplicate—while imaging—various incidental properties of the events they believe would occur if they were to watch the corresponding, real events unfold, for example, the order and duration of particular component events. If the second task is the one subjects are performing, the fact that certain temporal patterns are obtained tells us nothing about the process used, only that the subjects can generate actual time intervals corresponding to those they believe would have occurred if the event had actually taken place. I have argued that this, in fact, is the task being performed by subjects in many such experiments. These and similar examples are discussed in chapter 8.

One more point needs to be made about reaction-time measures before we turn to the second major methodological principle. Whatever the human, functional architecture turns out to be, clearly it is capable of carrying out Turing-machine computations within the limits of its resources (limits which, in fact, humans can increase artificially by getting a piece of paper or writing a program for another machine). Like the universal Turing machine, not only can a person carry out any computable function, that person can *emulate* any conceivable functional architecture, perhaps to the extent of generating the appropriate time intervals or latencies (all the person would have to do is simulate the other functional architecture and, in doing so, arrange to take a constant length of time for each primitive operation of the functional architecture being emulated). If that is the case, though, how can we know that when we do experiments using reaction time as our dependent measure, we are not, in fact, getting measures of an emulated functional architecture? If we *were* getting such measures, the point of the reaction-time measurements would be destroyed, since we would be learning nothing about the person's native, or biologically endowed, capacities.

The reply to this concern is the same as that to most other such concerns: All methodologies are based on assumptions. The proof of the correctness of these assumptions is the continued success of the methods in revealing interesting, general properties of the system under study. Many assumptions underly the use of reaction-time measures in cognitive psychology. One is the assumption that in the range of times where many of the reaction-time regularities appear (on the order of 20 to 100 milliseconds per operation), only the operation of the primitive level of the functional architecture is being tapped. Even the highly complex operations involved in grammatical analyses of sentences appear to take place in time spans less than the duration of a syllable (a few hundred milliseconds). Much longer mental operation times have been observed under certain conditions (for example, the time required to generate a mental image appears to be more than 1 second). In these cases one must be wary of the possibility that more complex cognitive processes may be occurring. Even here, though, we *may* have reason to believe that primitive operations are being observed, only that they are slower operations. In other cases one may have good reason to think an operation one is measuring is not primitive but that the conditions have been set up so the variation in reaction times occurs primarily from the iteration of primitive operations. This assumption is made in explaining the results of the Sternberg short-term memory-scanning experiment (Sternberg, 1966).

On the other hand, an example of a widely used measure which theorists have begun to suspect is actually tapping a more complex decision process is the "lexical decision task" used to study access to the mental lexicon in such performances as reading (West and Stanovich, 1982). The lexical decision task requires subjects merely to state whether a designated string of characters is a word. Because of its sensitivity to various interesting manipulations (for example, it is shortened significantly by priming the subject with a semantically related word), this task has been studied extensively. Although reading time for words is only a few hundred milliseconds, lexical decision time is considerably longer. Thus the task may involve a complex decision (such as double-checking). In that case, it may be the decision component that accounts for the way reaction time varies with such things as the semantic similarity of the target item. This is a case where length of reaction time alerts investigators to the possibility that they are studying a complex composite process.

Then there is the converse concern. The question sometimes arises whether it is possible, with practice, for an operation that was once slow and complex to become a fast, primitive operation. It is certainly a well-known phenomenon that with a great deal of practice, a process that was once a deliberate and slow sequence of operations becomes automatic and very fast. Nothing in what I have said so far prevents the creation of new, primitive operations (analogous to running part of a program through an "optimizing compiler," creating an efficient subroutine). As we see in the next section, primitive functions of the functional architecture *can* change. They cannot change in certain ways, however; to do so would violate basic constraints on the functional architecture without which the notion of functional architecture, as a theoretical construct, is powerless to explain the cause of certain generalization in behavior. Whereas the functional architecture might change in systematic ways in response to biological, physical, or chemical causes, and perhaps to repetition, it must not change the way behavior changes when people find out new things and acquire new beliefs and goals. Those are precisely the regularities the cognitive process, realized by the symbol-processing facilities of the functional architecture, is meant to explain. Hence, those changes must not be internal to the functional architecture. All kinds of other causes of change can occur, however, and can alter the functions instantiated in the functional architecture. Presumably, that is what happens when infants mature and people become ill or depressed or, perhaps, when performance changes in response to prolonged, repetitive practice (see chapter 9).

To conclude this section, it should be noted that despite the caveats concerning the fallibility of using such indices as reaction time, these measures have been instrumental in allowing psychologists to begin taking seriously the notion of strong equivalence. To the extent that the methodology for determining strong equivalence stands up to repeated scrutiny and continues to yield new insights concerning the structure of cognitive processes, the skeptics' claim of indeterminacy will be just as empty in psychology as it would be in any other science.

Cognitive Penetrability

The second criterion for determining the appropriate level of aggregation is in many ways the more fundamental one. It relies on the distinction, already discussed, between phenomena that can be explained functionally and those that must be explained by appeal to semantically interpreted representations. The second criterion I will propose consists of little more than an application of this distinction to individual subfunctions of the model. What makes the difference between phenomena explainable functionally and those explainable in terms of rules and representations is exactly what makes the difference between subfunctions that must be further analyzed in terms of a cognitive process and those whose operation can be attributed to the functional architecture of the underlying virtual machine. Thus, *whatever* the reasons for deciding to give a cognitive, or representational, explanation for some phenomena, these reasons should apply, mutatis mutandis, in deciding whether also to give any hypothesized subfunction in the analysis a similar cognitive explanation, as opposed to assuming that it is an instantiated function of the architecture.

The need to distinguish between regularities that can be explained by appeal to intrinsic biological and physical properties of a system and those requiring appeal to what the system represents (its beliefs and goals) is a central concern of this book. Paralleling this distinction is the closely related distinction between processes governed by semantic principles (which I call "cognitive processes") and those realized in what I call the "functional architecture" of the system, the latter being a term borrowed from computer science, where it is used to refer to the basic set of resources of a computer system (either hardware or software) available for creating programs. According to the position I have taken, processes carried out in the functional architecture are processes whose behavior requires no explanation in terms of semantic regularities—that is, in terms of rules and representations. That position, examined indirectly in this section, provides the basis for a criterion I call "cognitive penetrability."

Because of their centrality, these distinctions, raised and discussed in a number of places, are examined at greater length in chapter 7 in connection with the distinction between analog and digital processes. In this section, I take the importance of these distinctions for granted, worrying instead about how we can provide principled constraints on what counts as functional architecture. Such constraints are necessary in order to prevent the trivialization of explanations one gets if the presumed basic operations of the functional architecture are allowed to range over, for instance, "decide whether P is true," "find the winning move," or "solve the problem" while at the same time preventing the architecture from being tied to the psychologically unrealistic but widely used operations of a von Neumann computer architecture. The criterion proposed in this section is a direct consequence of a view that might be called *the basic assumption of cognitive science,* the assumption, already discussed, that there are at least three distinct, independent levels at which we can find explanatory principles in cognitive psychology. Classically, they represent the principal approaches to psychological explanation—biological, functional, and intentional. Newell (1982), who recognizes many more levels than are relevant to the present discussion, refers to these particular levels as the device level, the symbol level, and the knowledge level. These are probably the most revealing names, though *symbol level* implies something more specific than *functional level,* and the term *knowledge* raises philosophical eyebrows (strictly speaking, it should be called *belief*). By implication, the term *knowledge* also suggests that one is ignoring other representational states, such as goals, as well as such propositional attitudes as fears and hopes. By the same token, *intentional,* as a term of art of phenomenology, carries too many superfluous connotations. That is why, in chapter 2, I use the term *representational* or *semantic* for this level and will continue to do so here.

If one accepts this trilevel characterization of cognition, then attempts to explain certain empirical regularities should proceed as follows. First priority goes to explaining the regularity in question in physical or biological terms, that is, at the physical level. If, under a description of behavior that captures the regularity in question, that regularity can be subsumed under biological or physical principles, we need go no further; we do not posit special principles when the universal principles of physics will do. This application of Occam's razor prevents us from ascribing beliefs and goals to streams, rocks, and thermostats. Of course, if the system is a computer, there will be some description of its input-output behavior (namely, the description under which the system is seen as executing a program) that will not be explainable by appeal to physical laws. The explanation of the machine's production of a certain output symbol when the machine is given a certain input symbol is not explainable at the physical level, for numerous reasons already discussed; for example, because of the failure of type-type equivalence of the physical and computational vocabularies, a different physical explanation holds for each distinct way of "inputting" a particular symbol, of "outputting" another symbol, and of physically instantiating the same program, despite the existence of a single explanation at the symbol level. In other words, a single program captures all the regularities that otherwise would have to be covered by an arbitrarily large disjunction of physical explanations. Hence, in this case, a symbol level explanation would have to be given.

Similarly—again, if the assumption about levels is correct—if regularities remain that are not explainable (under the description that best captures the generalizations) at either the physical or the symbol levels, appeal must be made to the semantic level. But what sort of regularities can these be? The answer has already been given: precisely the regularities that tie goals, beliefs, and actions together in a rational manner (*S* wants *G* and believes that *G* cannot be attained without doing *A*; therefore, everything else being equal, *S* will tend to do *A*). Just as physical-level principles provide the causal means whereby symbol level principles (embodied in the rules or the program) can be made to work, so symbol level principles provide the functional mechanisms by which representations are encoded and semantic level principles realized. The three levels are tied together in an instantiation hierarchy, with each level instantiating the one above.

It is clear that certain systems (humans, chimps, robots) exhibit regularities at all three levels. Now, suppose we have a hypothetical model of the cognitive processes of such a system. At the symbol level the model contains component subprocesses with subsubprocesses, and so on, which, in turn, are composed of basic operations, the primitive operations of the functional architecture (recall that that is where the symbol explanation stops). The question arises: How can we tell whether the hypothesized primitive processes are actually primitive? In other words, how can we tell whether they are instantiated in the functional architecture or are themselves the result of representation-governed processes?

Being instantiated in the functional architecture merely means being explainable without appeal to principles and properties at the symbolic or the semantic level. An operation whose behavior (under the description required of it by the model) *must* be given an explanation at the symbol level does not qualify, nor would an operation whose behavior (again, under the relevant description) must be explained at the semantic level. The first exclusion is discussed in the preceding section, where a general characterization of a computational primitive is given in terms of resource use. What, then, is a general characterization of a primitive operation from the perspective of the semantic level? The answer is obvious: The behavior of the putative, primitive operation must itself not require a semantic level explanation. In other words, in explaining the behavior of the hypothesized primitive, there must be no need to appeal to goals, beliefs, inferences, and other rational principles, as presumably there is in the case of explaining the behavior of the original, complete system. Thus (to use the terminology introduced by Dennett, 1971), not only must the reduced homunculi be increasingly "stupid," but at the most primitive level they must no longer be "intentional systems." The most primitive level must not behave in a manner that requires a cognitive (rules and representations) explanation.

In discussing the reason for appealing to the semantic level, I made the following observation. An outstanding characteristic exhibited by systems governed by rules and representations or by semantic principles is an extreme degree of holism and plasticity. If what goes on in such systems is, among other things, a process involving *inference*, then such holism and plasticity is just what one would expect. In general, whether a particular "conclusion" is permitted can depend on *any* premise. Further, changing a premise can have arbitrarily far-reaching effects. If behavior is determined by beliefs inferred from other beliefs and goals, then changing a belief (equivalent to changing a premise in an argument) can change the behavior in radical, albeit coherent and rationally explicable, ways. This plasticity of behavior, wherein regularities can be changed in rationally explicable ways by changing beliefs and goals, is seen as a prime indicant (though not the only one) of representation-governed processes. Consequently, it becomes a prime counterindicant of a function or component that is part of the functional architecture. The rationally explicable alterability of a component's behavior in response to changes in goals and beliefs is what I refer to as *cognitive penetrability.*

The essence of the penetrability condition is this. Suppose subjects exhibit behavior characterized in part by some function f_1 (say, a relation between reaction time and distance, angle, or perceived size of an imagined object) when they believe one thing; and some different function f_2 when they believe another thing. Suppose, further, that the particular f the subjects exhibit bears some logical or rational relation to the content of their belief. For example, they might believe that what they are imagining is very heavy, that it cannot accelerate rapidly under some particular applied force. The observed *f* might then reflect slow movement of that object on their image. Such a logically coherent relation between the form of *f* and their belief (which I refer to as the "cognitive penetrability of *f*") must somehow be explained. My contention is that, to account for the penetrability of the process, the explanation of *f* itself must contain processes that are rule-governed or computational—for example, processes of logical inference—and which make reference to semantically interpreted entities, or beliefs. The explanation cannot state merely that some causal (biological) laws exist that result in the observed function *f*—for the same reason that an explanation of this kind is not satisfactory in the examples examined in chapters 1 and 2 (for example, dialing 911 when an event is perceived as an emergency, or leaving a building when one has interpreted an event as indicating the building is on fire); the regularity in question depends on the semantic content (in this case, of beliefs) and the logical relations that hold among the contents. Although, in each case, some physical process causes the behavior, the explanation must appeal to a generalization that captures the entire class of such physical processes. As Fodor (1978a) puts it, there may be token reduction but no type reduction of such explanatory principles to physical principles.

A process that must be explained in terms of the semantic content of beliefs typically contains at least some inferential processes, or some processes that preserve semantic interpretation in some form (for example, such quasi-logical principles as heuristic rules). Thus the term *cognitive penetrability* refers not merely to any influence of semantic or cognitive factors on behavior but to a specific, semantically explicable (that is, rational or logically coherent) effect. The examples I shall describe in connection with a discussion of imagery are clear cases of this sort of influence (for more on this particular point, see chapter 8). It should be noted as well that being cognitively penetrable does not prevent a process from having *some* impenetrable components that are part of the functional architecture. Indeed, in my view, this *must* be the case, since it is the functional architecture that makes the thing run; it simply says that the behavior (or the particular phenomenon in question) should not be explained solely by appeal to the functional architecture or to analogue media (see chapter 7), with no reference to tacit knowledge, inference, or computational processes.

A good example occurs in perception. Without doubt, the perceptual process is cognitively penetrable in the sense required by our criterion. What one sees—or, more accurately, what one sees something to be—depends on one's beliefs in a rationally explicable way. In particular, it depends in a quite rational way on what one knows about the object one is viewing and on what one expects. This, the point of numerous experiments by the "new look" school of perception (see, for example, Bruner, 1957), clearly shows that, by and large, perception involves semantic-level principles—especially those of *inference*. Nonetheless, as Fodor and I have argued (Fodor and Pylyshyn, 1981), a clearly noninferential component is required as well, one that is part of the functional architecture. This component, called a transducer, may well be extremely complex by biological criteria, yet it counts as a cognitive primitive (I shall have more to say about transducers later in this chapter and in chapter 6). Furthermore, the transducer component is cognitively impenetrable (though, in the case of vision, its sensitivity can be dampened, as happens in pupillary dilation; or it can be redirected, as happens when the direction of gaze is changed—neither of which count as instances of cognitive penetration).

Although cognitive penetrability is an extremely simple idea, its use as a methodological criterion can on occasion be less than straightforward. This should not be surprising, since the practical problem of applying a principled distinction is always difficult, even when the distinction itself is simple and clear—for example, the distinction between a causal connection and a mere correlation. Following are three major reasons why the *practical* application of the criterion requires care and ingenuity.

1. The first problem in applying the cognitive-penetrability criterion in practice is that, while we want to draw conclusions about a hypothetical *subprocess*, all we have direct access to is the behavior of the entire, intact organism, which, we already know, is cognitively penetrable. The problem can occur even in the simplest demonstration of cognitive penetrability, for example, the f_1 and f_2 examples already mentioned. The question can always be raised: How do we know that the same relevant component was used in the f_1 and f_2 conditions? Even if it was used, how can we be sure that the difference in functions is due to the direct influence of instructions (or whatever is the cognitive manipulation) on the component of interest, rather than some other component? Perhaps the component in question (the putative, primitive operation of the functional architecture) actually was cognitively impenetrable, and the effect on the observed f came from some other component. That the difference in an observed function may have come from a component other than the one under scrutiny remains a problem. Note, however, that it is by no means unique to the application of the cognitive-penetrability criterion. In fact, the problem occurs every time one wishes to test an information-processing model. For this reason, it has led to the development of various sophisticated strategies for analyzing information-processing components, or "stage analysis" (see, for example, Sternberg, 1969; Massaro, 1975; Posner, 1978). Since there

is nothing special about this aspect of the problem, relative to any model-testing task, I have little to say about it except to recognize that it is something that must be taken into account.

2. The second problem is that a principle such as that of rationality is not directly observable in behavior. The principle represents an idealization needed to distinguish importantly different classes of principles. Particular instances of behavior may manifest this principle, but they will at the same time also manifest the effects of a variety of other factors. Just as symbol level principles (say, those embodied in a computer program) do not account for *all* aspects of a computer's input-output behavior (even under the relevant description of this behavior), because of the intrusion of physical level properties (for example, components may fail, physical memory or disk resources may be exceeded, real-time interrupts may occur, there may be spikes on the power line, and so on), so also does the semantic level not account for all aspects of behavior that fall under its principles. In particular, not all inferences permissible under the rules *can* be made, for the following reasons: not all beliefs are accessible at all times (*that* depends on the access key used), beliefs can become entrenched and may not change even when it would be rational for them to do so, people do not bother to consider all relevant factors, they get misled by surface cues, they cannot keep track of complex arguments or derivations, and so on. These are deviations from logical omniscience or rationality that reflect the intrusion of such symbol level principles as control structure, limited resources, time and space constraints on various information-handling mechanisms, as well as, possibly, physical-level principles. Consequently, deciding whether a certain regularity is an instance of the application of a rational principle or rule is not always straightforward—though sometimes it is, as we see in the examples taken from studies of mental imagery, in chapter 8.

3. The third reason for difficulty raises certain questions that need to be sketched in some detail, because the difficulty has been the source of many objections to the cognitive-penetrability criterion (for example, those in the responses to Pylyshyn, 1980a; but see Pylyshyn, 1980b). The difficulty is illustrated by the following observation. In addition to intrusions that prevent the observation of "pure" cases of semantic level principles, there are certain systematic relations that hold between principles at different levels. Far from being intrusions, they are the necessary links between semantic principles (and representations) and the physical world. When, on the basis of my goals, beliefs, and utilities, I decide on a certain action, certain behaviors ensue whose subsequent unfolding may be explainable only under a physical description. For example, if I am playing baseball and I infer, on the basis of my knowl-

edge of the rules of the game and my interpretation of the behavior of an opposing player, that the player is about to steal third base, I may decide to throw the ball toward the third baseman. Much of what happens after I initiate this action cannot be explained in terms of my beliefs and goals or those of anyone else; most of the relevant regularities first come under the generalizations of physiology, then under the laws of physics. Similarly, when I perceive the world, the relevant generalizations for explaining what I see begin with the laws of optics, then the principles of biophysics and biochemistry. Only later do they involve semantic principles (that is, at the point where perception involves inferences; see Fodor and Pylyshyn, 1981).

Now, the existence of obvious, causal connections between semantic principles (or, more particularly, semantically interpreted representations) and physical properties suggests the possibility of "mixed vocabulary" principles involving both semantic and physical terms. If true in the general case, this might undermine the thesis of autonomy of the semantic level, hence, the basis of the cognitive-penetrability criterion. Indeed, the criterion might have no point, since the distinction between functional architecture and symbolic processes would disappear. Clearly, however, this concern is premature. The existence of some causal interface between semantically and physically explainable regularities does not undermine the distinctions I have been pressing as long as there are principled constraints on this interface. But there must be such constraints; thoughts cannot have just any kind of direct effect on physical properties. (Of course, they can have almost unlimited indirect effects, since they can result in someone deciding to go and do something about changing some property in the world, say, by setting up the right physical conditions to induce the desired change.) If thoughts did have such direct effects, life would be a lot easier than it is. On the other hand, the notion that there must be *some* such bridging principles makes the application of the criterion of cognitive penetrability less straightforward, since we can expect that some effects of beliefs on behavior will always be mediated by such mixed-vocabulary principles, especially since both *internal* and external properties may ultimately be affected by goals and beliefs.

Several examples have been cited by way of suggesting that semantic principles and physical or biological principles can interact freely. What I shall do in the following is describe a few of these examples and suggest that, as far as the purpose of the cognitive-penetrability criterion is concerned, they represent effects of the wrong kind to count as cases of cognitive penetration, either because they are not instances of content-dependent regularities or because they are "indirect" effects (such as those mentioned in the preceding paragraph). First, I simply argue that the cognitive-penetrability criterion remains useful because one can see intuitively that a different effect is responsible for these examples and that clear cases of cognitive penetration (involving rational or inferential processes, as in the case of perception) are easily discerned. Thus I conclude that, in practice, this kind of example presents no problem. In a subsequent section I take up the more fundamental question of whether such examples threaten the autonomy principle behind the distinction.

What theorists have worried about in connection with the cognitive-penetrability criterion apparently are counterexamples of the following classes.

1. The first class involves clearly noncognitive or nonsemantic processes which, nevertheless, appear to be systematically altered by the contents of beliefs. For example, beliefs about an imminent threat cause the heart rate to increase and an entire set of physiological reflexes to occur. The digestive process can be altered, causing one to become distracted. Surely, however, these processes do not follow semantic-level principles; they are not cognitive processes.

2. The second class of counterexample is closely related to the first, except that in this case the effects are under voluntary control. The class includes not only such voluntary actions as extending one's arm but control over normally involuntary processes such as heart rate and alpha rhythm, something which, to a limited degree, can be achieved through biofeedback and other training methods. Once again, it appears that we can cognitively influence noncognitive processes.

3. The third class involves what I call the "indirect" influence of goals and beliefs on both cognitive and noncognitive processes. As Georges Rey (1980) points out, *any* process humans have learned to tamper with (from the reproductive cycle of chickens to other people's feelings and, indeed, to more drastic cognitive alterations produced through psychosurgery) constitute cases in which beliefs and goals influence something (sometimes cognitive, sometimes not), although, as Rey concedes, it is clear that the "paths of influence are [not] of the right sort."

These examples were presented by various people to cast doubt both on the usefulness of the criterion of cognitive penetrability as a methodological principle and on the assumption of the autonomy of the semantic level and, hence, on the distinction between functional architecture and symbolic process. Worries about the methodological usefulness of the criterion (on the grounds that its use depends on an ability to discern "paths of influence of the appropriate sort") are the less serious of the two. General methodological criteria *always* require judgment in their application; indeed, no psychological experiment has ever been performed whose interpretation did not depend heavily on making *the very judgment* questioned in these examples: whether certain effects are caused by "influences of the appropriate sort." That is because every experiment involves instructing subjects to, for example, attend to certain stimuli and ignore others. The assumption is always made that the semantic content of the instructions is communicated to the subjects and that that is what determines their understanding of the task. In other words, it is assumed that at least instructions have their effect through "paths of influence of the appropriate sort," namely, those governed by semantic-level principles rather than, say, by causally affecting the subjects according to physical principles that take the instructions under a physical description, for example, according to their intensity or duration.

One must remember that cases in which the criterion is used (for instance, the perception examples and the examples concerning models of mental imagery, discussed in chapter 8) involve applications as clear as those in the instruction example. Indeed, in most cases, the manipulations are precisely instructional differences designed to alter subjects' goals and beliefs in the most straightforward manner. Questions that arise about the interpretation of the results invariably are concerned with such issues as those outlined several pages ago under problem (1), where there may be a dispute over which component is being affected by differences in beliefs; or under problem (2), where one might want to attribute certain regularities to beliefs and goals, despite the fact that the relevant beliefs are extremely difficult to change (sometimes for good reason; changing one's beliefs can have wide ramifications and may disrupt many intimately related, deeply held existing beliefs, which is why it takes so long to change scientific beliefs).

One way to view the penetrability criterion when it is used as a methodological principle is as a method that allows us to exploit some of our most reliable and stable intuitions, those concerned with the description under which certain regularities should be addressed. This, in turn, allows us to drive a wedge between cognitive processes and the part of the cognitive system fixed with respect to cognitive or semantic influences. Thus intuitions concerning which phenomena are cognitive in the requisite sense (as in the case of the effects of instructions) are used as a "forcing function" to spread the underlying constraint into the details of operation of a system where our intuitions are notoriously suspect and where, for example, our intuitions are susceptible to such traps as those involved when we reify the objects of our thoughts and images as though such properties actually were properties of the mental processes (see chapter 8). This methodology is similar to that involved in the study of grammar, where intuitions of clear cases of well-formedness are used to help infer the deeper structure of the language code where we have few valid intuitions.

Seen this way, the cognitive-penetrability condition, *qua methodological principle,* amounts to nothing more than a closure principle for the domain of phenomena explainable in terms of rules and representations, a way to capitalize on the original conception of a semantic level explanation, by applying the distinction consistently throughout the model-building task. The deeper issue of whether a distinction is obtainable at that juncture is one that must be addressed separately. In the end, the only verdict is the one issued by the success—or lack of it—of attempts to build theories based on these assumptions. For now, all I offer are some arguments that the view is at least plausible, given what is known about psychology and what is internally consistent. It is to this task that I now devote a brief aside.

Is Everything Cognitively Penetrable?

Whether the alleged cognitive penetrability of virtually any process, as implied by the examples we have considered, threatens the autonomy (or "level") thesis depends on the nature of the interaction between levels. It is a question of whether the "mixed vocabulary regularities" observed involve the same sort of explanatory principles as those at the physical or symbol level, on the one hand, or the semantic level, on the other. If they do, then it would seem that there is nothing special about the distinction between these principles beyond any other differences that might mark off subsets of explanatory principles in psychology.

The examples presented in the preceding chapters make it clear that such semantic level principles as inference, though carried out by symbol-level mechanisms, differ considerably from either the nomological laws that govern physical-level properties or the symbol-level mechanisms themselves. For one thing, they are interpreted semantically, which is to say, certain regularities among representations such as beliefs and goals are captured only in terms of the meanings of the symbolic expressions. In other words, there are regularities over equivalence classes of symbol structures that are not necessarily expressible in terms of the properties of the symbols—for example, the principle of rationality (*Why that particular rule?—Because, given what I know, it is likely to enable me to achieve my goal.*). Even more obviously, however, semantic level principles differ considerably from physical laws; the latter must be stated over bona fide physical properties (or, at least, projectable properties), whereas the former apply to open-ended classes of physical properties. There is no limit on the combinations of physical properties that can be used to instantiate, say, modus ponens. Consequently, the categories of both semantic level and symbol level generalizations cross classify those of physical level generalizations. That is why the two sets of principles have nothing in common; they are merely related by the happenstance of design or instantiation.

If that is the case, how can it be that there are "mixed-vocabulary principles"—or laws at all? There must be an interface between semantically interpreted symbols and physical properties; that's what perception is. On the other hand, the attempt to explain perception by linking percepts directly to the perceived properties of the world (as was Gibson's goal) clearly fails, for reasons not irrelevant to the issue at hand. It fails because the causally characterizable link must be highly constrained. This causal link is a very small part of the relation between percepts and the world. The rest of the relation is mediated by inference, or semantic level principles, which is also true of putative "mixed vocabulary principles," and for exactly the same reasons. To introduce these reasons, I briefly anticipate some of the arguments developed in chapter 6.

The interface between physical and semantic principles is a special, functional component (instantiated in the functional architecture) called a *transducer*. A transducer is not a particular organ; rather, it is identified functionally. Its exact location, if it is locatable at all, is no more a matter of cognitive theory than is the location of various codes or other encoding functions in the brain. A transducer, however, is one of the more important basic functions, because one's cognitive theory depends to a great entent on assumptions made about the transducer. For that reason the entire chapter 6 is devoted to an analysis of transduction.

In analyzing transduction I conclude that what can count as a transducer must be strictly constrained in several ways. Following are two central constraints. The input (or output, depending on whether it is an efferent or afferent transducer) must be described in physical terms; and the transducer function must be input bound (in the case of an input transducer, this is referred to as being "stimulus bound," whereas, in the case of an output transducer, it must be "symbol bound"), which means it must produce a particular output *whenever* it receives a certain input, regardless of its state or the context. Now we see that the problem with the view that everything is cognitively penetrable is that it is exactly (though in the reverse direction) the Gibsonian view that everything we see is "directly picked up." The reason Gibson's view cannot be sustained is that very few properties (in particular, only certain functions over physical properties) are directly picked up (see Fodor and Pylyshyn, 1981). The other perceived properties are inferred. The same holds in the case of output transduction.[2]

[2] In the visual-detection case, only those properties qualify to which the system is stimulus-bound, and of these, only properties converted to symbolic form by the transducer

Consider now the claim that believing, say, that the Internal Revenue Service has scheduled an audit of your books causes certain acids to be secreted in your stomach (a plausible claim, I suppose). This cannot be one of those mixed-vocabulary explanatory principles, because it is not counterfactual supporting. Specifically, it is not a possible transducer function because it is not input-bound. Even though the relation between belief and reaction may be quite common, it is surely the case that under different conditions of subsidiary beliefs the principle claimed would not hold. For example, if you had been scrupulous about keeping your books, if you had had no income that year, if you believed (erroneously) that the IRS does audits in order to award prizes for the best-kept books, and so on, it would be unlikely that the acid secretions would occur. Evidently, the function from the belief under consideration to the physiological reaction is not transducible.

Perhaps, I have simply not put the principle precisely enough. Maybe it's not *that* belief but the belief in the imminent threat of being fined that is transduced. Whereas that belief, too, seems penetrable by other beliefs (you might believe you have so much money that such a fine would be immaterial), this suggestion seems on the right track. What we need is to identify basic, transducible, cognitive states. The apparent empirical fact is that there are few of these, certainly very few in comparison with the number of possible beliefs. If that is the case, then beliefs have their physiological effect in the same way percepts are generated from optical inputs (except conversely). They take part in inferences, the end product of which is a special, cognitive state that happens to be causally connected to a physiological state of special interest (because, say, it causes ulcers). The last link is completely reliable, since the cognitive state in question happens to be type-equivalent to a physiologically described state. This, at least, is a possible explanation why so few beliefs appear reliably to cause identifiable physiological changes: very few cognitive states are type-identical, or even largely coextensive with, physiologically described states.

Another consideration should be kept in mind in connection with

are said to be transduced. What this means is that it is not enough for some part of the organism merely to respond to a certain property P for there to be a transducer for P; it must also be the case that the response to P is in a class of properties that are *functional* for the organism. In other words, it must lead to an endogenous property that corresponds to a distinct, symbolic or computational state. That is what it means to say that the transducer "generates a symbol." In the case of an output transducer, only the subset of states corresponding to distinct symbols or computational states count as potential inputs to a transducer, and there may be transducers only for a small subset of potentially transducible states. This could mean that only a small subset of the system's symbols are transduced.

such claims as that particular beliefs affect, say, the digestive system, therefore, that the digestive system appears to be cognitively penetrable. This example, and many others like it, trade on a natural ambiguity concerning the system we are referring to when we mention digestion. As I have emphasized, when our concern is with explanation, the notion of system carries with it a presumption concerning the taxonomy under which we view the system's behavior. The system under scrutiny when we are concerned with the chemical interactions among substances in the digestive process is not the same system as the one we examine when we are interested in the effect of beliefs, even though the two may be partly or entirely coextensive; that is, they may be located in the same place in the body.

The system that takes nourishment, hydrochloric acid, and other fluids as input and provides glucose and other substances as output is a different system from one which, in addition, has as input the belief that the IRS is about to arrive. For one thing, the latter process must have access to other beliefs that make this news threatening, for example, *I did not keep all my records; I read somewhere that someone was fined a substantial amount of money as a consequence of an IRS audit.* The latter system *is* cognitively penetrable and *does* have to appeal to rule-governed processes, whereas the former does not.

If you have such a biological process as digestion, whose inputs and outputs are under a biochemical description, rather than being under some cognitive interpretation, say, as codes for something, the only way to explain how this process occurs, hence the only way to explain how it can be systematically changed, is to consider all its inputs and intermediate states under a biochemical description, or, at least, a non-cognitive one. If you take the inputs and outputs of a system under a biological description, then, by definition, its regularities will be subsumed under biological generalizations. No lawlike (counterfactual supporting) generalizations contain both cognitive and biological descriptions, not, that is, unless the cognitive categories happen to be coextensive with the biological categories.

The claim here is not that cognitive states do not cause biological states. Indeed, every state of a cognitive system is simultaneously a biological state and a cognitive state, since it is an element of both a cognitive- and a biological-equivalence class. Since, however, the classes are distinct (that's why they are described using distinct vocabularies that typically cross classify the states), and since biological regularities are stated over biological descriptions (that's what makes the descriptions biological), any explanation of how a biological input-output function can be altered must do so under a biological description of the influencing event. Thus, if we want to explain the influence of some event taken under a cognitive or semantic description, on a biological process such as digestion, we must first do something like discover a relevant biological property that happens to be coextensive with a certain class of cognitive descriptions.

The explanation of such cases as how digestion is affected by beliefs must proceed in three stages. First, we must explain how the variation in some biochemical or physical property (say the concentration of hydrochloric acid) causes digestion to change in theoretically predictable ways. Second, we must show that all the beliefs in question are related by a rule or cognitive generalization to a certain, specifiable class of cognitive states. Third, we must show that this last class of states is coextensive with—or causally connected to properties that are coextensive with—the biochemical or physical property mentioned in the first stage. (The last stage provides a description of the operation of a transducer.) My point here is that, to provide an explanation of the way in which what *appears to be* cognitive penetration occurs in such cases as this, where the function in question is given under a physical-level description, we must identify a system which contains a component that responds in a principled way to events under a cognitive- (or semantic-level) description, together with a cognitively impenetrable function, such as digestion itself, whose variations can be explained by biological and physical laws.

To make this point more concretely, let us consider a comparable case involving a computer. Suppose a robot is provided with the capability of altering its own hardware architecture, say, by mechanically removing an integrated-circuit chip when it is faulty and replacing it with a new one from a storage bin. Clearly, an account of how this is done requires both a description of the computational process, in terms of the program and data structures, and a description of the physical manipulation process. The first part of the account is entirely symbolic, whereas the second is physical and mechanical. The overall account of altering the hardware architecture by symbolic processes involves showing how the symbol structures affect transducers, which affect the behavior of the manipulators (here, the regularity is explained under a mechanical description), which, in turn, affect the architecture by changing the circuitry.

Note that, in this case, we cannot explain the change in architecture in terms of the program alone—even if we do not know independently that physical manipulation is involved—for the same reason that we cannot explain the particular changes in beliefs that result indirectly from deciding to take a drug or conduct self-psychosurgery. In both cases, the nature of the changes produced are, in an important sense, not accountable in terms of the content of the cognitive influence, because the nature of the changes depends crucially on physical properties of the world (for example, to provide an explanation of the changes that took place, we must specify what chips were previously placed in certain, actual physical locations), whereas content is independent of such physical properties, inasmuch as the same content can be conveyed in arbitrarily many physical ways. Therefore, in cases such as this, a description of nonsymbolic physical activity and the relevant physical laws must occur as part of the explanatory account, as well as a description of the noncomputational environment—for example, the location of various integrated circuit chips in the vicinity. Such factoring of the process into a symbolic stage, a nonsymbolic (nonpenetrable) stage, and transduction stages is precisely what is required in such examples as digestion. Note that, while decomposition into stages is required for the independent reason that different regularities hold at each stage, decomposition also leads quite naturally to separation of the process into computational and noncomputational modules, together with well-defined and motivated interfaces between them.

The conclusion I come to is that examples of mixed-vocabulary regularities must be explained by decomposing them into components. For reasons independent of the problem of dealing with the cognitive-penetrability criterion, we find that by explaining such cases we go back to making a distinction between functions that operate on the basis of semantic-level or symbol-level principles and functions that must be explained in physical-level terms. We return to this recurring theme when we discuss perception (chapter 6), analogue processing (chapter 7), and imagery (chapter 8). Thus the threat that everything will turn out to be cognitively penetrable seems remote in light of such considerations, while the distinction of levels remains a basic working hypothesis of cognitive science.