Brian Cantwell Smith
brian.cantwell.smith@utoronto.ca
March 3, 2018

The Correspondence Continuum

Brian Cantwell Smith

Abstract A1

It is argued that current techniques for analyzing the semantics of knowledge representation systems in Artificial Intelligence (AI) are too rigid to account for the complexities of representational practice, and unable to explain intricate relations among *representation*, *specification*, *implementation*, *communication*, *modeling*, and *computation*. Doing justice to such phenomena challenges such staples of traditional analysis as clear use/mention distinctions, strict metalanguage hierarchies, distinct "syntactic" and "semantic" accounts—even logic's notion of model-theory itself.

By way of alternative, the paper advocates the development of a *general theory of correspondence*, able to support an indefinite continuum of circumstantially-dependent representation A2 relations, ranging from fine-grained syntactic distinctions at the level of language and implementation, through functional data types, abstract models, and indirect classification, all the way to the represented situation in the real world. The overall structure and some general properties of such a correspondence theory are described, and its consequences for semantic analysis surveyed.

1 Introduction

Certain genitive phrases of the form ‘α of β’ are ambiguous. On what is known as the *subjective genetive* reading of “love of children, for example, it is the children who do the loving, as in (1). On the *objective genetive* reading, in contrast, children are recipients of the affection, as in (2).

1. Though bitter from years of being ridiculed by adults, the old man was grateful for the *love of children*.
2. Though increasingly impatient with his peers, the old man never lost his *love of children*.

The issue arises when the head noun phrase α (‘love’) signifies an asymmetric two-place relation, since it is then unclear which argument place is filled by the β term following ‘of’. As shown in these examples, the distinction is generally clear-cut, with the intended reading selected by context (this is why it is a question of ambiguity, not vagueness).

The phrase “the representation of knowledge” is of this ambiguous type. Oddly enough, though, in this case it is not clear which reading is intended. Is knowledge being represented (objective genitive), or is knowledge doing the representing (subjective genitive)? Both interpretations seem reasonable. For example, suppose we build a medical artificial intelligence (AI) system called DOC, using FKRL, our favourite knowledge representation language. On the objective reading, the ingredient structures would be viewed as *representing DOC’s knowledge*, presumably implying that a semantics for FKRL should map FKRL structures onto knowledge (or perhaps onto a set-theoretic model of it, such as a possible-world structure). On the subjective reading, in contrast, DOC’s knowledge, embodied in FKRL structures, would *itself* be taken as representational. In this case semantic analysis would map the representational structures (assumed to implement or constitute the knowledge) onto the states of affairs in the world that DOC knows about—states of affairs involving drugs, diseases, and diagnoses.

To add to the confusion, it is not even exactly clear, in the knowledge representation case, what the difference between the two readings would come to. It seems that a possible world structure modeling belief might be the same as a structure modeling the states of affairs that the belief is about. And yet beliefs and worldly states of affairs are not the same. The former, for example, are psychological, the latter not (at least in general). Thus, whereas an erudite doctor might be said to possess great knowledge, it would be senseless to say that they A1 possess great states of affairs.

Some of the confusion has a simple source: *both* 'representation' and 'knowledge' designate asymmetric, relational notions. Furthermore, the two relations are of the same general type; they both characterize phenomena that are *about* something—phenomena that refer to the world, that have meaning or content. For example, to say that a series of marks on a page is a representation of Winston Churchill is to say that there is some relation between those marks and the late British Prime Minister. Similarly, to say that your lawyer's knowledge is faulty is to comment on the relation between what is going on inside the lawyer's head and what is going on outside. Because they are both based on an underlying (asymmetric) relation of content, representation and knowledge are considered to be semantic or intentional notions (other intentional notions include language, belief, model, theory, specification). But to say that is not to say very much, at least not yet. It certainly does not explain how representation and knowledge differ. Nor A4 does it clarify our starting question of how, in the knowledge representation case, they are supposed to relate.

This paper will try to sort this all out. Specifically, taking semantics as the general enterprise of describing intentional A5 phenomena, I will address the question of what it is to give a semantical analysis of a knowledge representation system. I.e., whereas most semantical analyses focus on particular languages or systems, concerning themselves with particular types of semantic entities or structures—possible world structures, partial situations, etc.—my concern will be with the overall framework in terms of which such analyses are conducted.

There are several reasons this is an urgent task. The first I have already discussed: as implied by the confusion in the name, knowledge representation involves several interacting intentional notions, which should be sorted out. Second, it is increasingly thought necessary to give semantical accounts of proposed representation systems, in order to convey rigor and coherence onto what would otherwise be viewed as ad hoc symbol mongering. In 1974 Pat Hayes, long a champion of this view, called AI's reluctance to provide semantical accounts for representation schemes "a regrettable source of confusion and misunderstanding,"[1] and went on in 1977 to write as follows:

> "One of the first task which faces a theory of representation is to give some account of what a representation or representational language *means*. Without such an account, comparisons between representations or languages can only be very superficial. Logical model theory provides such an analysis."[2]

In writing these words Hayes was defending logic against what he took to be the a-semantical orientation of the proceduralist tradition. In this he seems to have succeeded: similar sentiments have since gained widespread allegiance. We A6
should certainly understand anything so popular.

[†] In the original version most references were included inline; it seems more elegant here to place them in footnotes. I have therefore indicated the numbering of the original footnotes in square brackets where appropriate (e.g., at the beginning of footnote 3, labeled '1' in the original).

[1] Hayes, 1974, p. 64.

[2] Hayes 1977, pp. 559.

On the other hand, this very success leads to the third reason for the present investigation. I believe that current theoretical tools, particularly the traditional model-theory that Hayes cites and most everyone uses, are inadequate to the knowledge representation task, and need substantial revamping. Perhaps ironically, many of the problems I will canvass are foreshadowed in Hayes' original papers—including the relation between so-called propositional and analogue representation A7, to take just one example, which has yet to be adequately reconstructed. Logical model theory, which at least in contemporary formulations does not address analogue questions, has if anything gained momentum as the knowledge representer's semantical technique of choice.

Fourth, and finally, many of the lessons learned in the knowledge representation case will hold for all computational systems, and will even impinge on general semantical analysis; so there is a certain universality to the inquiry.

Part I — State of the Art

2 A Model of Knowledge Representation

I will adopt a **two-factor** model of knowledge representation, A8 as pictured in figure 1 (next page). An agent, computational or human, is taken to comprise a set of internal structures, states, or aspects, that have some sort of content, and at the same time A9 play a role in engendering the agent's overall behavior. In order to focus on their internal nature, I will call these structures **impressions**, to distinguish them from **expressions**, assumed A10 to be elements of an external language. Think of impressions as data structures, as elements of a knowledge representation language, or as partial mental states—not much will depend A11 here on details. The essential point about impressions is that they have two partially independent, though coordinated, properties.

First is what I will call **functional role** (or ‘role’, for short)— A12 indicated as α in the diagram. Impressions must arise, somehow, in virtue of the history and coupling of the agent to its environment, and must (perhaps in conjunction with other factors) give rise to the system's future activity or behavior. Furthermore, as well as having these backwards- and forwards-looking aspects, impressions must be *causally efficacious in the present*—must bump up against each other, or be manipulated by some sort of internal agency, so as to constitute the whole of which they are the parts. So a given impression, such as one expressing the fact that a robot does not have much time left until it needs a recharge, might arise from the integration of information gleaned from internal sensor readings, engender inference involving time and expected electrical use, and lead the robot to scramble around the hall in search of an electrical outlet.

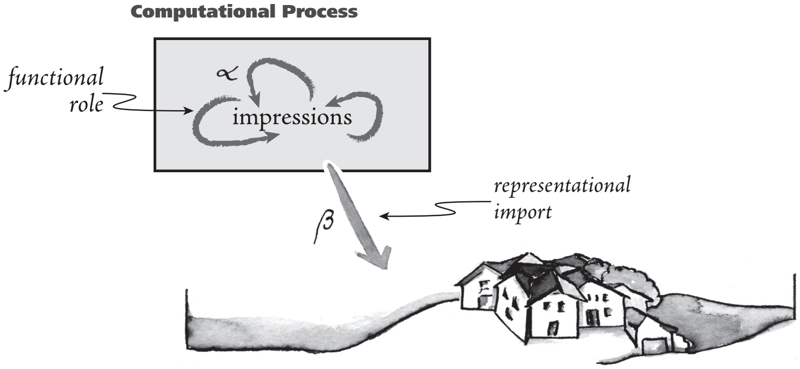


Figure 1 — A two-factor model of knowledge representation

Functional role is not enough, however. In order to count as *representations*, as opposed merely to being causal ingredients like the cam shaft in a car engine, impressions must also stand in a content relation to the states of affairs in the world in which the agent is embedded. I will call this second factor **representational import** (or just 'import', where the meaning is clear)—indicated in the diagram as β.

Representational import is not an *alternative* to functional role, or a particular *kind*; it is something additional. Thus whereas the level of sap in a maple tree arises from a complex history involving the weather, structure of the trunk, etc., and gives rise to complex future behavior, such as amount of sugar produced, density of new foliage, etc., that is about all there is to say about it. In spite of being correlated with facts in its environment, sap does not have any representational import partly because the correlation is too strong (sap cannot be A13 wrong), and partly because no concepts are employed (sap does not represent the world as *being one way as opposed to another*; it is merely locked into it as a totality). In contrast, for an impression to represent spring's being on the way, there would have to be an additional regularity relating its structure to the structure of that fact—a regularity that would be missed in an isolated account of functional role.[3]

A14 For example, suppose I have the impression that water conducts electricity. All kinds of backwards-looking functional roles could have led to this: my own hapless experience trying to heat the bath water with an electric iron; stories I have been

[3] [1] Saying just what distinguishes representational from purely functional ingredients is a difficult philosophical problem. My own emphasis on the two criteria cited here—a certain "disconnection" between representation and what is represented, and the claim that a representation must represent the world as being a certain way—is discussed in Smith (forthcoming (a), chapter 4) and in Smith (forthcoming (b)). The issue has been addressed by many writers in the philosophy of psychology, such as Fodor, Searle, and Stich, especially in assessing the relation between proposed functional and representational theories of mind. «Refs?» Computational readers will note, however, that many of these philosophers get at representation by analogy to computation, whereas my own view is approximately the opposite: that we must get at computation by first understanding representation. There is more overlap in subject matter than concurrence in views.

told; books I have read; deductions from knowledge of the ionization potential of molecules held together by hydrogen bonds. Similarly, at least within wide limits, there is no predicting what forward-looking role this impression might give rise to: things I might say, or situations I may strenuously avoid, such as climbing onto high-tension wires during rainstorms. The point is that, in spite of this richness of role, including inferential role, there remains a striking and relatively simple uniformity connecting the impression and the fact it represents—the most penetrating regularity in terms of which to explain my behavior. In brief, it is the connection between the impression itself and *the fact that water conducts electricity*. This is the regularity of content or representational import.

The two factors must be coordinated in a special way: the A15 states of affairs that the impression represents (its import) and the behavior that it gives rise to (its role) must be such that the agent can be truthfully said to *know* or *believe* the fact, A16 which involves being able to act in accord with it, etc. The trick, in spelling this out, is to tie the two roles together into an integral whole without thereby undermining the integrity of the distinction between them—a project that requires combining traditional semantical techniques with the AI and philosophical literature on knowledge as action, pragmatic reasoning, and even causal theories of reference. I will not attempt that integration here, but will merely call the coordinated combination of factors the **full significance** of an impression. A17

In Smith (1982a & 1985),[†] I labeled this two factor orientation to representational significance the **Knowledge Representation Hypothesis**. In the philosophy of mind an analogous view has been labeled a *dual-component* semantics for psychology.[4] Technical variations have appeared under various descriptions; what is perhaps most striking is its familiar-

[†] Both references are to the 3Lisp dissertation, parts of which are included here in Chapter 2.

[4] Field (1977, 1978); Loar (1982); Block (1985).

ity in even the familiar realm of formal logic. In a traditional proof-theoretic framework—say, if the agent was an implementation of a natural deduction theorem-prover for first-order logic—one might view representational import as the *semantics* of an expression, and functional role as its *proof-theoretic* consequence. This last characterization, however, misleadingly suggests that the full significance of a representation system must satisfy the following two constraints:

1. That the two factors be essentially *independent* (in which case I will call the representational system *declarative*); and A18
2. That functional role arise solely from *syntactic* properties of the representational structures.

Adherence to a general two-factor analysis, however, in no way commits one to this particularly strong way of dividing things up.[5] 3Lisp, for example,[6] a simple programming language designed within a two-factor framework, explicitly violated both assumptions: import and role were both essentially semantic;[7] it was also shown that they were theoretically explicable only in intimate conjunction.[8] Other analyses, such as that suggest-

[5] [2] David Israel has challenged the view, almost universally held in AI, that the notions of *proof*, *deduction*, *inference*, etc., even in mathematical logic, should be conceived in syntactic terms. «Refs» This syntactic orientation is not even universally accepted within what is called formal logic, since it rests on only one of many possible readings of the term 'formal' (see Smith [[forthcoming (a)]].

[6] Smith (1982a, 1984); parts of the former as included here in ch. 2; the latter, as ch. 3.

[7] [3] Reasons why the functional (procedural) parts count as semantic are spelled out in Smith [[forthcoming (a)]].

[8] [4] First factor derivability (⊢) and second factor satisfaction are traditionally tied together through entailment (⊨) and proofs of soundness and completeness, but these particular notions are coherent only as a kind of global constraint on what are otherwise locally independent factors. The kind of "intimate conjunction" employed in 3Lisp, and being imagined here for more general models of reasoning and computation,

ed by Barwise and Perry,[9] propose alternative ways of tying content and behavior together. In fact it is partly because there are so many ways of getting at roughly the same intuition that I have presented it here somewhat abstractly.

The two-factor nature of knowledge representation is the most important aspect for semantical analysis to clarify. In order to make sense of current semantical techniques, however, we need another distinction, which cross-cuts it.

Especially in the philosophical literature, semanticists sometimes distinguish the **meaning** of a structure from its **content** or **interpretation** (not, at least not in any straightforward way, to be confused with the computer science notion of interpretation; see §5, below, and Smith (1984)[†]). Very roughly, the former is what all instances or uses of a given structure type have in common; the latter, what a particular use or instance of that type refers to, or gets at, in all its specificity. Typically, facts about the context or setting provide the additional information that gets from meaning to interpretation. So for example the first person pronoun 'I', under this A19 analysis, has the meaning of referring to whoever uses it: when Mick Jagger says 'I' he refers to himself; if you do, you refer to yourself. This is why two people can scream at each other "I'm right; you're wrong!"—they both use the same sentence, and *the meaning is constant*; it is the respective interpretations or contents that are contradictory. So we might model the meaning of 'I' as the following function of speakers (s), times (t), and locations (l) as follows:[‡]

$$[\![\text{I}]\!] = \lambda S, T, L \cdot S \qquad \text{A20}$$

. . .

is one of much more local interdependence. As pointed out in Smith (1982b), computational practice already encompasses a wide range of such local interactions; see also Smith (1987).

[9] Barwise and Perry (1983).

[†] Included here as ch. 3.

[‡] Using '[\![...]\!]', as is standard, to signify the interpretation function.

In a given situation of use (speaker σ at time τ in location π) the interpretation would thus be σ.

It is tempting to identify meaning as the semantics of *types*, interpretation as the semantics of *tokens*—but the second of these is misleading. John Perry, for example, has imagined a A21 case of two deaf mutes, so poor they must share a single tattered card saying *I'm a poor deaf mute; won't you give me some money*.[10] Standing together at the street corner, they alternately hand the card to passers by. Each time the card is used, the words 'I' and 'me' change their reference: one token, constant meaning, changing interpretation. Similarly, consider an analogous computational example: a machine with a single distinguished internal structure used to mean 'now'. The meaning is constant, and the particular structure (token) may persist, but the interpretation changes with each passing nanosecond. A22 Uses, or utterances, are what have interpretations; not concrete instances or tokens.

The meaning/interpretation vocabulary is not common in the AI or computer science literature, but the circumstantial dependence with which it deals is ubiquitous. Even the simple inclusion of explicit environment and memory arguments in denotational analyses of programming languages[11] manifests a sensitivity to the importance to interpretation of contextual factors. In Smith (1986)[†] I lay out a whole variety of ways in which the content of computational structures, including impressions, can depend on facts of circumstance or context: internal facts (what program is running, how other internal structures are arranged, etc.), external facts (where the computer is located, whom it is conversing with, etc.), and even some facts that seem to cross the boundary (what time it is). The importance of these kinds of circumstantial dependence will be assumed in what follows.

[10] «Ref»

[11] See for example Gordon (1979).

[†] Included here as chapter 5.

Furthermore, both aspects of significance—functional role and representational import—can be circumstantially dependent. What ¬FLIES(x) means, when attached to the BIRD node in a default reasoning system, and what inferences it leads a system to draw, can both depend on the presence or absence of other intermediating impressions. I will use **functional meaning** and **representational meaning** to get at the respective factors of an impression's significance abstracted away from details and circumstances of particular instantiation or use. Similarly, **functional content** and **representational content** will refer, respectively, to the actions a use of an impression actually engenders, and to the situation it actually represents.

Given these distinctions, my overall question is this: *what would a semantical analysis be of the full significance of impressions?* In the broadest terms, it will clearly have to distinguish import and role, meaning and content, and show how they all come together into a coordinated whole. But we need details. I will proceed in steps, concentrating first on representational import. Later I will return to the question of how to tie it together with functional role.

3 The Present State

Virtually all the theoretical techniques in our current semantical arsenal were developed to deal with representational import. In particular, present practice proceeds roughly as suggested in figure 2. First, a source domain is identified as the set of elements for which a semantical analysis is to be given. Traditionally, this is called the *syntactic* domain; in the knowledge representation case it is the set of impressions comprising the agent (I will talk more about the difference in a moment). Second, a *semantic* domain is similarly identified, taken roughly to be what the elements of the representational domain, expressions or impressions, are about (more about 'aboutness', too, in a bit). Third, the semantic relation between domains, usually A23 called the *interpretation function*, is then described *extensionally*, A24 in the sense that particular elements of the syntactic domain A25 are mapped, piecewise, onto the corresponding particular elements of the semantic domain. It may be, in the theorist's actual presentation of the semantic relation to the reader or audience or a colleague or whatever, that considerable information about the structure of this relation will be manifested, but strictly speaking this additional structure is not part of what is provided (or perhaps, to borrow from the Tractatus,[12] we could say that it is *shown* but not *said*). Just as for functions and relations more generally, piecewise correspondence is assumed to be sufficient, at least for theoretic purposes.

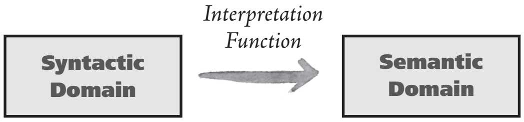


Figure 2 — The Standard Semantical Model

So far, however, I have not said enough to distinguish the extensional analysis of a semantic relation from the extensional analysis of any old relation at all. But in practice more assumptions are adopted. I will label as **classical** those seman- A26 tical analyses that accept (which I do not) the following additional claims:

1. *Compositional semantics:* The elements of the represen- A27 tational domain are assumed to be *linguistic*. Debates rage over what language is, but at least the following seems agreed: complex linguistic elements are taken to be linear sequences of some sort (strings, utterances, whatever), with an inductively specified recursive structure founded on an initial base set of atomic elements called a *vocabulary*, and assembled according to rules of composition specified in a *grammar*. Furthermore, the interpretation relation is usually defined *composition-*

[12] Wittgenstein (1921).

ally, so that meanings (not contents!) are assigned both to the vocabulary items and to the recursive structures engendered by the grammatical rules, in such a way that the meaning of a complex whole arises in a systematic way from the meanings of its parts. A particularly strong version of compositionality requires that the meaning of a whole be definable, often by function composition, in terms of the meanings of the parts, but other possibilities, such as that the whole's meaning be characterized, or even just constrained, by systems of regularities among the parts, are growing in popularity. We need not take a position here on details; I will assume that these are *variants* on classical approaches.

2. *Clear use/mention distinction* and a *hierarchy of metalanguages*: In a case where the elements of syntactic domain $\mathcal{S}$ correspond to elements of semantic domain $\mathcal{D}_1$, and the elements of $\mathcal{D}_1$ are themselves linguistic, bearing their own interpretation relation to another semantic domain $\mathcal{D}_2$, then the elements of the original domain $\mathcal{S}$ are called *metalinguistic*. Furthermore, the semantic relation is taken to be *non-transitive*, thereby embodying the idea of a strict *use/mention* distinction, and engendering the familiar hierarchy of *metalanguages*. This distinction is motivated by such obvious facts as that the six-character quoted expression «"Nile"»[†] designates a short word, which in turn designates a long river, but from those two facts it does not follow—nor is it true—that the original six-character expression «"Nile"» itself designates the river.
3. *Parameterization to deal with contextuality:* As suggested in figure 2, the interpretation relation is typically taken

[†] As in the versions of the 2Lisp and 3Lisp papers presented in Part B, I use French quotes ('«' and '»') as string quotation operators, in order to reduce confusion, when the strings being mentioned themselves contain apostrophes and quotation marks. See «…».

to be a *function*, implying that the import or content of an expression is not ambiguous. But ambiguity is a relative term: a linguistic element may look ambiguous if the circumstantial dependence of content has not been fully articulated, and may therefore be resolved by the meaning/interpretation distinction. We have already seen how the functional assumption is generalised to handle such complexities: whatever information disambiguates a given use of an otherwise ambiguous expression is included as a parameter of meaning; content is then obtained from the meaning by fixing that parameter. For example, the interpretation of the indexical expression 'I', discussed above, was parameterized on speakers (formally, for reasons to be explained in a moment, it was parameterized on speakers, times, and locations—though only the speaker affected the resulting interpretation). Similarly, if 'grue' means *blue if used before some time* t_0, *and 'green' afterwards,*[13] then its interpretation would be parameterized on time of use, leading to its being assigned roughly the following meaning:

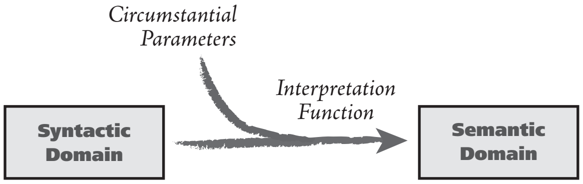


Figure 3 — Parameterized Interpretation

$$[\![\textsc{grue}]\!] = \lambda s, t, l \cdot \textit{if } t < t_0 \textsc{ then blue else green}$$

Thus the true situation is more accurately pictured by figure 3, with dependence on circumstantial or contextual factors folded into the interpretation. As mentioned earlier, the discussion in Smith (1986)† was

[13] See Goodman (1983).

†Chapter 5

intended to show how facts about both internal and external context can affect interpretation in this way.[14]

4. *Model-theoretic treatment of the semantic domain:* It is not necessary—not even usual—to require that the semantic domain $\mathcal{D}$ be the real domain that the expressions are about. Rather, $\mathcal{D}$ is required to be a set-theoretic structure, viewed as a *model* of the real semantic domain.

This last assumption serves a variety of useful functions: it means that semantical analysis can remain "purely" mathematical, rather than having to spell out complete metaphysical assumptions about the true nature of the world. So for example a belief or proposition might be modeled as a function from possible worlds to truth-values, without the theorist needing to believe that that is what beliefs *really are* (but of course they are not functions in fact: it is entirely reasonable to ask "What are your friend's most strongly held beliefs?", and absurd to ask "What are your friend's most strongly held functions from possible worlds to truth-values?"). Similarly, in the semantical analysis of a language used to describe Turing machines, the semantical domain is usually taken to be sets of quadruples, not actual devices complete with tapes, read/

[14] [5] Functional parameterization deals with circumstantial dependence, but in a specific and limited way. In particular, by assuming that the linguistic element, plus circumstantial facts, together determine the interpretation, it implies that this is the direction of "information flow"—that understanding proceeds from knowledge of language, plus knowledge of circumstance, to knowledge of content. In practice, however, the flow can easily run in the other direction: someone hearing an utterance may know about the situation being described, and use that information to determine the structure of the linguistic element, or of such circumstantial factors as discourse structure. For these and other reasons a genuinely relational theory of meaning and content would be preferable (see Barwise and Perry (1983); I use the functional analysis here only because of its familiarity, and because my current argument is not particularly sensitive to the distinction.

write heads, finite state controllers, and so forth. The quadruples are viewed as a *model* of the Turing machine, and—this is the crucial point—*modeling is assumed to be "free,"* in the sense that the theorist is granted license to engage in unconstrained modeling without having to account for it explicitly in their theory. To put it another way, modeling is invisible through the standard semanticist's glasses.[15]

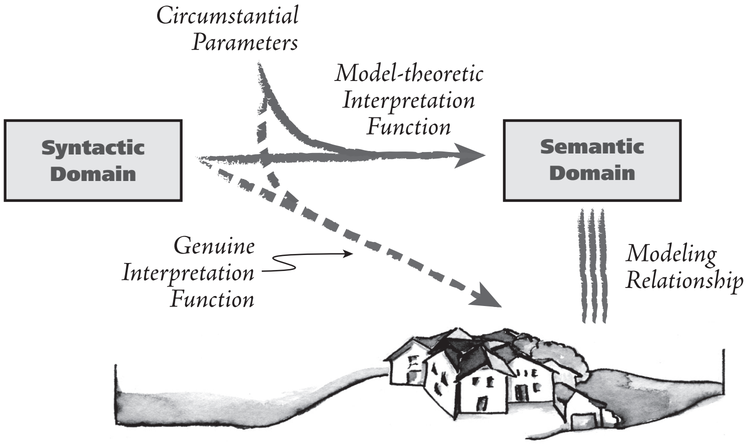

Circumstantial Parameters
Model-theoretic Interpretation Function
Syntactic Domain
Semantic Domain
Genuine Interpretation Function
Modeling Relationship

Figure 4 — Parameterized Model-Theoretic Interpretation

Sometimes, of course, when the linguistic or representational elements are genuinely about mathematical objects—theories of arithmetic, for example, or representations of the factorial function—the true interpretation (called the 'intended interpretation') may be one of the model structures. In general, however, and almost universally in the knowledge representation case in Artificial Intelligence, we are interested in representations of more general states of affairs in the world, such as levels of digitalis in heart patients. So the picture of

[15] [6] Sometimes, as for example in Montague semantics, the syntactic domain is modeled as well, but I will not worry about that here—it is merely an extension of the same points being made.

semantics should be updated as in figure 4 (previous page).

Finally, in discussions to follow, we will encounter complex situations that include both modeling and iterated representation of the sort discussed in the second assumption. So it is important to summarize how the standard picture would look in such cases. Since modeling is typically ignored, such a situation would traditionally be *described* as a strict series of non-transitive denotation relations, each analyzed piecewise. Our comments about modeling might suggest that the true situation is more complex, consisting of a series of non-transitive denotation relations, followed by an indefinite amount of promiscuous modeling. But in fact, since there may be promiscuous (i.e., invisible) modeling at each stage of the language hierarchy, as for example when a language is encoded in arithmetic (as is common in recursive function theory, for example), what we really have is this: a strictly non-transitive sequence, each step consisting of a denotation relation followed by an indefinite amount of promiscuous modeling. This situation is pictured in figure 5.[16]

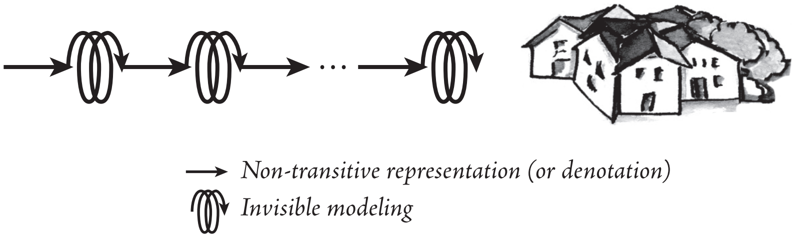


Figure 5— Model-theoretic Interpretation of Iterated Representation

[16] [7] At least in this paper, I do not intend these remarks to challenge the appropriateness of these techniques for the intellectual project for which they were developed: the metamathematical inquiry into the foundations of mathematics. My current complaint is only about its adequacy for use in AI, knowledge representation, and any other situation in which the true state of affairs being represented is in the real and messy world of everyday life.

4 Impressions

The first step, in analyzing the appropriateness, for the representational problems presented in section 2, of the semantical techniques described in section 3, is to decide how we are going to treat *impressions*. Because I specifically introduced the term to cover any internal aspect, state (or partial state), or structure, we want a fairly general answer. Also, it turns out to be a surprisingly complex subject. If we can clear it up first, therefore, subsequent semantical analysis will be that much more tractable.

The most important point is this: *as semanticists*—whether our home discipline is psychology, philosophy, logic, computer science, or artificial intelligence—*we do not yet have any developed theoretical terminology whose primary function is to describe impressions*. In particular, impressions are not necessarily *linguistic* objects, since the notion of language arises from the structure of communication and consensual interaction, not causal ingredients. Nor does mathematics provide any directly applicable notions: mathematical structures are abstractions—Platonic ideals, at least as classically conceived, not fragments or constituents of activity. For example, in discussing two-factor semantical analysis in section 2, I talked about impressions being "causally efficacious." These are not terms in the A28 standard mathematical repertoire; nor, at least in general, are pure mathematical objects thought to possess causal powers. I have introduced the term 'impression' as a small step towards repairing this deficiency (as, in the 3Lisp case, I also did with 'structural field'), but of course it is simply a general, covering term. What we lack is a theory of types of impressions, types of important relations among impressions, analyses of how impressions can simultaneously cause and represent, and so forth. It is not that we are entirely without terms for such things: *data structures*, *data bases*, *knowledge bases*, *data types*, *functions* (in the 'procedure' sense), and *code* are all types of impression, as are more specific AI constructs such as *semantic nets*, *inheritance graphs*, and *taxonomic lattices*. Rather, what we need is a general theory, in terms of which these diverse kinds could be characterized.

Lacking a general theory, what do we theorists do instead? Different things. Perhaps the most common practice, especially in AI and the philosophy of mind, is to treat impressions metaphorically—in particular, as analogous to language. Thus in the cognitive case we have talk about "language of thought," "mentalese," "syntactic" theories of mind, etc.—as for example championed by Fodor, Stich, and others.[17] Artificial intelligence typically follows the same path, talking about "expressions," knowledge representation "languages," etc.—as does anyone who views impressions as *formulae*. In philosophy this stance is commonly referred to as the *representational theory of mind*—a somewhat unfortunate epithet, not because the term 'representation' is inherently so narrow, but because this usage tends, without explicit admission, severely to constrain the notion of representation to its linguistic or even syntactic shadow. Instead I will call it a **linguistic theory of impressions**. Two facts about this theory are important for present purposes: (i) that we recognize its hypothetical nature—the fact that it represents a substantial claim; and (ii) that so long as this language remains metaphorical, we be careful to monitor connotations not necessarily warranted in the new domain.[18,†] For example, in 3Lisp I called certain number-des-

[17] See Fodor (1975), Stich (1985).

[18] [8] Boyd (1979) argues persuasively that metaphorical scientific language can play a role, especially initially, in enabling a community to establish increasingly substantial reference to a new domain. On such an account, the use of linguistic terminology to discuss impressions might, over the years, gradually lose its metaphorical overtones, and take on full-fledged referential connection to this new domain. But as Boyd himself points out, in order for this process to take hold, the metaphor must start out being at least partially correct. My concern in this particular case, as the rest of this section tries to suggest, is that many of the

ignating impressions *numerals*, but the metaphorical nature of the terminology misled me as well as others, causing me to attribute semantical properties to impressions motivated more by linguistic connotation than by genuine functional need (for example, my adoption of a strict use/mention hierarchy, distinguishing the number three, the *impression*-numeral ‘3’, and A29 the distinct *expression*-numeral “3”).

Those bred in the knowledge representation tradition may find the linguistic approach to impressions obvious, but it is important to recognize that it is not universally accepted. It is well known that philosophical debates rage about whether representation is the best notion in terms of which to characterize human mental states. What is perhaps more surprising is the fact that a number of alternative views are advocated even within computational circles. First, many people have realized, in opposition to the linguistic claim in its narrowest form, that there is no need for internal structures to be anything like *identical* to written ones. The mildest position of this sort is John McCarthy’s notion of “abstract syntax”,[19] A30 which effectively amounts merely to a way to free impressions from gratuitous details of notation. I made a stronger move in the same direction in developing 3Lisp, using the term “**structural field**” for the totality of impressions, even though I then described individual impression types using terminology that I now feel was excessively derivative from linguistic analysis. My move was stronger than McCarthy’s not only because the granularity of distinction in the 3Lisp field was less than is usual in even abstract linguistic cases, but also because the mapping between expressions and impressions (as well as that between impressions and the real world or task domain) was taken to be contextually sensitive. (Partly for reasons of circu-

. . .

connotations of the use of linguistic language to describe impressions are in fact unwarranted.

† Cf. also «whatever Rehabilitating Representation becomes».

[19] «Ref»

larity and structure-sharing, the external notation was neither isomorphic to internal impressions, nor complete. Furthermore, in certain complex cases like closures, the impression structure was far more complex than linguistic notation could readily express.)

Other positions on impressions have been proposed. The view embodied in the design of 3Lisp—that viewing impressions as syntactic or linguistic is non-ideal because it commits the theorist to too fine-grained a set of internal distinctions—was not mine alone; it is increasingly supported in various quarters of AI. Two suggested alternatives are of particular importance. Levesque retains allegiance to knowledge representation as a covering notion, but argues for a *functional* analysis of machine states, with explicit reference to the notion of an abstract data type, as opposed to a view of them as comprising "collections of symbolic structures."[20] Apparently more radically, Rosenschein criticises the entire representational stance, A31 which he characterizes as viewing "the state of the machine as encoding symbolic data objects," arguing instead for the notion of a situated automaton, with intentional properties (which he calls "knowledge") defined in terms of "objective correlations between machine states and world states".[21, 22]

Supporting these anti-syntactical proposals, moreover, is the attitude towards impressions adopted in current theoretical computer science. Spelling that approach out is difficult, however, because of a facade of potentially distracting theoretical techniques that are standardly employed, which obscure, from the present vantage point, exactly what is going on. So I will digress from the subject of impressions, for a moment,

[20] Levesque 1984, pp «…».

[21] Rosenschein 1985, pp «…».

[22] [9] History is often repeated, we are told, but here it is being repeated in reverse direction. The gradual shift from functionalism to representationalism in the philosophy of mind is apparently being played out backwards in AI, which started with a very strong representationalist

to examine what computer sciences calls the denotational semantics of programming languages, and then return to the present topic once we have that firmly in hand.

5 Programs, Processes, and Indirect Classification

The abstract data type movement in programming language design, and the denotational approach to programming language semantics, are best understood as attempts to characterize the structure of computational processes in other than linguistic terms. They are motivated by the following obvious fact: when we develop computational processes, we *cannot build processes directly*. Instead, we cause them to come into existence by writing *programs*. In their discourse, AI programmers often gloss the distinction between the program and the process, viewing programs as functional ingredients that are either *inside* processes (a move in which programs are effectively taken to be impressions—partly motivated by the widespread use of interpreted, interactive languages like Lisp), or sit in the background causing them to exist, etc. Such assumptions are betrayed in such informal parlance as *The program is still running, The program reads in a number and then prints out the answer*, etc.

Nonetheless, as every programmer knows full well, *programs*—textual objects that are printed out on paper or on the screen, that are edited with EMACS and other editors, etc.—do A32

. . .

stance, and is steadily moving away from it, towards what are explicitly admitted to be purely functional accounts (see Levesque (1984), Newell (1982), etc.). My own view is that both traditions, in opposite order, suffer from the lack of a full fledged theory of representation. Based on the idea that the only rigorous concept of representation is a narrow, purely syntactic version, they oscillate between its gratuitous detail and consequent semantic implausibility, on the one hand, and contextually insensitive and menacingly behaviorist pure functionalism, on the other. I believe both are inadequate, and conclude that we should free representation from its syntactic strictures, rather than rejecting the notion entirely.

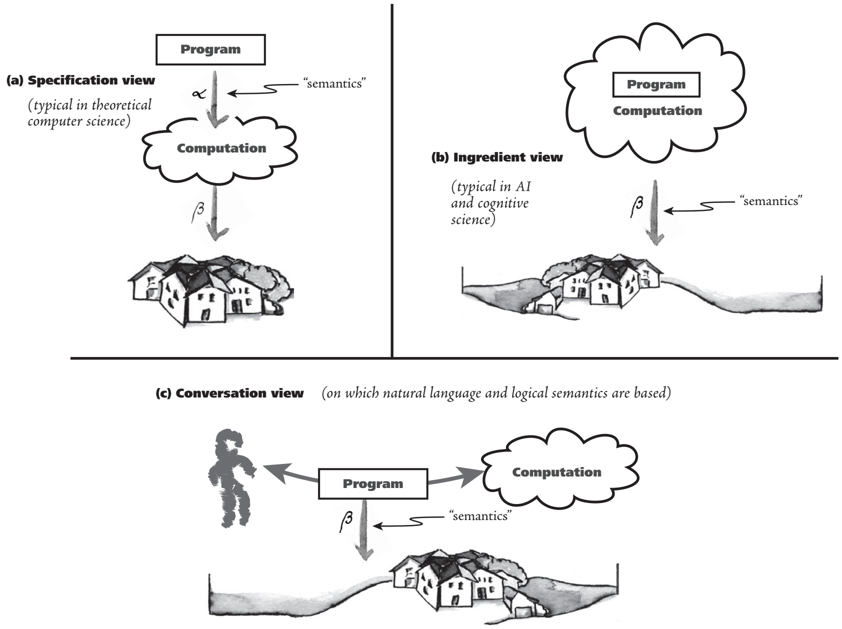

(a) Specification view
(typical in theoretical computer science)
Program
∝
"semantics"
Computation
β
(b) Ingredient view
(typical in AI and cognitive science)
Program
Computation
β
"semantics"
(c) Conversation view (on which natural language and logical semantics are based)
Program
Computation
β
"semantics"

Figure 6— Three Perspectives on Programs

not *do* anything; they are inert. Rather, what happens is that these passive structures are used by interpreters and compilers (about which more in a minute) to engender behavior with appropriate properties. To put it in the blunt vernacular: programs are *run*. A33

The situation is depicted in figure 6. As just stated, the AI and knowledge representation community typically views programs, along with elements of knowledge representation languages, as constituents of or elements within computational processes—i.e., as impressions. I will call this the **ingrediential** view, as suggested in figure 6b. By far the more standard computer science conception, in contrast, is what I will call the **specificational** view, pictured in figure 6a: programs taken as specifications or descriptions of computations, albeit as special descriptions that can be viewed as *prescriptions* by the machine or interpreter. (Different from both is a third, **conversational**, view, depicted in figure 6c, in which programs constitute the dialog or discourse that the programmer has with the machine—a view that I will examine in later, in section 6.)

Thus—and this will greatly affect our semantic analysis—*traditional computer science takes "semantics" to be the job of mapping programs onto processes or (uninterpreted) behavior*. To a computer scientist, that is, contrary to what external observers, philosophers, linguists, etc., might expect, the term "the semantics of a program" does *not* refer to any mapping of or analysis of any intentional relation of the resulting processes onto the world. It is only under computer science's "program to process" conception of programs that "interpreters" are properly named. **A34**

Concerned as I am with knowledge representation, my task is different: exactly to describe that relation with which computer science does not concern itself: that between those (resulting) processes and the worlds in which they are embedded. It follows that, in the traditional terminology, the *semantic* domains of traditional programming language analyses should be the knowledge representer's so-called *syntactic* domains. Confusion over this point amounts to the commission of a use/mention error—exactly the sort of thing that careful semantical analysis is so much at pains to eliminate.[23]

It may seem odd to look for impressions in the semantic domain of a semantic analysis of a programming language. Denotational semanticists, after all, typically deal in semantic domains consisting of abstract mathematical structures—functions, sets, numbers, partial orders, and the like—which do not seem very much like causally efficacious impressions.

[23] [10] Although I will eventually challenge the idea of a rigid use/mention distinction, that does not mean that many so-called "use-mention confusions," such as this, are not serious.

But this apparent discrepancy is explained by the fact, noted above, that traditional denotational semantics is *model-theoretic*. As we have already seen, the model is not the true domain of interpretation, but another structure, typically abstract, set in correspondence with it. As suggested earlier, this technique enables theorists at least partially to avoid exactly the metaphysical questions we are interested in: questions about the true nature of impressions themselves.[24]

Not all questions are avoided by employing model-theoretic strategies, of course, since the structure of the model is intended in some way to correspond to the structure of the impressions. The question is how the correspondence goes (i.e., what is the relation between a set-theoretic structure and an FKRL impression?). To get at the answer, note that modeling is an instance of the rather general practice of describing a set of complex phenomena only by setting them in relation to another, presumably more familiar, set of structures. Barwise and Perry call this "**indirect classification**."[25] An observer establishes (perhaps implicitly) a relation between the domain in question and some other domain, and then describes particular phenomena in the first domain only with reference to some corresponding phenomena in the second.

An obvious case, important to our present subject matter,

[24] [11] Some readers will object that computer science analyses treat computational processes only in terms of surface behavior—input/output relations without positing any internal structure at all, let alone impressions. But this is not so clear, not only because I have defined impression in a rather general way, but also because this view assumes a purely "extensional" reading of the semantical analyses themselves. As has been argued by Fodor and others in the mental case, some sort of representational ingredients will often be posited by theory merely in order to state the proper behavioral regularities. The abstract data types of denotational analysis can be viewed purely as theoretic entities, without classificatory import, but an argument would have to be made that they do not represent impression structure; the mere fact that they re not *claimed* to do so is not sufficient.

[25] Barwise and Perry (1983).

is the folk classification of people's thoughts and beliefs: we describe what a person P believes by describing the situation *that would be the case if what P believes were true*. When you ask me to describe my thought, there is a perspective from which I am literally incapable of answering, since in English we have neither vocabulary nor intuitions about the direct structure of thoughts—i.e., about what is inside our minds, which is where most people would say thoughts lie. Rather, I am liable to say something like the following: "I was thinking *that Palo Alto is too far from Finland*." That is, I describe my thought or thought process indirectly, by adverting to a fact (Palo Alto's being too far from Finland) that would be the case if my thought were true. The examples we looked at in discussing model-theoretic semantics were just like this: the general practice is to establish an association between something and something else, and then to get at the something else by referring to the something. So for example we set up a correspondence between Turing machine states and quadruples, which lets us describe a particular state by referring to a particular quadruple.

These examples illustrate an important general property of all indirect classification: what is *specific* about a given entity in the primary domain is set in correspondence with what is *specific* about the corresponding entity in the classificatory domain. A35 Thus a theoretical computer scientist need not encode, in the domain of quadruples, the fact that Turing machines have tapes, or that the third element of the quadruple corresponds to the mark under the read/write head, or that the numbers 0 and 1 are used to classify a mark or a blank, or *anything else that is true for all the relevant cases*. All that is required is that a *particular* quadruple contain enough information to determine what *particular* state, transition, etc., it is being used to classify.

What distinguishes the denotational approach to programming language semantics from arbitrary indirect classification, and leads to potential confusion, is the practice of *identifying* the classificatory entity with what is thereby classified. Such identification is not necessary; one could classify Turing machine #23 with quadruple #1437 without going on to claim that Turing machine #23 *is* quadruple #1437 (or even, more strongly, that to *be* a Turing machine is to *be* a quadruple—which of course is false). The identification is considered to be acceptable when the two structures are thought isomorphic, but *isomorphism is always relative to an assumed metric of equivalence*. In the computational cases we are concerned with, where a second semantical factor (functional role) lurks in the background in need of explanation, we cannot afford to identify, for one purpose, two things that may differ in respects that matter for other purposes. In particular, two structures that look to be isomorphic from the point of view of representational import may differ, crucially, in terms of functional role. For example, as we have already pointed out, no abstract mathematical structure is even a candidate for the kind of causal efficacy we will need in order to connect impressions with action. Distinct but isomorphic mathematical structures may be used to classify embodied mechanisms with very different causal pow- **A36** ers. So we need to proceed extremely cautiously.

We will encounter further issues about modeling in the next section, but for now let me return to programming languages.

In spite of its being contrary to the dominant view in AI and cognitive science,[†] in what follows I will informally adopt the *specificational* view of programs. It provides the most freedom; it is least biased with respect to impression structure; and it **A37** is most compatible with current computational theory. I will therefore assume: (i) that programs are inert linguistic entities, built up of expressions; (ii) that, in contrast, processes are ac-

[†] And contrary to my practice throughout the papers in Part B; cf. annotation A37.

tive, manifest behavior, composed in part of causally-effective impressions; and (iii) where the specificational perspective takes hold, that denotational semantics in computer science is an analysis of the program-process relation that *indirectly classifies computational processes in terms of abstract mathematical models*. The situation is pictured in figure 7.

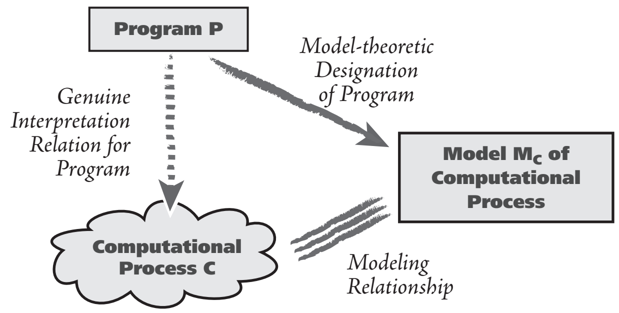

Program P
Model-theoretic Designation of Program
Genuine Interpretation Relation for Program
Model M_C of Computational Process
Computational Process C
Modeling Relationship

Figure 7 — The Model-theoretic Analysis of the Semantics of Program (i.e., Semantics as Analyzed in Computer Science)[†]

In terms of this picture, I can now explain the theoretical distraction I alluded to earlier, in introducing this section. It arises from the combination of two problems: (i) failing to distinguish between the specificational and ingrediential views of programs; and (ii) being seduced by model-theoretic properties of the model (specifically, its abstract, mathematical character) into thinking it must model content. As a result, one is apt to identify the model $\mathcal{M}_c$ A38 of the computational process $\mathcal{C}$ with the model $\mathcal{M}_w$ of the state of affairs $\mathcal{W}$ that the process is genuinely about—as shown in figure 8 (next page).

The fact that the programming language tradition calls its analyses *semantical*, in other words, coupled with the fact that it tends to use abstract domains for purposes of indirect classification, is liable to mislead AI researchers into thinking that the semantic domains of programming languages model the content of the computational processes that the programs en-

[†] Ignoring the semantic relation between process C and task domain.

gender. But this is false, at least in general. *There is simply no assumption, in the standard semantical analysis of programming languages, that computational processes are themselves semantic or intentional entities at all.* That is, no *further* semantic relation is admitted or described. All that is explained is the relation between program and engendered computational process, treated structurally.

In the AI case, however, and particularly when dealing with knowledge representation systems, we assume that the ingredients inside the processes we are interested in, which we are calling impressions, are themselves intentional (this was the essence of adopting a representational, as opposed to a merely functional, stance in section 2). Even if we were to adopt a

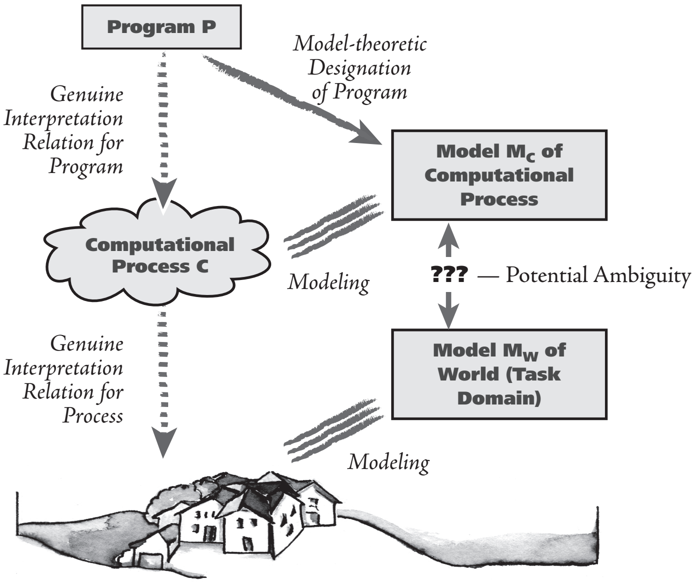

Program P
Model-theoretic Designation of Program
Genuine Interpretation Relation for Program
Model M_C of Computational Process
Computational Process C
Modeling
??? — Potential Ambiguity
Genuine Interpretation Relation for Process
Model M_W of World (Task Domain)
Modeling

Figure 8 — Models vs. Interpretations of Computational Processes

model-theoretic approach in our semantical task, therefore, we would be interested in the relation between impressions in $\mathcal{C}$ (or in the model $\mathcal{M}_c$) and the model $\mathcal{M}_w$.

I have already said that there is no a priori reason to assume that these two models $\mathcal{M}_c$ and $\mathcal{M}_w$ will be the same. But a stronger thing can be said: if one assumes that $\mathcal{M}_c$ is an adequate model of process $\mathcal{C}$, and that $\mathcal{M}_w$ is an adequate model of what $\mathcal{C}$ is about, then:

> *To identify $\mathcal{M}_c$ and $\mathcal{M}_w$ is to assume that the representational import relation of knowledge representation systems is one of isomorphism.*

Far from treating impressions as a *language*, this would be to
treat them as a *simulacrum of the world*. Or to put the same A39
point another way, to adopt, as a model of a knowledge representation system's semantics, a denotational analysis of the programming language used to specify it, is either: (i) to assume that the primary representation relation, between process and world, is one of isomorphism, or else (ii) even worse, to ignore that relation completely (thereby maintaining a solipsistic or narrow functional-role stance towards computations themselves). Either result is unhappy: simultaneously false and terrifically misleading.

It helps to look at some examples.

In purely mathematical cases, as mentioned, $\mathcal{M}_c$ and $\mathcal{M}_w$ may truly coincide. For example, suppose we write a program to calculate the factorial function. We may presume this literally means the following: that we write a program to specify a process that is about numbers and the factorial relation. In this case $\mathcal{W}$ is a structured domain of numbers and functions. Moreover, a denotational semanticist in computer science would almost surely use the same structures (numbers
and the factorial function) as an abstract mathematical model A40

($\mathcal{M}_c$) in terms of which to classify the process. Not only can $\mathcal{M}_c$ and $\mathcal{M}_w$ be identified, in other words; in this situation $\mathcal{M}_c$, $\mathcal{M}_w$, and $\mathcal{W}$ are identical.

This identity, however, relies on special properties of the example. Suppose, in contrast, that in designing a robot to pull off bank heists, we represent (in FKRL) the fact that anything to the *right* of the robot is neither to the left of it nor straight A41 in front. In order to motivate an appropriate $\mathcal{M}_c$, we need to understand the relation between FKRL programs (now viewed as specifications) and FKRL impressions. So imagine the notation for FKRL programs is reminiscent of logical notation, and that we could "write down" something like the following in FKRL:

$$\forall x \ [\ \mathrm{RIGHT}(x) \Rightarrow (\neg\mathrm{LEFT}(x) \wedge \neg\mathrm{FRONT}(\mathrm{x}))\ ]$$

A42

Suppose, furthermore, that this FKRL *expression* is more specific than the *impression* that it will generate in two ways. First, there is to be no fact of the matter, in the resulting impression, about what particular variable was used in the program; the expression might equally well have used y or z. Second, although matters of lexical notation force one of the conjuncts to be first ($\neg\mathrm{LEFT}(x)$ in this case), we will assume that impressions A43 are internally realized as unordered sets. Thus the following expression would have generated an indistinguishable impression:

$$\forall w \ [\ \mathrm{RIGHT}(w) \Rightarrow (\neg\mathrm{FRONT}(w) \wedge \neg\mathrm{LEFT}(\mathrm{w}))\ ]$$

Given these assumptions, we can then take on the task of providing a semantical analysis of FKRL programs—which is to say, an analysis of the relation between the FKRL expressive A44 specifications and the resulting FKRL impressions—using the model-theoretic approach of indirect classification. It is unlikely that we would do nothing more than constrain the models of this impression to those that satisfy the logical implication, since we can presume that more fine-grained details of the impression's structure will play a functional role in licensing inference—such as the fact that the negation signs have not been pulled to the front, as they have in the semantically equivalent:

$$\neg\exists x\ [\textsc{right}(w) \land (\textsc{front}(w) \lor \textsc{left}(w))]$$

So we would probably be tempted to classify it[†] using something like a term model, with the set of all equivalent expressions (including all those expressively differing only in the names of bound variables and/or the ordering of conjuncts). It might be, however, that for some reason we would be warranted in taking a more abstract approach, we might develop our analysis in terms of an interpretation function that mapped RIGHT, FRONT, and LEFT onto three distinct unary predicates, and were therefore to classify the impression in terms of the set of all models satisfying the given implication.

To relate this to figure 8, I will use E for the quantified expression, A45 I for the engendered impression, C_1 for the first classification, C_2 for the second, and W for the impression's interpretation—as suggested in figure 9 (next page)[26] It should be obvious, first, that C_1 and C_2 are both more abstract than E, in the information-theoretic sense of being less rich. Second, C_2 is in turn more abstract than C_1, since C_2 makes fewer distinctions A46 (identifying all semantically equivalent expressions). Finally, both C_1 and C_2 have additional properties that are not properties of I itself, nor do they model, nor do they classify any properties of I, E, or W. For example C_1 and C_2 are both sets, even though none of E, I, or W is a set.

Given all of this, we are finally ready to ask the question to

[†] I.e., the impression.

[26] The impression is depicted as inside the robot's head because the real interpretation function is being understood as holding between the robot's mind and the policeman in front of it.)

which this has been leading. Is either of c_1 or c_2 a candidate for being a model of w—i.e., a candidate for serving as a model-theoretic stand-in for the representational import of i? And the answer, to bring it all home, is *no*. A47

The fundamental problem is that "being to the right of" is not a one-place relation: one thing is "to the right" of another thing, in the world, only relative to the position and orientation of the first. Thus c_2 will not do, as a model of the representational content of i, since it does not contain enough information to determine, for example, whether impression i is *true*. If we wanted to model w, then various additional circumstantial factors—including the position and orientation of the robot—would have to be brought in explicitly, since in dealing with w we need to deal with actual position in the

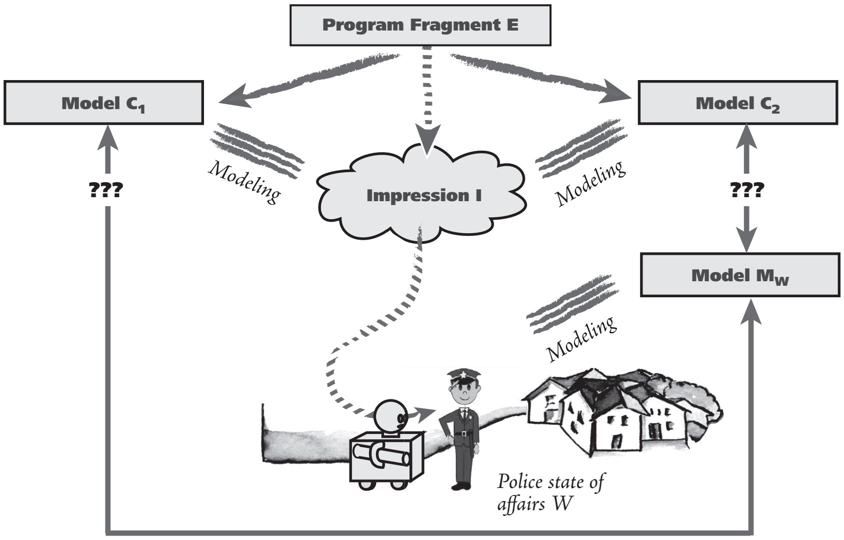


Figure 9 — Indirect Classification of the Semantics of fkrl Progams

world (that is where ones encounters police).

There is no formal problem with adding circumstantial parameters to an interpretation function, and thereby distinguishing meaning and content; we saw how to do that in section 3. Rather, the point of the exercise is to see that these circumstances *affect the semantic relation between process* (I) *and world* (W), *not the relation between program* (E) *and process* (I). In fact it is crucial, in order to get at the proper regularities in the process, that the circumstantial relativity *not* be included in the wrong place. It is not accidental that we are considering a context-dependent case, since context dependence (a virtually ubiquitous semantical phenomenon, in my view) brings into focus the absolute importance of locating all relevant semantical phenomena and relations in their proper place. It is far more likely that the robot's behavior will revolve around regularities framed in terms of what's in front of it (or to its right, or to its left), not in terms of what is in a given allocentric position. If the robot's external circumstances were mistakenly introduced in the $\text{E} \Rightarrow \text{C}_2$ relation, the resulting C_2 would fail as a model of I. For example, it would be of no help in explaining matters if I somehow broke and caused the robot always to ignore things on its left, since "on its left" would not be a notion in this modified C_2.

In general, of course, nothing prohibits a theorist's classifying something by its content (as we did in the factorial case). Exactly such a strategy, in fact, is arguably what underlies our standard (indeed, at the moment, only) way of describing the propositional attitudes constitutive of folk psychology ('*knows that*', '*believes that*', '*hopes that*', '*fears that*', etc.).[27] The point is

[27] [12] Folk psychology faces exactly the same problem we have just surveyed. In particular: (i) it classifies people's mental states by content; (ii) the purpose of these classifications is to explain how people behave and what they do; and (iii) the content of people's mental states is determined in part by their circumstances. These facts have led some writers, such as Stich (1985) to conclude that folk psychology will never be scientifically reconstructable, but in my view this seems to be an un-

only that we must not assume that *all* indirect classification is of this type. More seriously, simple indirect classification by semantical content will in general fail as a strategy for semantically analyzing the impressions of circumstantially dependependent agents. A48

6 Impressions, Expressions, and Complications

I said in section 4 that there is no generally agreed, direct way of describing impressions. So far we have seen two quite different alternatives: a metaphorical approach, using the language of linguistic expressions (§4), and an indirect approach, classifying them in terms of abstract mathematical structures (§5). Before leaving the subject, we must recognize a third.

It is common in informal AI practice, and standard in what is called *operational* semantics in the programming language community, to describe the impressions and behavior of a given computational process in terms of *the corresponding impressions or behavior of a lower-level machine on which the process is implemented*. This relation is depicted in figure 10. For example, if we were to adopt this approach to analyze the semantics of FKRL impressions, we might do so by presenting the Lisp code that has been developed to serve as the implementation of FKRL impressions.

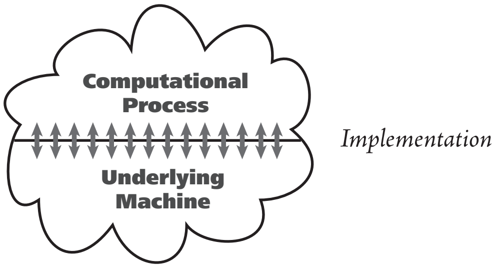


Figure 10 — The Implementation Relation

From a theoretical point of view this approach is hardly satisfying, since it just causes the problem to recur at a lower

...

warranted pessimism; the problem, rather, is to see how folk psychology compensates for the external circumstantial dependence.

level—raising questions about how to describe the implementing machine. In practice, however, it is widely accepted because it is often possible either: (i) to refer to a familiar underlying machine,[28] or (ii) to model the input/output behavior of the resulting machine in terms of ordinary mathematical functions. The relation between traditional denotational and operational semantics of programming languages, therefore, is primarily one of *abstraction*: by using coarse-grained functions as classificatory devices, the so-called "denotational" account gets at less detail than does the operational account. A49 But the fact that they are theoretically distinct ways of getting at the same phenomenon is betrayed by the fact that it is standard practice to prove the two types of account *equivalent*. In particular, they are two different theoretical approaches to analyzing the nature of the computational process *itself*; neither takes up the question to which we have been addressing, about that process's semantic import![29]

[28] [13] As usual, and as the example about Lisp code suggests, practice is in fact one level more complex than this analysis suggests. One gives the operational semantics of a programming language L, viewed specificationally, by translating expressions types of L into complex expressions types of programs, written in an implementing language L' that implements L. The language-process relation for L' is what is usually assumed.

[29] [14] There was some misunderstanding, when 3Lisp was introduced A50 (Smith (1982, 1984)) [see chs. 3 and 4] about the two semantical factors in terms of which it was analyzed and designed ('φ' and 'ψ', they were called, but they corresponded directly to first and second factors in the framework being presented here). Unfamiliar with the two-factor framework, many computer scientists assumed they were merely new names for operational and denotational accounts, respectively. This was false, but in retrospect the confusion can be attributed to three things: (i) the fact that 3Lisp was designed on an "ingredient" view of programs, whereas, as described in the text, programming language analysis is typically carried on within the specificational tradition; (ii) 3Lisp's represented "world" was constrained to being one of pure mathematical abstractions and internal structures (since it was presented as a computational model of introspection), so that the *domain* that 3Lisp impressions represented was the same one that would normally be used

For our purposes, the importance of this third approach lies in its introduction of *implementation* as yet another intentional relation for semantical analysis to contend with. As with representation and belief, implementation is a directed, asymmetric, intentional notion: to say of x that it is an implementation is to imply the existence of a y such that x is an implementation *of* y. Furthermore, the implementation boundary is opaque to other semantical relations—i.e., it cannot be A51 viewed as invisible modeling, or easily composed. For example, if we implement FKRL impressions in Prolog, and if the representational import of Prolog impressions can truthfully be given as standard first order model-theoretic semantics,[30] then it would not follow that the representational import of

. . .

for both operational and denotational semantics—i.e., the domain of impressions and of the obvious mathematical models of them; and (iii) because of this restricted domain, the interpretations of 3Lisp impressions were not dependent on external circumstances, so that the clear difference between model and interpretation, noted at the end of section 5, did not apply.

These three reasons conspired together; it has only been in the last few years that the various intricacies of their relationship have become clear to me.

[30] [15] I doubt this, for reasons that can easily be explained using terminology we have already introduced. As classically understood, standard first order logic is both declarative and syntactic, in the sense of section 2. Real-life Prolog programs, however, violate the assumed independence of factors: their role affects their import. Lacking techniques for spelling this out (i.e., techniques for providing explicit two-factor analyses), most computer scientists who give semantics for Prolog programs in fact provide model-theoretic analyses of functional role, using term models and such, in the sense explained in section 5. Logicians, expecting analyses of representational import, quite reasonably find these reconstructions odd. Furthermore, to the extent that it is functional role, not representational import, that is retained, Prolog's claim to clear *semantics* is thereby undermined.

Note that a model-theoretic analysis of functional role (first factor), on the ingredient view of programs, is liable at least partially to coincide with a mathematical model of representational content (sec-

FKRL was the representational import of Prolog. At best the *interpretation* of Prolog impressions—the elements of Prolog's semantic domain—would be FKRL *impressions themselves*.

It is almost time to summarize the various distinctions we have made, and assemble a coherent overall picture. Before doing that, however, we must tie up two loose ends.

First, in the previous section I distinguished the representational content of impressions from the entities that theorists use to classify them indirectly, identifying a *modeling* relation between the two. But I have not yet taken this observation to its obvious conclusion: modeling, like representation, specification, knowledge, implementation, etc., is itself a semantic, intentional, notion. Like many other things we have seen, a model is not a model all on its own; models are models *of* something. A balsa airplane, for example, might be a model of a real airplane no longer around, or of one being designed. Similarly, the sets of quadruples we have talked of are models of a Turing machine; the numbers 0 and 1 are often used as models of Truth and Falsity. Thus we need, ideally, to give a semantical analysis of the modeling relation, if techniques of modeling or indirect classification are ever used. I.e., in the terms of figure 9, we need semantic analyses of the $\text{C}_1 \Rightarrow \text{I}$ (or $\text{C}_2 \Rightarrow \text{I}$) and $\mathcal{M}_{\text{W}} \Rightarrow \text{W}$ relations, as well as of $\text{E} \Rightarrow \text{I}$ and $\text{I} \Rightarrow \text{W}$.

Second, all the computational processes we have looked at so far are limited in the following obvious way: we have imagined them acting in the world (driving around, computing factorial), but we have not provided them with any *communicative* abilities. They cannot talk. In order to be realistic, therefore, we should complicate our pictures yet one more time, as indicated in figure 11 (next page). In order to contain the complexity, I have omitted all models and indirect clas-

. . . ond factor) of the programs used (on the specificational model) to describe them. The subject matter is rife with such potential semantical confusions. A52 sification from the diagram, showing only the genuine intentional relations that actually obtain in a given case (i.e., have omitted additional relations posited by the theorist purely for analytic purposes). I will use the general term **notation** for the relation between expressions and impressions that they give rise to or express, and the more specific **internalization** and **externalization** to get at each direction of information flow. The analog, in the human case, is the relation between the sentences we speak and hear, and the impressions in our minds (mentalese or whatever) to which they correspond. To the extent that impressions are viewed linguistically, internalization might be analyzed as a species of translation, but it is important not to bias terminology in advance. A53

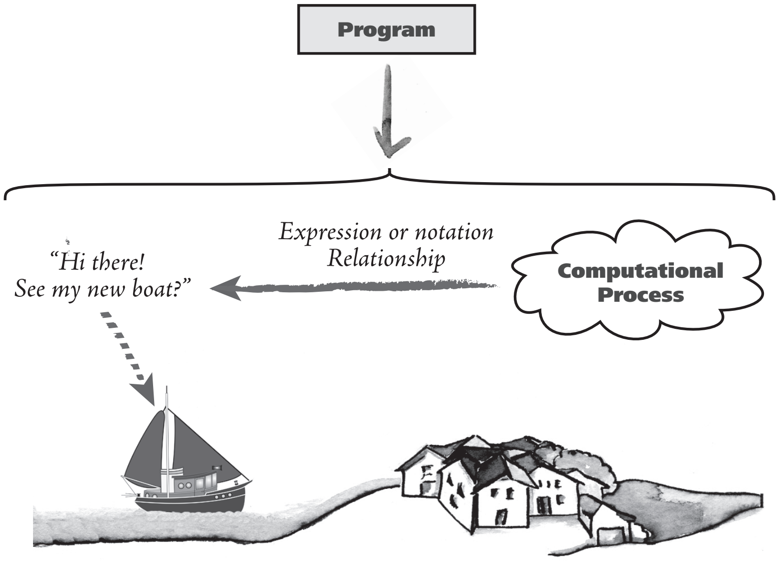

Program
Expression or notation Relationship
"Hi there! See my new boat?"
Computational Process

Figure 11 — Programs that Specify Communicating Agents

Issues of notation tie back to an issue I left unresolved above. Very often, the languages that computer systems "speak"—

query languages for data bases, editing commands for word processors, manipulation protocols for spread sheets—are visibly distinct from the programming languages used to create them. Many AI programming languages, however, such as Lisp, Smalltalk, Logo, and recent versions of Prolog, are primarily interactive, suggesting the third model of programming suggested in figure 6 (c), above.[†] Furthermore, the increasing incidence of "user-friendly" computers suggests that this interactive model of computer language will only spread. In addition, since it is the correct model for natural language, people will be biased towards an interactive stance to the extent that people understand computer languages by analogy to their native linguistic skills. Thus we have a genuinely triple ambiguity in the term 'program,' which only raises the chances of semantic confusion. Ironically, confusion between the specificational and interactive models of programming, coupled with the fact that the program ⇒ process relation is mediated by what is called an *interpreter*, has lead many computationalists to think of internalization as the fundamental semantic relation—thereby embracing exactly the view that Lewis deridingly calls "markerese semantics."[31] On the other hand, AI practice suggests what Lewis' analysis does not: that internalization is a substantial intentional relation in its own right. If nothing else, more adequate vocabulary might facilitate better interdisciplinary communication. A54

We are ready, then, to summarize four major themes in the investigation so far.

1. We distinguished *functional role* and *representational import*, and set ourselves the long-range goal of an integrated account of full significance, consisting of partially independent but coordinated accounts of each semantical factor.

[†] Page «…»

[31] Lewis (1972).

2. We claimed that since we do not yet have adequate vocabulary for talking directly about impressions, we typically avail ourselves of any one of three alternative approaches:
   a. Using *metaphorical terminology*, such as the language of linguistic expressions;
   b. Using *indirect classification*, typically in terms of abstract mathematical structures; and
   c. *Abstracting over implementations*, which makes the problem recur.

   Differences among these alternatives, and differences in the fields in which they are popular, have obscured our ability to agree on underlying impression structure itself.

3. Setting aside considerations of functional role, we identified the following important relations, each at least a candidate for its own semantic analysis:
   a. The *specification* relation, between a program and the process or impressions it engenders;
   b. *Internalization* and *externalization* relations, between expressions used by a system to communicate with its users, and the impressions they give rise to or express; A55
   c. The *implementation* relation, between impressions at one level of description, and other lower-level impressions in terms of which they are implemented; and
   d. The *primary representation* relation, between impressions (process) and the states of affairs in the world with which the agent is concerned.

   All four of these can be called *genuine*, in the sense that they are all a necessary part of the life of the representational agent in question—they have not been posited solely for purposes of theoretical analysis. Other relations between the same structures could be added, of which the most important is probably the relation between communicative expressions (language) and the world—the subject, in the human case, of natural language semantics. I will adopt these four relations, however, as primary, because they are all candidates for full two-factor accounts. Put another way, they are all in part *causal* (are causal consequences, have causal impacts in ways that pertain the their semantics), in a way that focusing solely on the direct relation between language and the world does not highlight. Note also that impressions participate in all four relations (which puts extra pressure on our ability to describe them in their own right), being the semantic domain in the first three, the so-called "syntactic" domain only in the last.

4. In addition to identifying these genuine semantical relations, we uncovered numerous relations of *modeling* or *indirect classification*, cross-cutting all of the above three. To distinguish them from the genuine relations, I will call them **metatheoretic**, since they are introduced for the purposes of us, *qua* theorists, rather than for the agent itself. Nonetheless, if we as theorists employ them, they too must be semantically understood. If we were to use model-theoretic techniques to understand the four genuine relations listed above, we would bring to eight the total number of interacting correspondence relations. The complexity can get a little daunting. It is no wonder that it is sometimes hard to tell, when presented with a "semantic analysis," just what it means.

All these results contribute to the general series of challenges I am mounting against straightforward model-theoretic semantics. The first specific challenge was implicit in the two-factor analysis itself, and its concomitant rejection of the independence of functional role and representational content. The second arose when we removed the constriction that impressions be syntactic or linguistic in nature, and embraced instead a much wider range of representational possibilities. The third stems from the multitude of genuine intentional relations just cited—*specification*, *internalization*, *implementation*, *representation*, etc.—more than one of which will require its own two-factor analysis. The fourth derives from the fact that standard theoretical techniques of indirect classification and modeling introduce, at the level of theory, a whole spectrum of additional correspondence relations, at least distractingly similar to semantic relations, if not semantic relations in their own right. If we do not understand them they will pollute our attempts to clarify the semantic relations we are primarily interested in.

Nor are we done raising challenges. In the next section I will turn to a fifth, coming to a sixth at the end of the paper.

Part II — The Correspondence Continuum

7 Semantic Soup

I said in section 3 that the model-theoretic tradition characteristically assumes a non-transitive interpretation or denotation relation, motivated by clear linguistic cases. An English A56 description of a French description of dessert, for example—such as "the four words *neige, la, à***,** and *oeufs,* in reverse order"—is a description of a *linguistic phrase*, not a description of something to eat, even if the described phrase itself denotes a custard concoction. At the same time, we saw traditional analyses freely compose modeling relations, as for example when a number encoding a description of a Turing machine is identified with the Turing machine in question. This free composition goes hand in hand with modeling's traditional invisibility.

Unfortunately, however, these two cases—non-transitive denotation, and transitive modeling—do not cover the whole spectrum of semantic relations. In the general case, intentional relations combine in much more complex ways. We will look at three examples.

First, suppose I remark on a photograph you have taken of one of my favourite sailing ships, and you then present me with a copy made by photographing the original. It would be pedantic for me to maintain, on grounds of use/mention hygiene, that the copy is not a photo of the ship, but rather a photo of a photo of a ship. For most purposes, the relation between the copy and the original print is sufficiently close that I can harmlessly compose the two correspondence relations (copy-original and original-ship), yielding a result (copy-ship) essentially identical to the second. But not for all purposes: if, on close inspection, I claim that there is a tear in the ship's sails, you might appropriately reply that *no,* the tear, rather, is in the original photograph that the copy was made from. Or I might be interested in the quality of your photographic technique, and use the copy as a representation of your original work. The appropriateness of the ability to compose, or to "look through" a copy to what is represented, can depend on the purpose to which a semantic relation is put. A57

Second, imagine connecting an FKRL system to a visual recognition system, consisting of a TV camera, special-purpose line-finding hardware, a figure-recognition module, etc. In such a case one might be tempted to say that the configuration of pulses on the cameras represented in the intensity of incoming light, and that the resulting FKRL impression represented the object under view. Yet although the former objects play a causal role in supporting the latter, it is not clear how the two representation relations fit together—the second seems to "leap completely over" the first. In spite of systematic correspondences among the constituent structures, the representation relations seem curiously independent. It is as if the structural correspondences compose, but the representation relations do not.

Third, in designing 3Lisp, I distinguished impressions called *numerals* from canonical impressions denoting them (identified as a species of *handles*), in spite of the fact that the denotation relation was an exact isomorphism. I did so because, trained in avoiding use/mention confusions, and viewing impressions as analogous to language, I thought representation relations *could not* compose. Various colleagues suggested that this strictness bordered on pedantry, and recommended that I simply identify the two impressions. Others even suggested that I identify both of them with the number designated, since as far as they could see the impression-number relation was also one of isomorphism.[32] But my allegiance to semantic strictness was strong: as shown in figure 12, I refused to say that the two-character expression written «23» (without the quote marks),[†] represented the number twenty-three; rather, when speaking carefully, I said that it *notated an impression that designated that number.* Similarly, I was forced

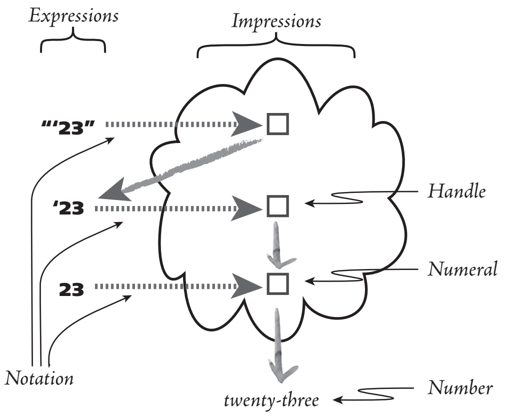


Figure 12 — 3Lisp's Plethora of Representation Relations

[32] [16] In point of fact only one factor of the full significance was an isomorphism.

[†] Cf. footnote '†' on p. «…».

to say that the three-character expression «'23» (i.e., a straight single quote mark prefacing the two-digit numeral) notated a handle impression that designated a numeral impression that designated a number. By the same token, the five-character expression «'“23”» *notated a handle that designated an expression that notated the numeral impression that designated the number.* And so on.

3Lisp was certainly semantically clean, but in retrospect some of its rigidity seems gratuitous, even if I remain opposed to any *identification* of strings with impressions, or of impressions with numbers. It is overwhelmingly convenient to be A58 able to point to a figure on a computer screen and say, simply, that it represents (or even is!) a number. More seriously, it is not obvious that one might not even be *correct* in doing so. And yet at the same time there are occasions when it is crucial to distinguish among expressions, impressions, and numbers.

All of these examples illustrate my fifth challenge to traditional model theory: *neither strict non-transitivity, nor indiscriminate identification, is always appropriate*. In each cited case, as so often happens, theoretical technique is not up to the demands of practice. The true situation is more accurately pictured in figure 13 (next page). The idea is this: a given in- A59 tentional structure—language, process, impression, model—is set in correspondence with one or more other structures, each of which is in turn set in correspondence with still others, at some point reaching (we hope) the states of affairs in the world that the original structures were genuinely about.

It is this structure that I call the **correspondence continuum**—a “semantic soup” in which to locate transitive and non-transitive linguistic relations, relations of modeling and encoding, implementation and realization, and the like.

Several points are important.

First, I will not presume, in the general case, anything about

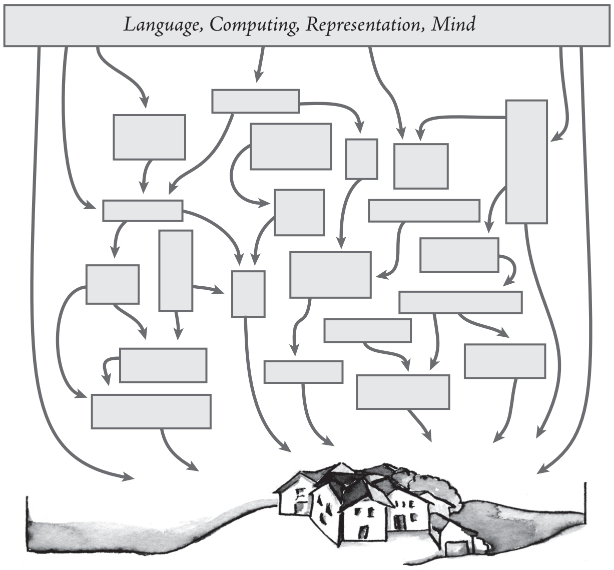


Figure 13 — The Correspondence Continuum
("Semantic Soup")

composition, relative structure, circumstantial dependence, or any other traditional issue: such questions will have to be answered individually, based on particular facts about specific cases. Sometimes, and for some purposes, these representation relations will happily compose; other times not. Sometimes *some* properties (such as ambiguity!) will be preserved even across a whole string of such correspondence relations, even though other properties (such as one-to-one correspondence of objects) are lost. In the next section I will begin to sketch out an analysis of correspondence relations that will show how this might go.

Second, one should not think of this as necessarily a single dimension; the diagram is meant to be able to accommodate the multiple dimensions of representation (notation, representation, specification, etc.). As we have just seen in 3Lisp's case, and as we saw so often in the last section, part of the task, in analyzing the semantics of computational processes, is to tie together different correspondence relations that are neither totally independent, nor arranged in a simple linear order.

The general picture given in figure 13 is intended as a replacement for the simplistic diagram of figure 2, even for the most basic intentional relations. In the remainder of the paper I will try to address a few of the numerous questions it raises. A60

Here is one, for starters. Which, if any, of these correspondence relations should be counted as genuinely semantic, intentional, representational? Surely not all. For example, to take another visual example, at the very moment I write this there is a series of correspondences of some sort between activity in my visual cortex, the signal on my optic nerve, the pattern of intensity on my retina, the structure of the light waves entering my eye, the surface shape on which the sunlight falls, and the cat sitting near me on the window-seat. And yet it is the *cat* that I see, not any of these intermediary structures. A causal analysis of perception, that is, would require a cascade of correspondences, but in this case only the full composition, *but not any of the ingredients,* would count as a genuine, or anyway "personal-level," representation (though it does not follow that these intervening structures are thereby any less important). Similarly, even if I indirectly (i.e., metatheoretically) classify impressions with functions from possible worlds to states of affairs, and then map those mathematical structures onto genuine situations in the world, the agent itself will attend only to the situations in question, entirely unaffected my abstract classifying structures.

Both of these cases, and many of the phenomena cited in the previous section, suggest that the number of important correspondence relations greatly outstrips the number that are of a genuinely semantic or intentional nature. Such arguments lead to a simple and almost flat-footedly obvious conclusion: *structural correspondence is a far more general phenomenon than representation or interpretation.*[33] First, it permeates theory, in terms of indirect classification and modeling. Second, it permeates practice, as manifested in such notions as implementation, encoding, realization, presentation, specification, internalization, and externalization, as well in as our initial concerns of representation and knowledge. Third, although not all these correspondence relations should be counted as fully intentional, there is no chance that we will understand semantics unless we are first clear on how they all fit together. So my recommendation is that we peel correspondence away from more difficult semantic issues, and make it a subject matter in its own right.[34]

Let us look, then, at what a theory of correspondence might be like, before returning to semantics and to knowledge representation.

[33] [17] This implies, of course, that there must be much more to representation than correspondence. Hence footnote 3 (page «·7»); correspondence on its own requires neither disconnection nor registration.

A61 [34] [18] Strictly speaking I do not believe this, for two reasons. First, my metaphysical predilection is to attribute the notions of object, property, and relation to a collaborative interaction between mind and world, so that the world *alone* need not be held responsible for objects' boundaries and kinds (naive realism), nor need they be viewed as pure constructs of cognition (variants of solipsism or idealism). Second, I am at least prepared seriously to entertain the hypothesis that minds, fundamentally, are embodied representational processes. In conjunction these two views raise the following "chicken and egg" problem: if minds are required in order to know how the world is structured, and if minds are representational, then representation must seemingly be studied before correspondence, in order to establish the categories in terms of which the correspondences will be articulated. On the other hand, for reasons spelled out in the text, I think the chances of getting representation right without a prior theory of correspondence are rather limited.

These considerations interact with another distinction. Which person is being held responsible for the categorisation of the domains in

8 A Sketch of a Theory of Correspondence

In broad outline, I will adopt a very simple approach to the structure of correspondence. First, I will identify two domains, presumed to consist of a predetermined collection of situations, objects, properties, and relations. Call them **domain** and **co-domain** (though this is not category theory), and say that an element of the domain **corresponds to** an element of the co-domain. Furthermore, without introducing any assumption of symmetry, I will speak most generally of correspondence *relations*, rather than functions, and make room for circumstantial parameterization in the usual way. The situation is pictured in figure 14. (The resemblance to figures 2 and 3 is obvious; we can now see those figures were right for correspondence, but wrong—because too simple—for the complex general story about semantics). A62

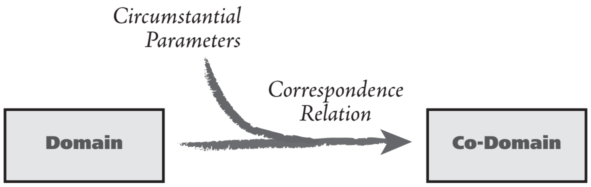


Figure 14 — The general structure of correpsondence

Given these two domains, specific correspondence relations are defined between **states of affairs** in each domain—not between the domains themselves, nor between objects, properties, or situations on their own, but between *things being a certain way* in one domain, and *things being a certain way* in the other. Thus, the light's being red corresponds (or so we hope) to the car's stopping. Similarly, we might say that the sequential concatenation of the numeral '2', the sign '+', and the numeral '3' corresponds to the addition function's being applied A63

. . .

question: the agent under study, or the theorist? I assess the interaction among these issues in Smith «forthcoming (b)»; the net result is simply the rather predictable conclusion that the two notions (correspondence and representation) must be viewed as something of an indissoluble pair. This conclusion, however, does not in any way challenge the view being expressed here: that they are not the *same*.

to the numbers two and three, which in turn corresponds to the number five.[35] Even in cases where there is a simple correspondence of objects, as when the numeral '3' stands for the number three, it is really the object's *being that and not some other numeral* that corresponds to the number's *being that and not some other number*. The numeral may have all sorts of other properties—such as consisting of one curved and one straight line—which assemblage of properties does not correspond to anything in the co-domain at all.

There are several reasons to require an explicit specification of domains, and to lay responsibility for the correspondence relation on states of affairs (rather than on objects *per se*). In general, objects exemplify infinitely many properties, and participate in infinitely many relations—in this sense the world is overwhelmingly rich. Even questions of object identity do A64 not escape this richness, as precise attempts to define numerals quickly reveal (does the expression "124+124" contain one, two, three, four, six, or eight numerals?). It is therefore nec- A65 essary, in characterizing a particular correspondence relation, to identify in advance the particular set of objects, properties, and relations in each domain that are constitutive of the significant states of affairs—what I will call a prior **registration** A66 of the domains—and then to identify, with reference to that registration, how states of affairs in the domain correspond to states of affairs in the co-domain. This is partly because states of affairs, at least as I am using the notion,[36] are individuated by the relations and properties they instantiate (a number's being the sum of two plus two, and the same number's being the positive square root of sixteen, are different states of affairs,

[35] [19] Note that this phrasing suggests iterated correspondence: expressions to function applications, and from there to values. The connection between iterated correspondence and so-called "intensional" analyses of functions and relations is discussed at the end of this section.

[36] My intention is to employ the term in a way compatible with its technical use in Situation Theory (Barwise 1986a), although nothing in the text requires that particular analysis.

on this view). But it also seems true to common sense, as the red light example suggests.[37]

(As well as adopting these two theoretical assumptions, there is another which I will explicitly set aside. Many writers, including theorists as far back as Peirce, have embraced a deep intuition that representation is a three-place, not a two-place, relation, involving not only representation and represented, but also *interpreter, observer,* or, in Peirce's case, *interpretant.* Thus a text, and probably even a simple map, is taken not to be a representation on its own, but to represent only for some other agent or purpose—or both. I sympathise in the representational case, but we are talking here about a simpler notion of correspondence, where the question is much less clear. A67 For example, one could view a binary correspondence relation between X and Y as a relation that an interpreter posits or reacts to, in taking X to represent Y. Thus your map may not represent New York unless you or some other person takes it to do so, but that act of taking it to represent New York involves attributing or establishing a *binary* correspondence relation of a certain type—of a type, furthermore, that might be characterized in terms of the theory I am proposing. In addition, given my general recognition of the importance of circumstantial dependence, it is not obvious that the role of interpreter has been excluded. But however this goes—and even if one were to argue convincingly that even correspondence should be analyzed as tripartite—my present purpose is to define a project, not to report on its conclusion. Such questions should ultimately be answered by theory, not prejudged. And I would hazard that the distinctions to be made, here, in terms of correspondence treated as binary, would carry over,

[37] [21] The theoretical stance of taking registration as prior to correspondence, and correspondence as at least partially independent from representation, is not one I am ultimately satisfied with; see footnote 34, and Smith «forthcoming (b)». It seems well motivated, though, at least as a way of getting to the next stage in semantical clarity.

though perhaps be thereby complicated, in a three-element version.)[38]

I will call the relevant states of affairs in the domain and co-domain the **source** and **target**, respectively. So the source expression “72°10’ E, 44°20’ N” might correspond to a bucolic target in northeast Vermont. In general, correspondence relations will be defined in terms of source and target **types**, or more generally to what in mathematics would be called the *signature* of the two domains, in such a way that instances of the source type would correspond to instances of the target type in some determinate fashion. For example, the mapping from sets of quadruples to Turing machines would be established so that a particular quadruple's having certain elements would A68 correspond to the controller of the corresponding Turing machine's satisfying a particular transition function (though what Turing machine that transition function was a transition function *of* might be assumed, for the whole set of quadruples, and thus not explicitly “corresponded to” by anything). This approach makes sense of the intuition about modeling suggested in §5: that what is specific or particular about one state of affairs (source) determines what is specific or particular about another (target).

In setting out an initial analysis of this sort,[39] I call a particular correspondence relation **iconic** if each object, property, and relation in the source corresponds, respectively, to some object, property, and relation in the target. In a case of iconic correspondence, that is, the abstract type (object, property, or relation) of the source would be the same as the abstract type

[38] [22] In cases where a third agent—an interpreter—is present, a possible solution is presented to the problems raised in footnotes 18 «check» and 21 «check»: the agent can register both representation and represented. But there are two problems with this. First, of course, we have to ask how agents register, which brings the problem back to roost. Second, it is a strong and possibly false claim that interpreters register signs and language they use (as opposed to mention).

[39] «See Smith (forthcoming(c))—check!»

of its target. A particularly important case of iconicity occurs when a source object, property, or relation corresponds to *itself* in the target: I will say in such a case that the target structure is **absorbed** in the source. For example, left-to-right adjacency A69 is absorbed in the grammar rule "EXP → OP(EXP,EXP)" for a simple term language for arithmetic. Similarly, to suppose that the necessity of set membership, in a model-theoretic analysis of modality, models necessity in the world is to assume, counter-factually, A70 that necessity is absorbed. In contrast, I will say that a target property or relation is **reified** if it is corresponded A71 to by an object in the source (reification is not defined on objects). Thus for example the syntax of predicate calculus reifies properties, because it represents them with (instances of) predicate letters, which at least in standard syntactical analyses are registered as objects.

A correspondence relation is called **polar** when an existentially positive source (something's being the case) corresponds to an existentially negative target (something's not being the case), or vice versa. Hotel lobbies provide an example, where a key's being present in the mail slot at the registration desk indicates A72 the fact that the client is gone. A relation is called **typological** if it can be defined without reference to distinguished individual objects in the domain or co-domain. Thus the standard Cartesian relation of ordered pairs of real numbers to points on a plane fails to be typological on four counts: *origin*, *orientation of x-axis*, *unit length*, and something to distinguish *left and right orientation*, such as a distinguished normal to the x-y plane. Finally, when either or both domains are analyzed mereologically—in terms of notions of part and whole—either or both ends of the correspondence can be defined **compositionally**, in the sense that what corresponds to (or is corresponded to by) a whole is systematically constituted out of what corresponds to (or, again, is corresponded to by) its parts. A73 If the part/whole relation is itself absorbed, a very strong version of compositional correspondence obtains, where parts of a source correspond to parts of that source's target.

Many other such relations can be defined, ranging from this simple sort up through more complex cases having to do with sentences, quantification, use, circumstantial dependence, etc. The intent here is not solely to develop a theoretical typology (though that is often useful, especially early in theoretical development), but eventually to identify an algebraic basis of correspondence in terms of which to analyze arbitrary relations. Given such an algebra, for example, and an analysis of two relations c_1 and c_2 in terms of the orthogonal set of basic features, it should be possible to predict the exact structure of the composed relation $c_1 \circ c_2$. Thus we would expect the composition of two iconic relations to be iconic, iconic relations to be both left and right identities (with respect to this algebra), A74
and so on and so forth. Note, however, that the appropriateness conditions for composition are very strong: $c_1 \circ c_2$ makes sense only if the targets of c_1 are of *exactly the same type* as the sources of c_2. Traditional isomorphism will not do, since isomorphism is just another correspondence relation c_3; the combination would have to be analyzed as $c_1 \circ c_3 \circ c_2$. (Conditions on the theory of correspondence, that is, are so strong that isomorphism may not be absorbed.)

As the isomorphism example suggests, a correspondence theory of this sort would provide theorists (I primarily have semanticists and computer scientists in mind, but of course the account would be general) with an extraordinarily fine-grained pair of glasses with which to analyze arbitrary structured relationship between domains. Every conceivable coding, representation, modeling, implementation, and isomorphism relation would be made blatantly visible. Whereas category A75
theory can be viewed as highly abstract, in other words, correspondence theory would be exactly the opposite: *unremittingly concrete*. This does not mean that abstract objects could

not be studied within such a framework, of course; only that *no further abstraction by the theory* would be permitted unless explicitly accounted for (beyond that provided by the initial registration of the domains). Thus, whereas a model-theoretic A76 analysis of the interpretation of the English word *cat* might map it onto a mathematical set, a correspondence-theory based semantic account could not do so (or if it did, it would be wrong). There is no problem in providing a correspondence-theoretic analysis of the relation between the word *cat* and the set-theoretic structure used by model theory to *classify* it, but that, as the correspondence theory would make explicit, is quite a different thing.

It is a consequence of this fine granularity that many standard mathematical techniques, such as that of identifying structures "up to isomorphism," would be inapplicable. But this result is to be expected: since the whole point is to avoid gratuitous modeling, and to explain arbitrarily fine-grained distinctions, the theory cannot indulge in any loss of detail.

As well as focusing on the detailed structure of specific correspondences between states of affairs, an adequate theory would have to address general questions about particular relations, such as whether every source in the domain corresponds to exactly one target, whether every target has a source corresponding to it, etc. It would be natural, that is, to define correspondence-theoretic versions of such standard notions as totality, completeness, and ambiguity. But this starts to feel a little odd, because of its familiarity. Are we just reinventing traditional mathematical accounts of functions and relations? How do our categories of correspondence relate to such standard notions as isomorphism, homomorphism, injection?

The answer appears to be the following. It has often been pointed out that standard so-called *extensional* analyses of functions and relations, in terms of piece-wise pairings, officially ignore the structure of the connection between the domain and co-domain, even though that structure is often A77 important in practice—such as when the function is to be *computed*, or the relation *recognized*, or when the connection is *causal*, defined in terms of the constituent properties. Extensional mathematical analyses abstract away from such concerns. When, epistemologically, we describe functions in natural or formal languages, or embody them in machines, we typically betray a great deal of additional information. Thus the standard term designating the factorial function

```
if n=0 then 1 else n–factorial(n–l)
```

implicitly suggests a way of *computing* factorial, even though that information is lost in the standard extensional analysis, which would merely map the foregoing expression onto the infinite set of ordered pairs {<0,1>, <1,1>, <2,2>, <3,6>, …}.

In the general case the information conveyed by a functional description can be sorted into three kinds, as suggested in figure 15: information about (i) the structure of the domain, (ii) the structure of the co-domain, and (iii) the structure of the relation between the two (the first two clearly merge when, as is often the case for simple functions, the domain and co-domain are the same).

Recognizing the importance of this other information, various people have attempted to develop what are called *intensional* analyses of functions, relations, etc. The idea, or so it is claimed, is to make this extra information explicit. But from our point of view there is something curious about the way in which this is traditionally conceived. Because these efforts have arisen in the context of computation, recursive function theory, and a general concern with procedures, the approach is in fact not one of making these three kinds of information explicit, but rather of making explicit *the structure of an algorithm for computing the function* (or relation). Thus Moschovakis[†] has proposed treating an algorithm as a first class mathematical entity in its own right, and a variety of writers have at least argued for dealing directly with procedures, such as those recommending procedural treatments of semantics.[40]

Needless to say, there is nothing wrong with explicating the notion of an algorithm. But there is no reason to suppose that, even if successful, this project will make explicit the three kinds of information cited above. For example, no matter how explicit I am in giving you directions for driving across Boston, the structure of the city will at best be borne implicitly in the resulting descriptions of routes. Imagine trying to reconstruct a Boston city map by sorting through every route traveled by a long-time cab driver, gradually culling information about the town from such sequences as "Drive two blocks up Trapelo Rd, turn right on Grove," or heroic attempts explain how to get from Jamaica Plain to Logan airport without using a tunnel. Making the *algorithm* specific will not even make explicit the structure of the relation it computes, let alone the structure of the related domains.

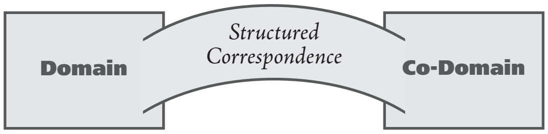


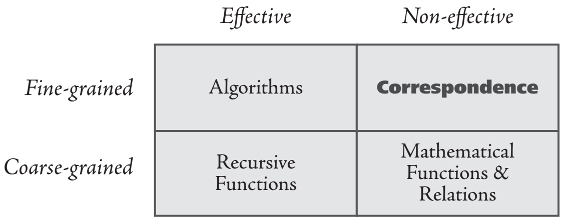
Figure 15 — The three structures of correpsondence

In contrast, a correspondence theory can be viewed as almost a dual project: it would provide an informationally rich account of the structure of the relation between structured domains, though it would remain silent (unless that project were explicitly taken up) on any question of *computing* this relation. It would focus *directly* on the three relevant structures (of domain, co-domain, and correspondence), rather than taking them to be indirectly manifested by specific ways of going from a given domain element to its corresponding co-domain element. A78

† Moschovakis (1984).

[40] E.g., see Woods (1981).

As for which project has a better claim on being an "intensional" analysis of functions and relations, I cannot say. Nor, presumably, does it matter. For one thing, the very theory of correspondence I am proposing will among other things obviate the worth of such terms as "intensional" and "extensional." A79 More important is to recognize the essential difference, and compatibility, between the two accounts. As suggested in figure 16, the distinction between fine-grained ("intensional") and coarse-grained ("extensional", or piece-wise) analyses is orthogonal to the question of effectiveness or computation. We can thus classify the standard set-theoretic model of functions and relations as *coarse-grained and non-effective*, recursive theory as *coarse-grained but effective*, and the theory of algorithms as *fine-grained and effective*. A theory of correspondence then occupies its rightful place as the fourth possibility: a *fine-grained but non-effective theory of relationship*.



	Effective	*Non-effective*
Fine-grained	Algorithms	**Correspondence**
Coarse-grained	Recursive Functions	Mathematical Functions & Relations

Figure 16 — Analyses of relationship

The location of a correspondence theory in this diagram is well suited to the semantic purposes for which it was designed. One of the most fundamental facts about most genuine semantic relations, such as reference, is that *they are not computed*, in any coherent A80 sense of that word. When I say "Bach died in 1750," and thereby refer to a long-dead composer, nothing *happens* in order to make the reference work; it just *is*. It is thus entirely to be expected that semantical examples should push us towards a fine-grained but non-computational analysis of structured correspondence.

9 Semantics Revisited

The availability of a correspondence theory would change semantical analysis in at least these ways:

1. A81 As promised, the following traditional notions would be replaced: (i) a strict hierarchy of (meta-)languages; (ii) invisible but promiscuous modeling; and (iii) the notion of an absolute use/mention distinction.
2. It would provide the theorist with sufficient equipment to analyze such otherwise unanalyzed notions as *encoding*, and to discern and thereby avoid problems of gratuitous artifacts. A82
3. It should provide, for the first time, adequate vocabulary in terms of which to analyze and assess such non-linguistic representational structures as images and analogue representations.
4. It would enable us to explain some lurking problems and unexplained worries that have plagued traditional approaches.

I will look at each of these briefly.

First, dismantling an absolute use/mention distinction does not mean licensing automatic composition of all correspondence relations. On the contrary, the intent of the algebraic basis of correspondence sketched in §8 is to enable us to see what sorts of properties will propagate through iterated correspondences, and which ones will not. The popular closed-world assumption in AI, for example, is in essence an assumption that *object identity is absorbed*; in any given application it should be straightforward to verify whether this property is preserved across one or more correspondence relations in question. Similarly, the assumption that *words* A83 (not just singular thoughts) have referents could be justified, even by someone committed to the logical priority of mental impressions, just in case the internalization and representation relations could be unproblematically composed. Even in written natural language, use vs. mention apparently shades off into matters of degree; thus we have (in something like increasing degrees of "semantic withdrawal"):[41, 42]

1. Margaritaville is lively. A84
2. Margaritaville is so-called for dubious reasons.
3. They call it *Margaritaville*.
4. When I asked where they lived, they said "Margaritaville."
5. "Margaritaville" is a fictional name.
6. I am sorry to have to tell you, but in this case "Margari-taville" is hyphenated;
7. "Margu[illegible]ville" is smudged.

Particular analyses of use and mention would depend on the semantic relations employed; once again letting go of the strict theoretic distinction paves the way for accommodating a wealth of familiar facts.

As well as undermining use/mention distinctions, the correspondence continuum challenges the clear difference between "syntactic" and "semantic" analyses of representational formalisms—an especially important consequence given the allegiance commanded by this historically entrenched distinction. On the face of it, it might seem that we are simply removing an important method of discriminating accounts, which would be a negative result. The claim, though, is that no *simple* "syntactic"/"semantic" distinction gets at a natural joints in the

[41] Acceptance of the last two seems to vary, among people I have informally surveyed.

[42] [23] Introspection suggests that quotation marks are primarily, if not always, used to refer to linguistic *types*. As a possible counterexample, Geoffrey Nunberg has suggested: "'Fiat lux' started this whole mess," but at best that refers to an utterance of the Latin sentence different from the (enclosed) token used here to refer to it—and that they *are* tokens of the same type is crucial to the sentence's success. There does seem to be merit to the view that quoted expressions cannot be used to refer to their constituting internal *tokens*.

underlying subject matter, no matter how profound the ultimate difference, as it were, between map and territory.

For example, many writers have claimed to provide semantical analyses using models set-theoretically constructed out of basic syntactic elements such as sentences, ground terms, etc. (i.e., so-called “term models”). A typical AI case is found A85 in Moore and Hendrix’s proposal for a semantical model for belief;[43] similarly, term models are often used in giving semantical analyses of logic-based programming languages, such as in Goguen and Meseguer’s EQLOG.[44] Although stamped with the official “semantics” insignia, they are often used as abstract models of (i.e., to classify) syntactic or computational properties, such as inter-reducibility of terms in a rewrite system (α-interconvertibility in the λ-calculus, for example), effective derivability, etc.

My point is not to indict this practice, nor to dispute its theoretical importance. Rather, the point is this: *if* one is committed to a simple binary “syntactic”/”semantic” distinction, as on the traditional view, then such proposals would have to be counted as syntactic, and hence as false advertizing—since for example the semantical interpretation of a formula such as DEAF(BEETHOVEN) would have only to do with syntax, nothing to do with the composer himself. On the more complex view I am proposing, needless to say, room is provided for such analyses as these. Whether they are labeled ‘semantical’ becomes an issue, of perhaps debatable importance—but the main point is that the theorist would need to make plain exactly what kinds of relations are being analyzed, what kinds of facts or properties or states of affairs (e.g., in models) are being used to classify what others; what relations in the overall picture are computational, representational, whatever. The crucial points are just two: (i) the space of possibilities is not constricted in advance, by the nature of the theoretical frame-

[43] Moore and Hendrix (1979).

[44] Goguen and Meseguer’s (1984).

work; and (ii) a substantial (and presumably intellectually hygienic) premium would be put on stringent honesty about what is being claimed to be what.

The second main consequence of the new approach arises from its fine-grainedness, which thereby facilitates direct views onto otherwise invisible relations. These last fall into two kinds: (i) subject-matter relations that have heretofore evaded satisfactory analysis, such as encoding and implementation; and (ii) theoretic relations such as modeling, which have affected and sometimes distracted analysis. With respect to this fine-grainedness of approach, correspondence theory can be understood, in its relation to traditional semantics and model theory, as analogous to the relation between situation theory[45] and traditional set theory. In both cases, the classical system makes far fewer distinctions than at least some analyses demand. Thus situation theory, like other property theories, populates the world with properties, relations, facts, states of affairs, and the like, thereby embracing a much richer ontological foundation than the set theory we are used to. My brief against traditional model-theoretic analyses of languages and modeling is similar to Barwise and Perry's against set theory: not only that it glosses much of the very detail we need to understand, but more seriously that the nub of the phenomenon inheres in that glossed-over detail. Moreover, the enterprises of situation theory and correspondence theory are related in much stronger ways than by analogy. Any candidate A86
correspondence theory will have to be based on a much richer ontological foundation than is espoused in set theory, for at least the following reason: in virtue of its explicit rejection of invisible modeling, correspondence theory will have to be able, in its own right, to cope directly with full registrations of domain and co-domain.

For example, suppose someone wanted to use the proposed correspondence theory to assess the familiar Cartesian

[45] Barwise (1986a).

representation relation between pairs of real numbers and points on a plane. In the model-theoretic tradition, the first job would be to develop models of both phenomena. However, since ordered pairs are an eminently good model both of themselves and of points, the representation relation would look to be one of identity. For a correspondence theory to see the relation, it would have to license both ordered pairs of real numbers and points on a plane as legitimate, distinct, entities—as *first class citizens*, to use the computational phrase. Thus a set-theoretic base would simply not work. A87

Given an adequate ontological foundation, however, and a concomitant account of correspondence, one should be able to repair some well-recognized deficits in current computer theorizing, all of the "too coarse-grained" variety. The broad metric of Turing equivalence (relied on to demonstrate the "equivalence" of various models of computing) is a particularly blatant example—since virtually every imagined computer language, modulo standard idealizations of indefinite memory and time, turns out to be of equivalent power. The problem is A88
that the *very notion of Turing equivalence itself* rests on promiscuous modeling; in showing one machine equivalent to another, one does not *really* show them to be the same; rather, what is shown is that one can *implement* one in the other. More seriously, all sorts of rather close correspondence relations—implementation, encoding, modeling, etc.—have similarly fallen between the cracks of theoretical assessment, being "closer," so to speak, than is typical of the representational import of language, but still distinct from identity. The hope is that a proper categorisation of correspondence will be a first step towards more adequate foundations and more subtle comparisons.

The third semantical consequence has to do with the potential integration and unified treatment of a wide variety of apparently disparate kinds of representation. Ever since the earliest days of Artificial Intelligence debates have raged about the relative merits and properties of so-called *analogue*, *pictorial*, and/or *imagistic* representations, vis. a vis those that are *sentential*, *propositional* or, as Sloman calls them, "Fregean."[46] Maps and diagrams are paradigmatic examples of the former; natural language sentences and formulae in first-order logic, of the latter. In spite of a diverse literature probing these distinctions and explicating cross-cutting distinctions buried in them, however,[47] no comprehensive framework has emerged in which to reconstruct the underlying insights. It is difficult not to notice that writers on these topics often refer back to Wittgenstein and Peirce, who wrestled with these issues before the development of modern semantical technique.

This literature conveys an unmistakable picture of complexity inherent even in the most paradigmatic examples. For example, Sloman attempts to differentiate *analogic* and *Fregean* representation by supposing that the former manifests a certain kind of correspondence (he neither explains nor constrains it) between the *part* structures of representation and represented.[48] On the face of it, this would seem to amount to a structural correspondence between relations, of the sort we saw in discussing iconicity, coupled with a mereological registration of both source and target domains. The pure characterization, in other words, seems exactly the sort that a correspondence theory should be able to explicate. Sloman's proposal, however, seems much less successful as a way of clearly discriminating between analogue and propositional representation. For example, as many have pointed out,[49] it does not have the intended bite unless one ties down the notion of "part." For a bar chart to remain analogue, the conception of part in the target domain must be taken quite liber-

[46] Sloman (1975).

[47] [24] A representative series of articles by Dennett, Fodor, Kosslyn & Pomerantz, Pylyshyn, and Rey can be found in Part Two (Imagery) of Block's (1981). See also Sloman (Pylyshyn (1984); Sloman, (1975) and Pylyshyn (1984 chapters 7 & 8).

[48] Sloman, op. cit.

ally; on the other hand, such sentences as "Aaron, Adrian, and Amelia arrived in that order" seem to employ part relations in source (sentence) structure to signify part relations in the target (what is described). So the distinction is not so clear. Furthermore, there is no doubt that even paradigmatic analogue representations or images represent only with respect to a correspondence relation,[50] so the constraint on mereological correspondence would need to be spelled out, in exactly the way that the proposed algebra of correspondence types suggests.

Annotation on Haugeland's "representation Genera"

Without delving into specific examples, several general things seem clear. For one thing, the persistent intuition that representations come in a wide variety of kinds seems exactly right. For another, analyzing these kinds will require exactly the sort of fine-grained correspondence theory we are proposing. Finally, it is unlikely that common examples will sort into any small, mutually exclusive, set of nameable classes. Instead, we should license a full range of types of correspondence, kinds of circumstantial dependence, and varieties of registration (continuous, discrete, compositional), in terms of which subsequently to characterize pictures, maps, graphs, schedules, models, images, and so forth, as well as sentences, formulae, and elements of language. The latter group, one would guess, will in general be more *complex* than the former, and may involve additional kinds of circumstantial dependence, compositional structure, or relational complexities such as polarity. But they surely will not be totally distinct.

In section 7 I introduced the phrase "correspondence continuum" to connote the interacting complex of different correspondence relations we often find connecting representation and represented. However, I equally intended the words to suggest the different kind of continuity arising here: of a full range of variation of type of representational structure.

A simple example will illustrate how continuous these types

[49] See for example the discussion in Pylyshyn (1978).

[50] See for example Fodor (1975).

can be. Contemporary architectural blueprints used in building construction contain what, to the uninitiated, can be a bewildering range of symbols, ranging from obviously analogue outlines of room shapes, through suggestive icons indicating plumbing and kitchen fixtures, heaters, etc., through slightly stylized icons for electrical outlets, light switches, etc. (with a number of slashes to indicate number of individual outlets, an 'S' to mark whether they are switched, etc.), through general purpose furniture icons with simple inscribed names ('desk,' 'bed,' etc.), through icons with manufacturer's annotations ("Vermont Castings," "Wolf," and so on), through intermixed sketches, diagrams, and annotations on construction technique, all permeated with arrows, English comments, stamps of approval, scribblings to cancel out parts of the specification, and so on and so forth. That there is a rich variety of representation seems without doubt; that a theoretical scalpel could carve the assemblage into a few neat categories, extraordinarily unlikely.

The moral is unchanged: in variety, detail, and forms of correspondence, current representational practice vastly outstrips current semantical technique. Recognizing that our current theoretical apparatus was developed primarily in service of very particular representational systems employed for logic and meta-mathematics, we should instead embrace what Ken Olson has suggested:[51] a return to as various and thick a structure of correspondence relations as Peirce ever imagined. Unlike Peirce, however, we can avail ourselves of the full battery of rigorous mathematical methods, axiomatic systems, and so forth, that have been developed since his time. Given such a project, we might even be able to rescue some of the richness of the "semiotic" tradition from what has been perceived to be its vagueness and descriptive complexity.

The fourth and final consequence listed at the beginning of this section has to do with lurking problems in the traditional

[51] «Ref Olson—phd at Stanford?»

approach. Those problems, however, arise from fundamental metaphysical questions, and will as such be addressed in the next section.

Part III — Metaphysics

10 Theories, Models, and Metaphysics

Figure 13 painted a continuum of relations, starting on the left with the linguistic or representational structure under analysis, and progressing in some fashion towards the "real world" on the right. I have suggested that a correspondence theory would provide us with an ability to characterize the relations among the structures comprising this whole, but I have not addressed the question of how one would locate oneself in the resulting continuum. If, as I have suggested, the practice of calling certain relations "syntactic" and others "semantic" is not helpful, is there any other way to distinguish one analysis from another? Or, to put the same question the other way around, can we say anything about traditional approaches? How are they located on this as-yet rather unstructured map?

Four things can be said.

First, if the picture I have been developing is even roughly correct, it predicts that we will encounter structures at various stages across the continuum—relatively more "linguistic" or "syntactic" ones, closer to the primary representational source at the top, others midway across, perhaps having to do with meaning or other semantic (or efficient) uniformities, and others relatively more directly metaphysical or ontological, closer to the full buzzing confusion at the bottom. That the distinction becomes a matter of degree, rather than a binary decision, makes sense of various traditional debates and disagreements. In particular, it is somewhat of a theoretical relief.

To be specific, many people (I am one) have worried about the metaphysical foundations of particular model-theoretic analyses of language,[52] feeling that the proposed model structures reflect, at least in part, the structure of language, not the structure of the world the language is about. For example, consider an analysis (such as a term model) that posits distinct one, two, and three-place relations for various different uses of the verb 'break' (as in "The window broke," "The hockey puck broke the window," and "I broke the window with a hockey puck"). Or imagine an analysis that distinguishes the Pope's saying Mass from the *fact* of the Pope's saying Mass. Or imagine (not hard!) debates about the metaphysical reality of possible worlds, with some people saying that they are real, others saying that they are merely theoretical devices with which to classify language, others claiming that arguments about the reality of semantical constructs miss the point, which is after all to prove various mathematical facts about the linguistic structures themselves. Or suppose someone were to doubt, on metaphysical grounds, the received wisdom that positive and negative facts are on a par, believing instead that this symmetry is a device of language, not a fixture in the world. A89

If one were to adopt the traditional binary view, then all such questions must be settled one way or the other. I.e., you would have to reject an otherwise appealing semantical analysis if the semantical structures it proposed were metaphysically unconvincing. On the sort of view I am suggesting, however, the whole continuum of possibilities is *exactly what one would expect*. You could accept a term model (leaving aside whether to call it 'semantics'), for example, but understand it as living rather close to the left hand side, and then ask for further relations to anchor it in, or relate it to, states of affairs further to the right. The structure of the continuum, that is,

[52] [25] The difficulties are blatant in term models, evident in Kripke style possible world structures, but still apparent, at least to my mind, in the structure of the situation-theoretic universe (Barwise (1986a)).

gives you a way of accepting your fellow theorists' intellectual contributions, even while disagreeing with their metaphysical predilections.

Second, there are several ways one might locate a particular correspondent structure in a given semantical analysis. For example, it was pointed out early on that much of the semantical contribution of linguistic use arises from circumstances of utterance, not directly from the structure of the sentence used (as in the "I'm right; you're wrong!" example). One of Barwise and Perry's chief points about language[53] is that this property, which they call *efficiency,* is necessary to the proper functioning of communication. It is natural, then, to imagine an analysis of language use that spelled out this circumstantial dependence. It is also easy to imagine, as a semanticist, wanting to avoid the recalcitrant metaphysical problems that arise when you try to map specific vocabulary items onto the world itself (see below). So the following approach might suggest itself: develop a correspondent structure midway between utterances and the world, in such a way that the entire circumstantial dependence of language, up to questions about the metaphysical foundations of vocabulary, has been discharged. The resulting structure is liable to be infinite, but of course that is not a theoretical problem.[54]

This seems a productive way to understand the semantical structures posited both by possible world semantics and situation theory. Needless to say, there are important differences between the two proposals, some of which we can describe: possible world semantics *models* what it calls the interpretation of sentences, whereas situation theory (at least in recent variants) tries to deal with interpretation directly. But the

[53] «Ref»

[54] [26] John Etchemendy [personal communication] once suggested that the situation-theoretic universe itself could be viewed in this way (the world of situations, types, states of affairs, etc.—not the language or notation used to describe it): as the world's only non-situated language.

point is to reject as too simplistic the question of whether the structures they each propose are to be viewed as: (i) *the structure of the world*, albeit highly idealized; or (ii) *the structure of language*, albeit decontextualized. Instead, they can both be A90 understood as intermediate analyses.

Third, it is important to dispel a false assumption about how correspondence relations will go, as we move from left to right. As many writers have noted, far more distinctions are made in the syntax of most formal languages than in the model-theoretic structures posited as their interpretations. The most extreme example is the traditional (Fregean) interpretation of all sentences as denoting one of two values: Truth or Falsity. But the general situation is much more common than just that example: different spellings with the same meaning; different procedures designating the same function; etc. Similarly, logical proof theory, defined in terms of syntax (towards the left) pays attention to far more details than does traditional model theory (though of course proof theory does not pay attention to all details, such as to when a formula was written, or to whether parentheses or brackets were used). All of these examples suggest, in general, that correspondence relations will gradually lose information, as they move towards the right, as suggested in figure 17. This assumption is for example embedded in approaches that use initial and final algebras as interpretations for programming constructs. A91

Considerations of circumstantial dependence, however, and some metaphysical arguments, suggest that this neat structure may be an artifact of formal languages, not a general truth of semantics.[55] In the general case, in other words, semantics should not be viewed as a way of moving from fine- to coarse-grained linguistic distinctions. This stance is clearly false if circumstance is ignored: different uses of the word ‘I’, as we have pointed out so often, can refer to indefinitely many different people, as can ‘now’ refer to arbitrarily many different

[55] Barwise (1986b, p. 331), in fact, defines “formal” languages to be ex-

times. But more complex phenomena suggest other structures, A92 too. For example, imagine an analysis of natural language, along the lines suggested above, that ignores different people's sense of the reference of some term 'guilt', say, or 'like'—about which interpersonal agreement is rare. If there is a fact of the matter, when a given person says "They like feeling guilty," as to A93 what aspect or property of the world is thereby named, then it follows that the *real* connection from utterance to world will discriminate more finely than our chosen semantical analysis.

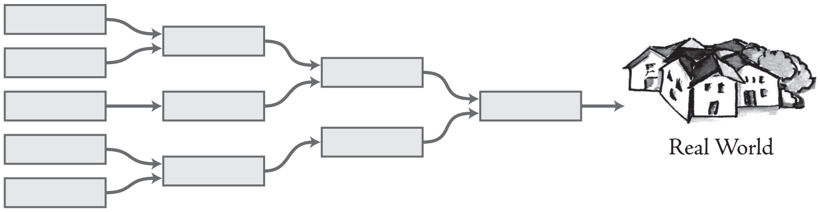


Figure 17 — The "losing information" view of semantics

I choose this example because I can imagine that it would be a serious mistake to try, in the analysis of language, to compensate for such differences by writing them in terms of an explicit parameter for something like "speaker's conceptual scheme"—what I will call **registration scheme**—and then to try to connect such a thing to our previous conception of a "pre-registered" correspondent domain. For some purposes, that is, we may not *want* to capture all the richness of the representation, nor all the richness of the world, nor all the richness of the connection between the two. But this fact still does not allow the conclusion that richness recedes as one moves to the right.

Fourth and finally, there remains the very serious metaphysical question of how any analysis at all is going to deal with the

. . .

actly those that are not circumstantially dependent in this way.

right hand end: the world itself. In fact our continuum seems to suggest that one of the great appeals of the model-theoretic semantical approach—for natural language, AI, and other systems—is that it stops the analysis half-way across the continuum. As suggested above, there are those who worry that the resulting models are still infected with the structure of the A94 languages they purport to analyze, but this has its advantages. Theorists who disagree wildly on the actual structure of the world itself (if that even means anything coherent) can nonetheless agree on a model-theoretic structure. More specifically, one would expect proportionally more agreement—among realists, skeptics, idealists, and theorists of every conceivable metaphysical stripe—to the extent that one's semantic analysis establishes a correspondence to a structure further towards the left. In fact any two people who agreed on an analysis *all the way towards the right* would by definition be of exactly the same metaphysical persuasion; that is what such agreement would *mean*.

The strongest claim I will make about metaphysical grounding will arise in the next and final section, when I return to the semantics of knowledge representation, but a preliminary point can be made here. It has to do with semantics as an instance of theoretical inquiry. To start with, make the following two relatively non-controversial assumptions.

1. Assume that we human theorists, when we use language, are somehow able to refer to the world itself, even if we do not yet know how. I.e., assume something like the most modest form of realism possible: just that there is a world, that we are in it, and that A95 our words somehow enable us to get at it. This is all perfectly compatible with everyone's carving it up in radically different ways, as dictated by nature, nurture, or just plain whim.

2. Assume as well that theories are linguistic vehicles with which we communicate our understanding to our fellow person. Or assume that theories are linguistic entities claimed to be true; for these purposes the difference does not matter.

Once these two assumptions are granted, the following is an immediate conclusion: to the extent that our theories are legitimate instances of language, and thus that we who use or understand them are able to refer to the world, it follows that, as theorists, we do not lack ways of getting to the right hand end of the diagram. I, for example, can get there right this minute with the phrase "this lukewarm cup of coffee to my right." The problem, of course, is that I do not necessarily know various things: not only how it is that I manage to refer to the cup, but also the way in which I have thereby referred to it. So the metaphysical problem for semantical theorists is not one of *referring* to the world by using theoretical language, but rather something closer to the opposite: there is no way of referring to the world *except* by using language. Neurath's boat once again.

A96 This much is obvious. What is important about it is that it is true *all the way across the continuum*: we have no way to refer to the representational structure on the left, or to any intermediating correspondent structure, outside of language either. It only feels more problematic towards the right because it is there that we encounter a natural tendency to want to escape our own particular conceptual schemes, especially if we and the representational structure in question part company. What he calls "shame" she calls 'guilt."

This may indeed be a real limitation: the chances of *completely explaining*, all the way to the right, the semantical interpretation of a system whose conceptual scheme differs radically from one's own, is almost certainly nil. Radical indeterminacy of translation, if there is such a thing, surely has what we might call radical indeterminacy of semantics as a sub-species. But there are more interesting conclusions, as suggested in figure 18.

A97 To the extent that theorist's language and representation overlap on registration scheme, the problems are clearly that much less. This is the happier case, of course, but it has this curious consequence: as analysis moves towards the right, it will *look,* to an outside observer, as if the representation in question is gradually being translated into the theorist's own language—rather as on the model of deflationary accounts of truth and reference. I.e., we might say that the French noun 'chat' (towards the left) is modeled by the objectified CAT relation (middle), which in turn characterizes the set of *real cats* (right). Viz.: quotation on the left, reification or nominalization in the middle, and mundane use of a term in the theorist's native language on the right. But this is just as it should be; it is predicted by the diagram. There is absolutely no reason to conclude, from this observation, that semantics inherently involves translation.

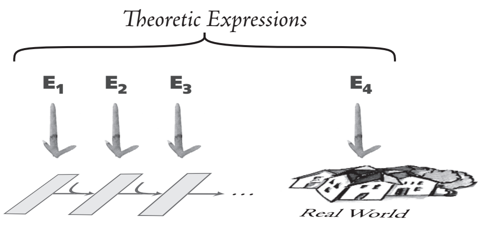


Figure 18 — Semantics of theories of correspondence

On the other hand, to the extent that the theorist's registration scheme is his own, it will be so all the way across the diagram. Just because the theorist registers the representational structure itself in terms of a given set of properties and relations (say, as having a particular syntactic form), there is no reason to believe that the representational system registers itself in this way—if indeed there is any reason to suppose that it registers itself at all. I.e., if, as I am inclined to suppose, registration involves representation (as well as *vice versa),* then the subject system will register *only* what is to the right; the rest is registered *only for theoretical purposes.*[56] As before, conflict can occur only at the right hand end, but only because that is the only thing that *both* system and theorist register. A98

In sum, the idea that semantics involves translation is a superficial rendering of the much deeper though perfectly straightforward fact that semantical analysis, like all theoretical investigation, is carried on in language, left through middle through right.

11 Knowledge Representation Revisited

Although we may seem to have strayed a fair distance from knowledge representation, its demands have been our constant motivation. First, we have seen that the semantical competition between 'representation' and 'knowledge' was merely the tip of a rather large iceberg: without even trying to enumerate an exhaustive list, half a dozen other intentional notions were added to the semantical roster. Second, with respect to appropriate semantical technique, I argued for the prior development of a comprehensive theory of correspondence, and sketched some preparatory philosophical foundations. One way to view this proposed theory is as a branch of semi-mathematics that would immeasurably aid semantics in two ways: by clarifying the semantical project itself, and by providing conceptual vocabulary in terms of which to classify genuinely semantic relations.

On the other hand, I have tried to say plainly that a theory of correspondence would not itself be a theory of semantics, or representation, or knowledge. In spite of all the ground we have covered, in fact, I have said virtually nothing here about the essence of any such notions. Even §9, which tried to sketch A99 some of the structure in which semantics would proceed, still

[56] [27] The theorist, of course, can either be us, or else the system introspecting on itself; see Smith (1986) [ch. 5].

did nothing to resolve this piece of homework. Nor can I do more here. My only intent, by way of a last conclusion, is to make one brief foray in this direction, which will tie the whole analysis back to the primary distinction made at the outset, between representational import and functional role.

The point is simple. I said that functional role and representational import must be coordinated: the agent must be able to act sensibly in terms of what it represents, and (perhaps) represent what it can act sensibly towards. This coordination can be viewed as a kind of "coming together" of knowledge (second factor) and action (first factor). Thus, suppose, knowing the paper is almost over, I reject the tepid coffee on my right in favour of the following rough plan: that when I have completed it, I will avail myself of some of the Lagavulin in the cupboard. When the time comes, I would like my internal impression that represents the Lagavulin (as part of my intention to have some) to engender the action of my crossing the room, pouring out a glass, and raising it to my lips. What is of paramount importance, for our purposes, is the following fact: in the terms of the continuum diagram, this coming together or "coincidence" of representation import and action (which is one kind of functional role) must be *all the way to the right.* A100 I want to drink *what is in the world,* not a model or indirect classification of smoky whiskey, nor a term model of 'Lagavulin' expressions, nor a set-theoretic assemblage of sentences or impressions containing representations of the property of being whiskey. Whatever "stuff itself" is—whether it is ontology or "beyond ontology"—this much is certain: it is stuff itself towards which my actions must be directed.

This observation, merely a theoretical consequence of the dual facts that action takes place in the world, and that functional role is a kind of action, is the grounds for my sixth and final challenge to the model-theoretic tradition, promised earlier. Because computer systems participate with us in the world—stop our cars, launch our weapons, deliver our mail—it is imperative that our analyses of the representational import of impressions take us *all the way to the real world situations* towards which the engendered action will be directed. Tooth decay among children will not be reduced by a computer's injecting a mathematical model of fluorine into a set of possible worlds. In order to see the coordination between functional role and representational import, that is, both parts of our two-factor analysis of significance *must reach all the way to the right*. Let's call an analysis that reaches out that far a **grounded** account. A101

So far, then, the only coordination requirement I will put on theories of full significance is that they be grounded. At least for the moment, that will have to be requirement enough.

Annotations†

A1 ·1/79 «…Intro; accepted but submitted too late; prep for fan-calculus… One of the most important papers—or anyway insights—in the book. A bit hard to get going (warning)…»

A2 ·1/2/3 The phrase “circumstantially-dependent” signals one of the most important aspects of the overall view expressed in this paper: that whether any given distinction (e.g., between a numeral and a number, between different tokens of a numeral type, between the copy of the file in memory and on disk,[0.5] etc.) is relevant or germane—needs to be taken into account—is a matter of circumstance and perspective. What the paper does not do explicitly is to embrace one of the most important aims of the proposed fan-calculus, which grows out of this paper: that whether such distinctions should be visible should be a dynamically-controllable fact of the viewer (the “registrar,” to use the vocabulary of O3).

A3 ·3/1/-2 See «…» re my use of ‘they’ as a semantically singular but syntactically plural pronoun of unmarked sex.

A4 ·3/2/-4:-3 By ‘representation,’ here, as was typical in AI at the time, I meant “representational structures playing a causal role in engendering behavior in a computational system,” not representation *simpliciter*. Even on that reading, however, while nothing in the paragraph explains or even suggests how representation and knowledge differ, philosophical readers will know these as distinct technical terms, and so to them this passage will read oddly. In AI, however, especially when this was written, no such clear understanding of the difference was in place. Of particular importance is the fact that knowledge (and belief) would or anyway should be taken as personal-level characteristics, whereas the sense of representation in play is clearly at the sub-personal (sub-system) level.[1]

One might therefore imagine that the paper would aim to clarify the distinction for AI readers. And it is certainly true that I do nothing here to consider issues of truth or commitment—issues which

†References are in the form page/paragraph/line; with ranges (of any type) indicated as x:y. For details see the explanation on the next page.

[0.5] Or, for that matter, between the copy of the file held in the disk controller’s (fast) memory cache vs. the version actually “written” to the (slower) disk or non-volatile solid state memory.

[1] Cf. the discussion at «…» of the only just apparent use of this distinction throughout these papers.

with any real analysis of knowledge would need to engage, especially as regards the relation between knowledge and belief. Nevertheless, it was closer to my intent, given the reading of 'representation' just identified, to undermine philosophical presumption that any of these distinction—personal/subpersonal, representation/belief,

Annotation Text References

Text references are in the form page/paragraph/line, with ranges (of any type) indicated as α:β. Thus in a reference *x*/*y*/*z*:

1. *x* is the page number
2. *y* is the paragraph number
   a. *y* omitted ⇒ no ¶; if *z* present, *z*th line on p. *x*; else all p. *x*
   b. $y = 0$ ⇒ partial ¶ at top of p. (cont'd from previous p.)
   c. $y > 0$ ⇒ *y*th paragraph starting on p. *x*
   d. $y < 0$ ⇒ count from bottom (-1 is last ¶ starting on p.)
   e. Footnotes indicated as paragraph *n1*, *n2*, etc.
   f. Section headings are not considered ¶s, but indented points are (e.g., bulleted or numbered).
   g. Tables, figures, etc., not considered ¶s; should be referred to explicitly.
3. *z* is the line number
   a. *z* omitted ⇒ no line no.; if *y* present, whole ¶; else whole p.
   b. $z > 0$ ⇒ if *y* present, *z*th line in ¶; else *z*th line on p.
   c. $z < 0$ ⇒ count from bottom of ¶ or p. (-1 is last line)
   d. Example: 8/-1/-3 means 3rd-from-last line on p. 8, even if last ¶ (-1) continues onto next p. (9/0/...).
4. x_1:x_2, y_1:y_2, and z_1:z_2 mean from x_1 to x_2, etc. (pages, ¶s, or lines). Missing values default from the left:
   a. 23/4/5:7 ⇒ lines 5 through 7 in 4th ¶ on p. 23
   b. 23/2/6:4/3 ⇒ 6th line of 2nd ¶ through 3rd line of 4th ¶ on p. 23
   c. 23/1:24/7 ⇒ 1st ¶ on p. 23 through 7th ¶ on p. 24
   d. 127/2/9:128/4 would mean 9th line of 2nd ¶ on p. 127 through 4th line of 128th ¶—probably not what was intended. To signify 9th line of 2nd ¶ on page 127 through 4th ¶ on p. 128, use 127/2/9:128/4/.
   e. 127//-9:128//4 ⇒ 9th line from bottom of p. 127 through 4th line of p. 128

etc.—is as clear as has traditionally been thought. Still, I would be the first to admit that the first sentence of the following paragraph ("This paper will try to sort this all out") is rather disingenuous.

A5 :3/-1/1: :4//1 As noted above (annotation A4), there is more to full-blooded intentionality than semantics, at least on widely-accepted readings of the term 'semantics.' Consider as well the widely-disputed differences between authentic or original intentionality, on the one hand, vs. derivative or derived, on the other; semantical analyses typically ignore all such issues. Not only was it semantical issues with which I was primarily concerned when I wrote this paper, however, but—with the possible exception of consciousness—at the time this paper was written (mid 1980s) semantics was pretty much the only facet of intentionality that had made it into the AI imaginary.

A6 :4/-1 Not only is it still considered important to provide semantical analyses of contemporary computer languages, including the increasingly popular web and interchange languages such as OWL, RDF, etc., but, almost forty years after his initial remarks, Pat Hayes remains a force not only in pressing for semantical analysis and clarity, but in recommending the use of logic and traditional model-theory. Cf. «…».

A7 :4/1/-6 Cf. ch. 10 (including its annotation A3), which in a severely restricted case presents a semantical analysis of a continuous (analog) form of representation.

A8 :5/-1/1 Although I adopt this two-factor analysis here, as used for example throughout the analysis of 2Lisp and 3Lisp in Part B, part of the aim of this paper is to undermine the clarity of the very distinction between the two factors—especially any sense that they are in any sense *independent*. See for example §«…». Cf. also the discussion in §«…» of the Introduction, about the use of simplistic "formal" dimensions to get at intentionality's semantic complexities.

A9 :5/-1/4 As noted in annotation A5, above, nothing need be assumed at this point about whether such content is authentic, derivative, instrumental, ascribed (e.g., alone the lines of something like Dennett's "intentional stance"[2]), etc.

A10 :5/-1/7 I was certainly aware of the Humean allusion when I introduced this term, though I was taking no stand on liveliness, relation to ideas, or anything of the sort.

[2] «Ref…»

A11 ·5/-1/-5:-2 Not much depends on which of these conceptions of an impression was had in mind, but I did assume, albeit implicitly, that impressions were "sub-personal" (or "sub-system") ingredients. Cf. «…»

A12 ·6//1 'Functional role' is of course not my term. Within the philosophy of mind, it is a general term of art of functionalism, where it is often used synomymously with *conceptual role*,[3] to signify not only the causal or effective role that a mental state plays in engendering behavior, but more significantly, that by which mental states are identified or individuated (implying that any mental states that plays the same functional role are, perforce, the same mental state).

A13 ·7/1/3:-5 These statements about tree sap might be viewed as ill-advised, on at least two counts. First, although (depending on one's view of representation) one might say that the level of sap does not function as a representation unless so interpreted by an external observer, it might still, especially these days, be counted as intentional, for example in *carrying information*[4] about the environmental circumstances of the tree—history, rainfall, etc.). Second, causal relations, like intentional relations, are also intensionally fine-grained—relating *ways things are* (states of affairs), not just the objects that are that way. To confuse fine-grained (property-level) correlation with "concepts being involved" was at best naive. The last sentence in the paragraph—that such regularities "would be missed in an isolated account of functional role"—would thus be false, if the account of functional role were fine-enough grained to get at the details of how and why the internal structures in question were causally produced and led to causal consequences.

A14 ·7/-1/1:2 "[S]uppose I have the impression" is meant technically, here—i.e., as having roughly the same meaning as: "suppose that my head is

[3] Not only is the term 'conceptual' specifically *mental*, blocking any natural application to the computational context, but to presume that conceptual role is equivalent to functional role depends on believing that mental content must be conceptual content—a thesis denied by advocates of nonceptual content (of whom I count msyelf as one, though the phrase is unfortunate in being defined in opposition to something, rather than stating positively what it intends). Cf. "The Nonconceptual World," ch. «…» in Volume II.

[4] Not only is carrying information now widely taken to be intentional; it is popular, these days, to take information-carrying to be the ground of intentionality—the "ur" intentional case in terms of which (with the possible addition of evolutionary function) more complex forms arise. I myself do not subscribe to this view, however. Cf. AOS, especially Volume IV.

internally constituted, in part, by a subpersonal impression whose representational content is that water conducts electricity, and whose functional role, at that level, is to lead me, as a whole person, to believe or even know that water conducts electricity."

A15 ·8/1/1 Cf. the discussion in §… of the Introduction regarding the coordination of content and causal coordination constitutive of logic.

A16 ·8/1/4 Differences between knowledge and belief, which philosophy takes very seriously, were not prominent in AI in the 1980s. Cf. annotation A4, above.

A17 ·8/1/-1 In general, my use of the term "full significance" is an attempt to reach out not only towards an indissolubly entangled conception of functional (conceptual) role and representational import, but all other aspects constitutive of full-blooded intentionality. I used the phrase in the discussions of 2Lisp and 3Lisp in Part B, where it was signified as Σ, but in that case went only as far as the integration of the two factors. In the context of this paper, as well, it is largely restricted to this narrower meaning (e.g., cf. the use at 12/2/1:3).

A18 ·9/1/1 For an analysis of the sense of *independence* meant here, and implicated as well in the widely held view that a computation proceeds "independently of the semantics of its ingredient symbols" (the so-called "formality condition"), cf. AOS Volume II. A glimpse of that analysis is provided in ch. 1.

A19 ·10/2/8:9 It would have been happier if this sentence had been phrased as "what a given use or instance of that type refers to, or gets at, in all its *particularity*." At the risk of being pedantic, I take it that 'specific' and 'specificity' should be used to what is related or connected to a *species*, whereas what I intended here was to point to the semantic import of the *concrete individual instance or use*.

A20 ·10//-1 This equation exemplifies the "λ*context.content*" style of semantic analysis of contextual depedendence discussed at «…; include ch. 2»

A21 ·11/1/1:3 As discussed at «…», whereas the focus of the present paper was on correspondences of a roughly semantic or intentional sort, it is based on an underlying theme that we need to be able to make distinctions between like things if and when appropriate, rather than never or all the time. As mention in annotation A1, above, one of the intents of the fan calculus («…») is to deal with this theme in its many manifestations—of which distinctions between and among type, token, instance, use, etc., are a paradigmatic example.

A22 ·11/1/-3 Cf. ch. 10.

A23 ·12/-1/1 It might seem that the term "our current semantical arsenal" means the semantical techniques of logic and philosophy, since (as I myself argue elsewhere—cf. for example «…») virtually all of what goes on in semantical analyses in computer science focuses on functional role and behavior. The point, however, is not that our tools in our current arsenal are *used* to analyze representational import, but that they were originally developed within the context of logic and philosophy, where representational import was at issue.[5]

A24 ·13/0/1 As noted above («…») this is interpretation in the logical/philosophical sense, not as in computer science's notion of an *interpreter*. See «…».

A25 ·13/0/1:2 Unfortunately, the phrase "described extensionally" is ambiguous, as to whether what is extensional is the description itself, or the description's referent. As is evident from the remainder of this sentence, and from the subsequent one, the latter is closer to what I had in mind—except that the two alternatives relate in complex ways.

I would have done better to put it as follows. "The task of semantic analysis is usually taken to be one of describing, in a referentially transparent way, the *extension* of the interpretation function. That is: if α, a term in the semantical theory, is used to describe or denote[6] the interpretation function F, then α would be assumed to be used in a transparent context to refer to the range of F. In practice, however, to assume that theoretical rigor requires no more than transparent reference to F's extension belies the enduring epistemological fact that theory is intended not only to be true, but also to lead to human understanding. In many cases, the *intensional* character of α will convey additional information about the *intensional* structure of F—information critical not only to human comprehension of the theory, but comprehension of what they theory is about."

[5] It could be argued that computer science has developed traditional tools in new ways, which adjust them categorically towards causal/functional use. A dramatic case, for example, is Girard's linear logic («ref…»), , which I believe has torqued the notion of logic almost wholly to purely mechanistic ends. Still, I believe that the declarative/representational origins of even contemporary semantical frameworks are of intellectual as well as historical significance.

[6] Whether to say *denote*, *describe*, or *refer to*, in this sentence (I am using the terms roughly interchangeably here) depends, of course, on issues closely related to the very point being discussed.

Intricate relations between intension and extension in meta language and object language are exactly the sort of issue with which the paper is ultimately concerned. (Cf. also annotation A53, below.)

A26 ·13/1/4 In the original version of this paper, I used the term *model-theoretic* in place of *classical*, but on reflection that does not seem the best label for the four commitments taken to be constitutive of the view: (i) compositional semantics, (ii) clear use/mention distinction, (iii) parameterization to deal with contextuality, and (iv) model theoretic treatment of the semantic domain. 'Model-theoretic' is really a name for only the fourth. While it is true that many or even most model-theoretic accounts of semantics meet all four conditions, to call the entire suite model-theoretic would unncessarily narrow the target of criticism.

A27 ·13/-1:14/0 Cf. §… of the Introduction, as well as the characterization of language in Fodor's classic …, in … ⟦reply to Smolensky⟧

A28 ·19/2/13 The notion of *causal efficacy* indicated here is exactly what it is that I take the (so-called) mathematical theory of computation (or computability) to be a theory *of*. Cf. «…» and «…».

A29 ·21/0/-2:-1 Strictly speaking, at least in 3Lisp, there is no way to denote an impression with an expression; denotation is defined only over impressions themselves. Impressions can be denoted with other impressions (called handles); thus the 3Lisp expression «'3» notates a handle that denotes the impression mentioned in the text—i.e., the (canonical) impression denoting the number three. Similarly, the 3Lisp expression «"3"» notates an impression that denotes the expression that notates the string that denotes the character (expression) that notates the impression (numeral) that denotes that same number—the character that in English we would normally mention using single quotation marks, as in: '3'.

The pedantry of the machinations involved in handling this sort of situation carefully are of exactly the sort that this paper was motivated to address—and that the fan calculus is aimed to ameliorate.

A30 ·21/1/12 Abstract[7] syntax is designed to ignore (or translate[8]) manifestly irrelevant aspects of the so-called "concrete syntax" of external expres-

[7] Etymologically: to draw or pull away.

[8] Manifestly concrete aspects of external expressions are sometimes used to indicate structure—as for example indentation was used in KRL to indicate scope. In such a case the "abstract syntax" would typically be an internal data type or data structure (impression) in which scope was explicitly encoded in

sion—although, interestingly, the result is always a perfectly concrete impression, not an "abstraction" in the sense of being mathematical or Platonic. How much is ignored, in any given case, is a matter of design aesthetic. Typically, the resulting impression is "less abstract" that it might be—retaining more of what could be argued to be "contingents facts of one- or two-dimensional expression than strictly necessary. What would usually be meant by an "abstract syntax" for the λ-calculus would likely retain the notion of a named variable with one or more occurrences inside complex λ-expressions, for example—and would thus continue to require α-renaming in a rewrite-rule regimen in order to avoid variable capture. In contrast, any need for α-renaming would be obviated in a more "abstract" version if multiple occurrences of the "same" variable—including its "occurrence" at the binding site—were replaced in the corresponding impressions by actual co-occurrence (i.e., if the result were allowed to be a graph, without any "names" at all).

A31 ·22/1/7:11 The similarity to functionalism in the philosophy of mind is evident; what is philosophically interesting is the recruitment of computer science's notion of an abstract data type as a mechanism to implement such a functionalist approach (cf. the remarks in §... of the Introduction about the meeting the requirements of concrete construction; also footnote 22 [9]).

A32 ·24/0/1 EMACS (originally developed as a set of Editor MACros for the programming language TECO) is a highly extensible text editor, developed in the mid 1970s and continuing to be used today. 2Lisp and 3Lisp were developed using EMACS. «...other annotations?...»

A33 ·23/-1: 24/0 Both (i) distinguishing programs from processes as strongly as this paragraph suggests, and (ii) distinguishing a program viewed or "registered" as a (textual or graphical) expression from the impressions that will become causally efficacious parts of the process that results, are examples of the sorts of excessive strictness that this paper was ultimately an attempt to undermine. So there is something ironic in my striving to be as strict in making these distinctions as I am in these pages.[8.5]

At the time, I was mostly focused on excessively strict semantic/

the data structure as such (e.g., via hiearchical arrangement in a tree), with the fact of its having been externally indicated wih indentation set aside.

[8.5] The strategy does exemplify that I often tell students—that, curiously, the deepest insights often take the following two-part form: first of showing that intentional distinctions—and not so much concerned with cascading complexities of general ontological distinction, particularly of the "one vs. many" variety. But as indicated above in annotation A21, such ontological distinctions are definitely targets of the fan calculus—so as to allow one to say that "the program prints out 'HELLO WORLD'," without that being sloppy, or a category or type error.

A34 ·25/1 Cf. "100 Billion Lines of C++," included here as chapter 7; and much of ch. 2.

A35 ·27/2/2:5 Re the term 'specific': cf. annotation A19, above (and note, too, the last sentence in the paragraph).

A36 ·28/0 For example, suppose one were to construct a theorem prover in which: (i) impressions were sentences, modeled by the sets of all possible worlds in which they were true; and (ii) a variable C was defined, assumed to denote 0 if the set of sentence is consistent, and 1 if not. Then consider two proposed state transitions:

τ_1: If some sentence S is entered into the memory, them the process should halt; and

τ_2: If C changes from 0 to 1,[9] then the process should halt.

As is perfectly evident to any programmer, τ_1 would be trivial to implement, whereas τ_2 would not—potentially being either extremely (impractically) time-consuming, or perhaps impossible altogether, depending on the notion of consistency being used.

There is nothing odd about the example. The point, rather, is that, if, as suggested, the sentences are modeled as sets of possible worlds, and consistency by a simple integer, then nothing in the model provides any explanation of why one transition is feasible, the other infeasible. In this way the models are explanatorily incomplete.

some notion or phenomenon α is not in fact unitary, as commonly assumed, but instead must be understood as an unfortunate fusion or admixture of β and γ, which must be understood and analysed separately; and then second, once one β and γ have been adequately disentangled, of showing that, far from being independent, β and γ are intimately and inexorably interrelated.

[9] Or, equivalently: "τ_2': Variable C should be set to 1 if a sentence is entered that makes the whole set of impressions inconsistent. τ_2' would be just as infeasible to implement as τ_2; I use the example in the text only because this latter case, though seemingly simpler, depends on what it means to bind a variable, a more semantically complex issue than is generally recognized within computer science).

A37 ·28/-1/2:3 I apologize for what will inevitably be distracting to the reader of this volume: that, because of adopting the specificational view of programs in this paper, in favour of the ingredientiel one used throughout the analysis of 2Lisp and 3Lisp, the framing of very similar issues will be different here than in Part B.

I changed frameworks in part out of some disappointment with the fact articulated in ch. 2: that the semantical thesis on which the 2/3Lisp architecture was based had seemed to have no impact at all, either AI or in computer science, in spite of the recognition that 3Lisp otherwise garnered. Unfortunately, adopting the specificational view did not help. The challenges to accepted informal ways of understanding programs and processes that arise from attempting to be strict about semantical and intentional issues cut deeper than a simple choice of words was able to affect. More seriously, as supported in part by the ultimate conclusion of this paper, combining semantical insight with practicable usability presents a more profound challenge to our understanding than has yet been met.

A38 ·29/1/-6:-5 Model $\mathcal{M}_c$ of computational process $\mathcal{C}$ *does* model content, of course, as 'content' is understood on the specificational view: that the process (or uninterpreted behavior engendered by that process) is the content of the *program*. What the passage in the text was meant to imply was that modeling the process set-theoretically would tend to suggest that one had already analyzed its content, because set theoretic structures are so commonly associated with semantics, in way that dealing with the process directly—i.e., as as something concrete, with causal antecedents and effects, perhaps constituted of impressions and activities between and among them—might make it more evident that there remained a semantic relation, yet to be analyzed, holding between that process and the task domain or world $\mathcal{W}$.

A39 ·31/3/1:2 AI's (in)famous "closed-world" assumption—that each (relevant) object in the task domain is representation by a unique object in the process or structural field—is a particular instance of such an assumption of isomorphism between the process and the world that it is about.

A40 ·31/-1/-2
·32/0 This passage ignores the fact that mathematics itself engages in ubiquitous modeling. For example, rather than being treated as an ac-

[10] Or, to put it in the language of §8 (see in particular ·55/0/5:8), the closed-world assumption is an assumption that "object identity is absorbed." Cf. the discussion at ·61/-1/-5:-3.

tual function, the factorial function would likely be modelled as an infinite set of ordered pairs of natural numbers.[10.5] Had the point been noted in the paper, however, that would only have strengthened the conclusions that it reaches.

A41 :32/1/3:5 That is: "to the right of," treated as a unary predicate, is egocentrically deictic or indexical. Cf. «…».

A42 :32/3/1 «…check whether the following needs to be rewritten, given §6…». A substantial ambiguity runs through this and the next several paragraphs, serious enough to tempt me to rewrite the whole section, and undoubtedly distracting to the reader.

In the case of 2Lisp and 3Lisp, and throughout the papers in Part B, I adopted an ingrediential view of programs (cf. annotation A37). Because of this theoretical assumption, I viewed the relation under discussion here—that between an external *expression* and internal *impression*—in terms of *internalization* and *externalization*, labeled θ and θ^{-1}, respectively. Given that I am here working within a specificational view of programs, on the other hand, in what precedes these paragraphs I have said that I will take the "semantics" of programs to be a function or relation that maps programs, themselves viewed as static or at least passive expressions, onto the *processes* or *behaviors* to which they give rise upon execution.

In this paragraph, however, I somewhat mix the two perspectives up, by focusing on the relation between expressions and impressions, under a specificational view. The difficulty is that, as stated—at least as *I* have described it—the specificational model does not make room for impressions, per se. In some sense they are elided with the program, in the case of those impressions that are instructive or code; in cases of data structures (elements of a data base, etc.) they are more likely be elided with, or taken to be constituents of, the process or behavior.

Even on a specificational view of programs, strictly speaking one ought to distinguish external (text-like) expressions and the internal impressions (digital arrangements?) into which they are translated or converted. Or at least one should do that when, as here, the differences between them are at issue.

[10.5] Functions are not *actually* infinite sets—as is obvious from the fact that the very same function could have been modeled differently For example, the ordering of each pair could have been reversed, with the element of the range occurring first in each pair, and the element of the domain. second.

In the end I decided not to rewrite the paragraphs, as that would violate the presumption that this is indeed a 1986 paper being reprinted 26 years later. Instead, it seems best to draw two morals.

1. Distinguishing between and among expressions, programs, impressions, data structures, processes, and the phenomena or situations in the world that they are respectively about—and cross-cutting the lot of them with mathematical models of them adopted for purposes of theoretical purposes—is a perfect example of the complexities towards which the paper as a whole was addressed. Failing to make such distinctions can sometimes, as in these passages, engender confusion; requiring that one always make all these distinctions, on the other hand, quickly grows so baroque and fastidious that the results are hopelessly complex.

   This paper diagnoses the problem, but it barely points at a way towards solving it. The calculus of correspondence proposed later in this paper would be a framework in terms of which to *make* all these distinctions; it ducks the harder problem, of when to make them—and how to proceed, in a contextually-sensitive way, that allows them to be addressed (distinguished) when appropriate, and elided otherwise—without thereby generating fatal ambiguity or contradiction. It is to that task that the proposed fan calculus will be addressed.
2. Second, it does not escape my attention that, in spite of writing this paper on these very points, in the thick of the argument I myself was waylaid by the very sorts of confusion of which I was in the processes striving to make sense.

A43 ·32/3/-3 A critic might object that there is no way, within the bounds of physics, to implement anything as "unordered"—though as legions of programmers know only too well, the order need not be specified as a contract of the implementation, implying that two implementations of the "same language" may differ in how things are produced, or one implementation at different times, or in different circumstances, etc.

As discussed in the Cover to Part B, one of the design criteria for Mantiq was for it to have a relatively abstractly defined structural field, along the lines suggested in this paragraph (·32/3).

A44 :32/-1/4 Cf. annotation A42, above.

A45 :33/3/2 It is assumed in this case that the impression is an ingredient of the computational process called c in figure 8.

A46 :33/3/-6 It is somewhat misleading to say "identifying all semantically equivalent expressions," since if this were to be a real implementation strategy, at best the individuation criteria on (i.e.,identity conditions of) impressions would derive from *proof-theoretic* interchangeability. The presumption, in line with the analysis of logic discussed in «§…» of the Introduction, is that proof-theoretic interchangeability or interconvertibility would, in so far as was mechanically feasible, mimic semantic equivalence.[11]

A47 :34/0 A formal proof that neither of c_1 nor c_2 could satisfy the requirements of representational import would demand much more detailed specifications than is provided here of what such requirements came to, and of the nature of c_1 and c_2 themselves.

Take it as an informal requirement on import that it specify places in such a way as to make it evident whether the places indicated by the mental states of distinct robots were the one and the same place in the physical world (i.e., so as to make it manifest that two robots were about collide). It is unlikely that c_1 would stand a chance of being suitable, since, by stipulation, it consists of no more than sets of proof-theoretically equivalent indexical sentences; they could be used to determine an exocentric physical location only with reference to the position of the robot itself. Similarly, since c_2 is formulated in terms of unary predicates, only, again, if their relationality to robot position was explicitly encoded would they contain enough information to determine exocentric position.

The point is simple: neither c_1 nor c_2 would allow one to conclude, from the fact that the representational import of impressions in two different robots were the same, that they thereby "had in mind" the same location in external physical space (or, contrapositively, that though the c_1s and c_2s of two robots were *different*, they were nevertheless representing the *same* place).

A48 :36/0/2:5 It can be a complex issue as to how much circumstantial relativity needs to be retained, and how much discharged, in order to get at the appropriate notion of "meaning" or "content" to explain any given intentional regularity.

[11] The notion of semantic equivalence would be of course be relative to a particular conception of semantics (in this case assumed to be standard).

It is certainly insufficient to presume, in a binary way, that any given designator "is or is not" indexical or circumstantially dependent. At a minimum, distinctions must be made as between types, tokens, and uses—i.e., as to whether the circumstantial facts influence the semantics of tokens, in relation to the type; or whether different uses of a single token may have circumstantially different meanings or contents.[12] For example, consider the first-person pronoun in language, and the debate between Perry and Millikan[13] about whether its mental correlate is or is not indexical. Consider (external, communicative) language first. We say that the word 'I' is indexical, since, its designation, rather than being constant, varies circumstantially (roughly along the lines of λ*speaker*.*speaker*). But this is an informal characterization. Strictly speaking, what is indexical is the word *qua type*. The type is called indexical because, as we say, its various instances have different designations.

In this case, 'instance' is a gloss for something like 'use.' That is, in the terms introduced in the Introduction, there is a single presumptive "fan-out," from type to particular occurrences of use. But the situation is more complex than that suggests. In particular, there is a group of uses of the pronoun—all those used by a particular person—whose designation does not vary. Crucially, every time *I* use the word 'I' I refer to the same enduring (or perduring!) self; the same holds of you; etc. Thus a more illuminating analysis might treat the situation in terms of a staged set of fan-outs: a first stage, from type to groups of uses by individuals; and then a second stage, from each of those groups of uses to the group's members. The first stage of fan-out is clearly indexical; the second, needless to say, is not, since all members within each group co-designate.

This is the issue in the Perry-Millikan debate. Millikan, evidently enough, is concerned with the mental analogue of the second stage of fan-out—from a particular individual's internal representation of themself, to the reference *on occasions of that representation's being used or tokened*. Perry's concern (by analogy with language) is with the combined first-and-second stages.

It is not that one is right and one wrong; nor, in my judgment, do they disagree on what 'indexical' means. The issue is simply one of

[12] As suggested in the Introduction, the complexity of computational cases outstrips even this classical trichotomy.

[13] Perry (19…); Millikan (1990)

insufficiently clearly articulated ontology—specifically, with a richer fan-out structure than is imagined in our traditional analytic frames.

A49 :37/0/9:11 The claim that denotational and operational semantics are "distinct ways of getting at the same phenomenon" was intended to be in contrast with proof theory and model theory, which I presumed to be analyses of *different* phenomena (what can be formally derived from a given sentence or formulae, and what that sentence or formulae denotes or entails, respectively). One could claim, of course, that derivability (⊢) and entailment (⊨) can be proved equivalent in any logical system that is (demonstrably) sound and complete, but such proofs, I take it, are *substantive*: they make a non-tautological claim on the nature of the logical system in question. Equivalence proofs between operational and denotational semantics of programming languages, in contrast, are meta-theoretic and "purely formal": since at the object level (the level of the subject matter under consideration) there is no ontological distinction in subject matters, the fact that the two accounts can be proved equivalent is a condition on adequacy of the metatheoretic framework. Both are accounts of what happens, computationally, when the programs run; both thus fall within the "proof theory" side of logical analysis.

A50 :37/n29 ⟦Relate this to other annotations and meta-discussion in this volume (ch.2) (and vice-versa?)⟧

A51 :38/1/7:9 That implementation is "semantically opaque," at least as regards anything like denotational or referential semantics (as those words are being used here), is a hugely important point, the full theoretical consequences of which are still far from clear. The point is in some sense true only sub-personally (i.e., at a sub-system level); if a person who is thinking about the rise of the religious right is "implemented" (modulo issues of contextual location) in terms of a stupefyingly complex assemblage of neurons and other bodily cells, then at a personal level that complex assemblage, too, is thinking about the rise of the religious right as well. That, in effect, is what "personal level" *means*: the complex assemblage "is" the whole person. But at a sub-system level, no such implication holds. This is why it is perfectly possible, as mentioned in the text, to implement a non-sound inference system β in a sound one α. Suppose α is a sound first-order theorem prover, in which one implements a risky, heuristic higher-order belief revision system with a tendency to leap to false conclusions. β's unsoundness is no threat to the soundness of α. The natural denotation of the first-order expressions constitutive of α would be the *formulae* of β, taken syntactically.

As I put it in my teaching, and as described in annotation «…where?…», semantic interpretation (again, in the referential or denotational sense familiar from logic and natural language) "does not **cross implementation boundaries**." There are at least two reasons why this fact is of such theoretical consequence: (i) implementation is computationally ubiquitous—being one the most common of all programming techniques, as well as most powerful; and (ii) we as yet have no theory of implementation in terms of which to identify when and where one system (or part thereof) is implemented in another.

One natural suggestion is to use implementation's referential opacity for purposes of identification or even definition: i.e., if computational structure and associated set of behaviors X is "made up" out of lower-level computational structures and associated set of behaviors Y, then to say that X is *implemented* in Y, rather than being merely *mereologically composed* or *constituted* out of Y, just in case the (referential) semantics of X differ from the (referential) semantics of Y.

This suggestion has definite merit. Note, though, its double entailment: (i) that referential semantics be determined other than purely behaviorally,[14] implying in turn that implementation cannot be understood as a simple issue of "coarse-graining"; and (ii) that implementation, so conceived, becomes an intentional notion, not evidently naturalistic. Both conclusions are ones I am happy to accept. But whether the suggestion will ultimately prove workable I am not yet ready to say.

A52·38/n30/-1: What I must have had in mind is that, because model-theoretic
:39/nc/0 techniques often abstract away from (elide) what are considered "unimportant details," it may sometimes be the case that differences between functional role and declarative content are "disappeared" in the resultings models. Situations in which this is likely to occur in-

[14] Or at least not pure *mechanically*. In philosophy 'behavior' is normally understood as an intentionally-laden notion (to be distinguished from "mere bumping and shoving"), whereas in computer science the notion is ubiquitously understood mechanically (and hence would be assumed, if the question arose, to be perfectly naturalistic).

clude both: (i) mathematical cases, where such differences as those between numerals and numbers are ignored in the model (perhaps also functions and their designators); and (ii) metastructural situations, of the sort common in reflection (cf. Part B), where one computational structure (declaratively) designates another. In the latter case, what is "returned" may either *be* what is designated (as in traditional Lisps), or be a simple (what in 2/3Lisp I call a "normal form") designator of that structure, sufficiently similar or istomorphic to the computational structure thereby designated as to be identified with it in the model-theoretic analysis.[14.5]

A53 ·40/0/-4:-2 Especially among those of a psychological bent, there are some who use the term 'semantics' for that which I am here calling internalization: i.e., for the "language-mind" relation between external (communicative) linguistic sentences or utterances and their correlated (productive or receptive) internal mental structures. I myself resist the practice, since I believe the "mind-world" and "language-world" referential (and non-mechanical) relations to distal subject matters remain, especially normatively, the most important relations of intentional directedness. Nevertheless, one of the overall aims of the paper is to argue that the language-mind relation, among others, is part of the not entirely disentangleable "semantic soup"[15] of intentional relatedness that characterize complex intentional systems. In the end, there may be no reason to grant any one ingredient relation—no aspect of the whole complex that, as theorists, we focus on and identify as a single relation—any more of a claim on the term 'semantics' than any other.[14.7]

A54 ·41/0/-5 Cf. the discussion in annotation A42, above. This discussion, too, is more confused than it should have been.

A55 ·42/8 Again, cf. annotation A42.

A56 ·44/-1 This section (§7) is a good example of the point made in §... of the Introduction: something that may seem, on the surface, to be the arrival, after a mighty technical struggle, at a point all-too-familiar to discursive theorists of a wide variety of stripes. That one cannot single out any one binary relation that constitutes the "core" of the

[14.5] This was the case in Goguen & Meseguer's proposed "denotational semantics" for 2Lisp, which was in fact a model-theoretic analysis of functional role. See ch.2, ·.......

[14.7] Cf. also ·49/2.

[15] See ¶7 (p. ·44–51), especially at ·47/-1.

intentional phenenomenon, be it as contextual or indexical or cirumstantial as you please, but rather that every effable and ineffable dimension of the entire intentional situation contributes, in a reflexive and co-constitutive bricolage of ways, to the overall significance of intentional activity—something of this sort would be viewed, in many discursive traditions, as a methodologically-warranted *starting point* of inquiry, not as a conclusion needing arguing.

My response, here, would be the same as suggested there: that, as soon as one gets underneath the surface, it makes a tremendous difference to know how to work in detail through the constitutive complexities. This is a difference that will increasingly matter, given the ever-nearing prospects of constructing intelligent artifacts, and the almost inevitable imminence of understanding the detailed workings of the human brain. As an example, consider a juxtaposition of three points: (i) the referential opacity of implementation, discussed above at ·.......; (ii) the accompanying suggestion that implementation may only be able to be understood normatively,[16] not merely in terms of mechanical coarse-graining; and (iii) the sea change that has taken place, in recent decades of AI and cognitive science, towards neural modeling and machine networks, in place of explicit symbolic representation. If we are to have a theoretical analysis of these new networked architectures that has more more bite than “Lo! It is complex!”, especially an analysis that explains how full intentionality arises on such a base (if indeed it does), we will have to have a detailed understanding of the myriad technical issues being explored here: of how personal/subpersonal analyses relate, of how interpretation and semantical analysis plays at different levels, etc. of a sort that merely discursive analysis will never on its own be able to supply.

A57 ·45/2/-3:-1 This is the sort of purpose-specific contextual dependence that the fan calculus mentioned briefly in §... of the Introdution is aimed to deal with. Or to put it better, the aim of the fan calculus is not only to make room for such circumstantial dependence, and to handle such cascades of correspondence, but to be a theoretical framework in terms of which to *explain how ontology itself arises out of such circumstantially (and normatively) dependent “correlation*.”

[16] And not merely because all ontological delineation (as I would put it: all registration) is normative, but in the stronger sense described in §... of the Introduction, on logic.

A58 ·47/1/1:3 Cf. remarks both in the Introduction and in “Reflections on Reflection” (e.g., at ·…… and ……, respectively) about how this degree of semantic strictness made 2/3Lisp untenably baroque.

The issues are deeper than mere usability or “user-friendliness,” and illustrate an enduring theme of the entire 2/3Lisp experiment. It is traditional, in the surrounding intellectual context—especially in AI, logic, and philosophy of mind—to make a number of assumptions about the nature of computational systems that, on the face of it, do not seem especially contentious: that formal systems maintain or at least honour the semantics of their ingredient symbols, that one should not make gratuitous use/mention confusions, etc. On inspection, however, as has been discussed earlier, it turns out that in the vast majority of real systems these issues are not dealt with at all. In the design of 2Lisp and 3Lisp, in contrast, they were taken seriously, with the result was that these dialects were in some ways theoretically cleaner than their more commonly-used counterparts. By far the more important lesson, however, was to make in evident that there is a reason that these assumptions are so commonly ignored, in the thick of computational construction. When push comes to shove, the traditional assumptions simply do not work—they are either false, or rest on misunderstandings, or represent idealizations that fall apart in the face of real-world complexity.

A59 ·47/2/3:5 Cf. annotation A57, above; these are the sorts of insight that have led to the design of the fan calculus described in the Introduction (cf. ·:·····).

A60 ·49/1/5:-1 As suggested throughout these annotations, such analyses sometimes require making, and sometimes require *not* making, distinctions between and among some of the constituent types.

A61 ·50/n34 Cf. O3, though that book was not published until ten years after this paper.

A62 ·51/2/3:4 This assumption that the domain and co-domain each consist of a “predetermined collection of situations, objects, properties, and relations” is explicitly relaxed in §…, below.

A63 ·51/2/-4:-1 Ideological constructivists will object to this admittedly somewhat binaristic characterization, claiming that an object or state of affairs α corresponds to another β “only in the eye of beholder γ”—and so that correspondence should be analyzed along at least trinitarian lines. I am sympathetic to something in the background ontological intuition, but not to the framing. First, I want to distance myself from the "*only* in the eye of beholder" framing, even if (as I believe) observers are requisite to registration. Second, take the relations being studied here as *what beholders take to be the case*. For the purposes at hand, to register a beholder as a beholder accomplishes little. At best, it succeeds in making one level of semantic or objectifying ascent; it does nothing to extract itself from then occupying yet another beholder's view.

See also the parenthesized paragraph three paragraphs ahead (·53/1:54.0).

A64 ·52/1/6 I was prepared only to license ontological richness at this stage—as if the world consisted, a priori, of an infinitely rich plenum of objects from which we could choose. It was nevertheless clear, as emerges more and more towards the end of the paper, that I was well on my way to implicating human practices not only in the selection and identification of objects to be referred to as objects, but in the notion of objecthood itself—thereby sundering any clear ontology/epistemology divide, and denying that "being an object" is an intrinsic "property" of an object.

It was not until ten years after this paper was written, however, that *On the Origin of Objects* was published, in which I explicitly defend such an approach (cf. also "The Nonconceptual World" and "Representation and Registration," both in Volume II).

A65 ·52/1/8:9 One: the numeral '124.' Two: two numerals '124', both instances of the same numeral *type* (i.e., on a theory of numerals as tokens instead of types). Three: the numerals (as types) '1', '2', and '4.' Four: the *types* '1', '2', '3,' and '124,' Six: two '1's, two '2's, and two '4's. Eight: two instances, each, of the four types just identified. Etc.

A66 ·52/1/-5 This the first place in this paper where I use the term 'registration,' which I do not define until ten years later, in *On the Origin of Objects* (cf. annotation A64, above; as well as "The Nonconceptual World" and "Representation and Registration," both in Volume II).

A67 ·53/1/11 The case for correspondence may be "much less clear" (that for representation), but by the time of the publication of 03 I was convinced that, since registration of the states of affairs participating in the correspondence relation is itself relative to a registrar, then the argument for the third place in the relation might seem compel-

[17] Cf. §5 of the Introduction, especially ·.......

ling. Cf. annotation A63 (above), however, as well as the rest of this paragraph.

Today, I would say something like this: that for *some* purposes it may be important to "register" the registrar of the correspondence, but for other purposes not. The issue should be decided in a contextually-dependent way, by the theorist or analyst, not taken to be an absolute fact about correspondence itself.

A68 ·54/1/7:8 Whether it would be most useful to set up correspondences between the types themselves, or between instances or tokens of those types, would depend on the purposes for which the correspondence was being developed. And whether the objects in question were types or instances might not be clear, especially in mathematical cases (e.g., does "particular quadruple" four lines below (·52/1/11) refer to a particular quadruple *type*? It is not clear what "truth maker" would be relied on to answer any such question.

A69 ·55/0/4 The notion of absorption has proved especially useful, in the intervening years; and is used at various places throughout this volume. See for example annotations A… at ·……., A… at ·……., A… at ·……., etc. «…Redundant given next annotation? Sort this out…»

A70 ·55/0/5:8 Though the presentation here is extremely sketchy, some of the semi-technical notions identified on this page, perhaps especially including absorption, have subsequently proved very useful in both teaching and in research (and have been widely picked up by students). If it is possible for a property or relation to be absorbed, it is often high on criteria of usability to do so—and confusing not to. Typical examples include geometrical and formal properties—such as order, color, arrangement, etc. (For example, the injunction "CAPITALS in emails come across like shouting; stick with lowercase" is liable to be more successful than its opposite: "Capitals in emails come across like shouting; stick with LOWERCASE").

A71 ·55/-5:04 The fact that I did not define reification on objects betrayed the fact that I was not yet ready to embrace metaphysics and ontology full bore. Cf. *On the Origin of Objects*, by which point I would have been happy to do so.

A72 ·55/1/4:6 Physical keys in hotel lobby mailboxes may have been a dated example even in 1986. «…Is this said somewhere else as well?…»

A73 ·56/0/1:3 Cf. the richer discussion of compositionality in §… of the Introduction, esp. ·…….

A74 ·56/1/11:13 By "left and right identities (with respect to this algebra)" I of course did not mean that if a relation R_1 was classified, according to this typology, as iconic, then it itself would (at the object level) be a left or right identity. Rather, the point is a higher-order one: that if R_1 was classified as iconic, then however another relation R_2 was classified in this higher-order typology (e.g., as reifying), then $R_1 \cdot R_2$ and $R_2 \cdot R_1$ would be classified in the same way (as, for example, reifying).

The problem is that this is clearly false, given the typology as presented. If R_1 is iconic and R_2 absorbs left-to-right order, it does not follow that $R_1 \cdot R_2$ will necessarily absorb left-to-right order—since absorption is not entirely a higher-order property (it makes specific reference to object-level relations). The error illustrates what is anyway obvious: that this account of correspondence was more hope than theory.

A75 ·56/-1/-6:-4 Having *every* correspondence relation *always* visible—as suggested by the proposal as stated—would multiply the problems of fastidious brittleness that had already rendered 3Lisp effectively unusuable. It is in part because of the this that I take it as a constitutive requirement on the fan calculus that it allow such relations be circumstantially visible—so that only what is at point, in a given analytic situation, need be brought into explicit view.

A76 ·57/0/1:4 Cf. the "criterion of ultimate concreteness" in 03 (p. ...). «...move up...»

A77 ·58/1 Re this whole paragraph, cf. also annotation A25, above.

What makes the issue complex (and warrants the hedge "appears to be") is ambiguity over the phrase "extensional analyses." On the official story, an analysis—or more locally, a referring term—is extensional just in case the truth of the whole would be preserved by the substitution of another analysis or term with the same "extension" as the first. Thus the truth of a an extensional mathematical analysis of a function or relation may officially depend only on that analysis correctly identifying the extension—the set or ordered pairs—the constitute the function or relation in question. It is virtually never the case, however, that in practice we provide analyses whose epistemological work value depends only on their extensional correctness.

A78 ·59/-1/5:6 Whatever "computing a relation" means. Cf. «...».

A79 ·60/1/3:5 The point was not that a theory of correspondence would obviate semantical analysis. As I tried to make clear, the idea of correspondence was meant to be general, not focusing on intentional relations in particular. The theory was intended to aid, not replace, semantical analysis—including analysis of the semantic contribution of constitutive ingredients to the full significance of a complex sentence, thought, discourse, argument, etc.

The current point was just that anything like a simple binary distinction between the "intension" and "extension" of a sentence or term, or between "intensional" and "extensional" contexts, is likely to prove too simplistic to handle the complexities of real-world systems, and would be replaced, in a correspondence-theoretic frame, by facilities for registering rich cascades of parameterization, circumstantial dependence, varieties of types, etc.

By way of example, consider saying to an interlocutor who is facing you, "Can you more over here to the right?" As noted above, "on the right" can be seen as a one-, two-, three-, or four-place relation. The aspect of the significance of the phrase that the interlocutor must grasp, in order to respond appropriately, will likely need to make just *two* of the relata explicit; the third and fourth (which direction is "up," and which of the two symmetrical configurations we humans use to distinguish right from left) can be left implicit, because they are shared by speaker and hearer.

Does that mean that the "meaning" or "intension" of the phrase "to the right" makes two out of four possible parameters explicit? No; the point is to deny the coherence of the question. As programmers know well, complex situations often involve cascades of contextual parameterization, with different roles filled in at different stages—and different types of analysis require different numbers of these dependencies to be brought into view. And layers of contextual dependence are not the only issue. As we have seen in discussions of the kinds of example that motivate the fan calculus, there can also be ambiguity and richness in the variety of types of which the present concrete token or use can be analyzed as an instance (cf. the discussion of "mental indexicals" in annotation A48, above). Any attempt to (i) identify one of these types as the "correct" one, and (ii) to name or identify a small finite number of property-level distinctions and occasions of parameterization as most important, in order to generate a finite, named, labeled catalogue of types of semantic contribution (intension and extension, meaning and content, meaning and content *and character*,[18] "... meaning" vs. "...... meaning,"[19] and so on—will be too impoverished to do justice to real-world cases. By analogy, imagine trying to name, as a special class, functions whose recursive depth is 1, or 2, or 3, etc.—as if those formed analytically useful cases. What we need are frameworks to do justice to arbitrary layers of parameterization, cascades of finer and coarser types, etc.

A80 ·60/-1/5:7 That reference relations not only are not computed, but need not be (and in my view never are) *effective*, is perhaps the most important pervading theme in my entire understanding of computation and intentionality. Cf. "Representation and Registration" in Volume II. That makes it neither magic nor a priori; in the subsequent sentence, when I say that the relation between my reference to Bach and the long-dead composer just *is*, I do not mean to imply that it is not an achievement—of history, culture, etc.

The point is worth bringing up only because of the considerations brought forward in the discussion of "blanket mechanism" in the Introduction (cf. ·......). That non-effective (and thus, at least in a local sense, not obviously mechanical) relations could be theoretically critical seemingly challenges a large number of people's commitment to physicalism, mechanical explanation, etc. It is my firm belief, however, that neither computing nor intentional phenemena more generally can be understood except in full recognition of its overwhelming significance.

A81 ·61/2 Other distinctions that could be been listed here include (i) those among type, token, instance, and use; and (ii) between intension(al) and extension(al).

A82 ·61/3 By 'gratuitous artifacts' I mean in particular the non-signifying properties of models—such as the inflammation temperature of balsa models of airplanes, the order of the elements in each quintuple used to model a Turing machine, etc. While in some cases such properties are so obviously (literally) "insignificant" that they are unlikely to cause problems, it was an enduring view of Jon Barwise that they had distracted serious mathematical attention.

A83 ·61/-1/-5:-4 For example, object identity is not always preserved across implementation boundaries. Thus imagine an FKRL program that imple-

[18] Cf. Kaplan...

[19] Cf. Perry and Israel ...

mented the closed world assumption with respect to automobiles occupying spaces in a parking lot, by simply constructing a data structure (perhaps an array or matrix) isomorphic to the parking lot itself, and generating an unique object to locate in the spot in the data structure just in case a car occupied that spot in the lot. A clever implementation, however, such as that generated by an optimizing compiler, might engage in tricks to encode the matrix as a sparse array, so that no memory was actually allocated for each cell in the matrix. From the point of view of the FKRL program, the system would honor the closed world assumption; but from the point of view of the language in which FKRL was implemented, it would not.

A84 ·62/1:7 Note that, at least arguably, the referent of the term 'Margaritaville' differs in all seven cases, at least when any typographic details (italics, smudging, etc.) and attendant quotation marks are included. Yet to come up, *in advance*, with an ontological domain that includes all of the distinctions that would be required for semantical analysis not only of this full range of cases, but any conceivable coherent addition to them, would at a minimum lead to untenable pedantry, and in practice be impossible.

A85 ·63/1/1:4 Cf. the discussion in annotation A… to chapter …, regarding Jon Barwise' contention that the use of term models in logic almost invariably obscured rather than illuminated matter mattered about their structure. My predilections ran parallel to his, and in spite of the first sentence of the next paragraph—that "my point is not to indict this practice"—in fact I remain relatively staunchly opposed to their use *as a method of analyzing semantics*. They may have their use in proof-theoretic analyses, but a term model is not, in my view, an adequate approach to understanding what sentences and formulae *mean*.

A86 ·64/1/-8:-6 The point came up, in intellectual discussions at CSLI, in more than one context.

Regarding the study of situated language, first, our[20.5] insistence

[20] Available online at:
http://ageofsignificance.org/aos/en/aos-v1c0.html and
http://ageofsignificance.org/aos/en/aos-v1c0.pdf

[20.5] 'Our' refers not only to Jon Barwise and myself, but also to other senior members of the overall CSLI project (of which John Perry and Barbara Grosz were also principle investigators).

that language was context-dependent engendered a common reaction: that there was nothing new in the claim—that everyone already acknowledged that real-world day-to-day language is highly indexical and context-dependent. It was not our intent to deny this evident socio-intellectual truth, but rather to insist that, contrary to what others believed, the context-dependence of language was a *central phenomenon*, not a complication whose theoretical treatment could be added on, in a later stage, to a more basic theory formulated to deal with the context-independent case. The study of such context-independent systems as formal logic as "idealized" versions of language, we believed, was to miss the point.

Regarding situation theory, second, a similar sentiment was expressed: that one could of course study situations in set-theoretic terms, but that the structure of situations was the main event—and that to "reduce," "implement," or "model" them in set theory was distracting. To focus on them set-theoretically would lead theoretic attention away from the central issue of the intricate structures and relations between and among different kinds of situation.

A third example of such a theoretic inversion had to do with the relation between first, second, and third person pronouns—though in this case I found less support among my CSLI colleagues. It was during those years that I came to believe that to focus on third-person or "impersonal" reference, as if that constituted the core or basic or simplest referential phenomenon, and to view the intricacies of the first and second person cases as additional to or superimposed on top of that "more basic" case was similarly misguided. As is to detailed in my (forthcoming) "Who's on Third?," I believe the situation is approximately the opposite: with the first and second person cases being intentionally fundamental, and third-person reference being not only in some sense the most complex, but perhaps for that reason by far the least clear.

A87 ·65/0/-5:-1 ⟦Cf. the discussion of "direct semantics" in … (and refer to A33 in Clocks).⟧

A88 ·65/1/-8 There is a second issue, as regards the traditional demonstration of Turing machine equivalence, involving the identity of the machines in question. In order to show that machine $\mathcal{M}_2$ can "do the same thing" as another $\mathcal{M}_1$—or, more generally, to show that a universal machine $\mathcal{U}$ can do anything that any other machine $\mathcal{M}$ can do—one sets things up so that the emulating machine ($\mathcal{M}_2$ or $\mathcal{U}$) is given as input not only (i) an isomorph of the input that would have been given to the emulated machine ($\mathcal{M}_1$ or $\mathcal{M}$), but also (ii) an additional input, which is effectively a *program for the emulated machine* (i.e., instructions telling the emulating machine how to mimic or "implement" the operations of the emulated machine). What is "equivalent" to the emulated machine, therefore, is not strictly speaking the emulating machine on its own, but the emulating machine plus that program for emulating the emulated machine.

Though not normally given theoretical prominence, in my view these considerations of machine "boundary" and identity are far from theoretically innocent. Cf. the Introduction to the *Age of Significance*.[20] «…Say something about the motor theorem?…»

A89 ·70/1/-4:-1 I myself am such a "someone." I.e., I do not believe that positive and negative facts are on a metaphysical par.[21]

A90 ·72/0 Or so at least the authors of situation theory claimed (that situation theory deals with interpretation directly, not via a model). When we started the project I was a strong advocate of this form of "direct semantics"; as time proceeded I became less and less convinced that situation theory would actually suffice as a direct account. Cf. annotations «…».

A91 ·72/1/-3:-1 E.g., cf. 1970s and 1980s work of Jose Meseguer and Joseph Goguen.

A92 ·72/-1/-3:72//1 Regarding some additional complexities, cf. annotations A48 and A79, above.

A93 ·73/0/6 'Like' merely in the sense of liking someone. This paper was written before the use of 'like' as a direct thought quotation operator had taken root in informal English ("I'm, like, *Who is this guy?*") .

A94 ·74/1 It is here that the discussion opens up into the sorts of metaphysical consideration that were addressed in *On the Origin of Objects*, and that have permeated my work ever since. As such, this paragraph marks the opening of a critical stage in the overall intellectual development of this work.

[21] When I was a child, it was carefully explained to me that (a) through (d) of the following were acceptable, but that (e) was not:

a) Chicago is in Illinois.
b) It is a fact that Chicago is in Illinois.
c) Chicago is either in Michigan or in Illinois.
d) It is either a fact that Chicago is in Michigan or it is a fact that Chicago is in Illinois.
e) It is fact that Chicago is either in Michigan or in Illinois.

A95 ·74/-1 This view is what I called "metaphysical monism" in *On the Origin of Objects*—intended to be fully compatible with (in fact, to provide grounding for) various radical forms of *ontological pluralism*.

A96 ·75/-2/1:2 Again, this paragraph contains initial glimmerings of what I now take to be the necessary foundations of a coherent metaphysics.

A97 ·76/1/1:2 "To the extent that theorist's langauge and representation overlap on registration scheme"— «...explain; theorist and *subject matter*»

A98 ·77/0/4:5 I was inclined to suppose this, at the time—that representation and registration were mutually implicating—but I did not really know what it (or I) meant. Cf. however "Rehabilitating Representation", in Volume II.

A99 ·77//-2 I am not an essentialist, but the use of the term 'essence' was more innocent. I just meant that I had not yet dug very deep.

A100 ·78/1/-8 Cf. the figure at the end of *On the Origin of Objects,* on p. ...... (though what is indicated as towards the right, here, is in that figure depicted as towards the bottom).

A101 ·79/0 By 'grounded' I was making a stab at what in *On the Origin of Objects* I refer to as 'immanent induction'—a fundamentally important issue on which I have still said far too little.