LANGUAGE, COMPUTATION, AND REALITY

Donald Perlis

TR95

May 1981
ABSTRACT

The main theme of this work is the interplay of assertion and meaning, or quotation and un-quotation, in reasoning entities. This is motivated largely by analysis of the notion of possibility in several contexts, most specifically in relation to resource-limited computational models of belief and inference, as well as in philosophy of science.

A first-order treatment of quotation and un-quotation is given that allows broad and paradox-free expression of syntax and semantics. It is argued that this makes unnecessary the usual hierarchical constructions for notions such as default reasoning, theory subsumption, concepts, beliefs, and self-reference, and indeed that even greater expressive power is achieved than in those treatments, with reduced complexity of notation.

This is then applied to a model of belief and inference in which focus of attention is a key element. Effort is made to isolate certain automatic inferences apparently part of the very meaning of propositional beliefs, and then base more sophisticated thinking on these.

Finally, some thoughts are presented on how resource-limited computation may bear on the notion of possibility in foundations of physics and modal logic.
ACKNOWLEDGMENTS

I would like to express my appreciation for the advice, argument, and other forms of academic and sociable assistance I received in the development and elaboration of the ideas in my Ph.D. thesis, of which the present work is a very slightly extended version.

In particular, to the members of my dissertation committee, James Allen, Gerard Emch, Jerry Feldman, and Henry Kyburg, go my thanks for their encouragement and commentary. Most of all, to my advisor, James Allen, I express gratitude for and amazement at his patience in going over in minute detail my frequently highly disorganized drafts of this work.

Finally, my thanks to several other graduate students and professors at the University of Rochester for endless and endlessly illuminating discussion of various sections and/or various versions of this work, most especially Alan Frisch, Andy Haas, and Dan Russell.

CONTENTS

I. OVERVIEW

II. TRUTH AND SYNTAX
   1. Introduction
   2. Syntax
   3. Truth
   4. Formal results

III. THE LOGIC OF BELIEF AND THOUGHT
   1. Introduction
   2. Previous approaches to belief and inference
   3. Theoretical restrictions
   4. Representation and logic
   5. Active belief models
   6. General hypothesis and model

IV. GETTING ON IN THE WORLD
   1. Introduction
   2. Axioms
   3. Inference and focus
   4. Trial runs

V. CONCLUSIONS AND FUTURE WORK

APPENDIX A: PROOFS
APPENDIX B: POSSIBILITY REVISITED
APPENDIX C: IMPLEMENTATION
BIBLIOGRAPHY
CHAPTER I

OVERVIEW (OR. IS POSSIBILITY SEMANTICAL?)

Language is a kind of discrete representation of reality. We have in mind natural language as conceived in artificial intelligence, that is, as manipulation of strings (bits of the world). This serves the purpose of facilitating planning by providing us with discrete computable representations (beliefs) of what is possible in a given domain. So viewed, beliefs are part of the very world they purport to describe. Herein we find the main theme of this thesis: the interplay of assertion and meaning, syntax and truth, quotation and un-quotation. It is our contention that a proper understanding of beliefs involves this kind of self-referential ability.

Another observation derives from the above view, namely that language is tied to our imperfect and changing theories about the world, and indeed semantics and science grow and are clarified together. Semantics cannot be understood independently of (a model of) the rest of the world. Thus (at present) language is 'dirty' rather than clear and precise. In particular, linguistic usage often seems to mean something when on analysis it may fail to do so or at least may be ambiguous. This is a trivial point, but from it we conclude that we need suitably 'dirty' formalisms to deal realistically with natural language. In fact one of our central contentions is that the seductive elegance of modal logic has in some respects had a limiting effect on semantics, especially for possibility, causality, and belief. Our general concern then is to elucidate a computational view of beliefs via the notion of possibility.

Possibility is a key concept in any planning system. One must decide 'what could be imagined to occur'. These possibilities then help shape our beliefs (strings of symbols) about external causal relationships which form our models of reality. The role of semantics is to explicate the connection between the strings and the models, via the concepts of possibility, causality, and belief. We will see that understanding the various senses of possibility depends on widely different aspects of our world view. We contend that there are three distinct concepts here that will hinge on problematic issues in cognitive psychology, logic, and philosophy of science. All three are part of everyday usage, and all three are important in planning intelligent
behavior.

Let us examine the sentences

1. It is possible Skylab will fall.
2. It is possible the hostages are spies.
3. It is possible a student will graduate with no math.

Sentence 1 seems to be an acknowledgement of a certain capability of reality. Sentence 2 can be interpreted as an admission that the speaker knows of nothing that clearly shows the hostages not to be spies, and thus refers to the speaker's own beliefs. Sentence 3 apparently reflects a fact about school graduation requirements. These have very different semantic characters, 2 being the only one explicitly making a claim about one's state of knowledge, and 3 the only one whose truth is formally determined by consistency with objective rules. It is more difficult to convey the sense of 1, except perhaps negatively: we do not mean we are ignorant of enough physics to decide the matter - rather, we are claiming a 'fact' about the way reality works, and in a seemingly stronger sense than a mere consistency with formal laws. Roughly speaking, type 1 possibility invokes a principle of reality, type 2 a state of relative (self-) knowledge, and type 3 the existence of a consistency proof.

It is essential to know what is meant by Possible(X), in order to make effective use of such statements. It has been generally assumed that modal logic (via Kripke models) provides an adequate semantics for possibility. We claim this is not the case, and that the issues hidden there bear on key problems in natural language, to wit, belief and knowledge, self-reference, and causality. Moreover, the computational (discrete and resource-limited) paradigm significantly contributes to plausible solutions.

Thus if sentence 2 is read modally, i.e., that in some imagined world 'the' hostages are spies, we learn nothing at all except that we can imagine what we like; the same applies to sentence 1. There is no set of axioms with which to measure possible worlds, unless we can axiomatize precisely a person's beliefs or the universe's truths; both are tasks far beyond our capabilities and in any case not at all what we intend these sentences to be measured by. On the other hand, sentence 3 can be read as a plain assertion of the consistency of graduation requirements with taking no math, i.e., the existence of a model of just that. Now, if the speaker is commenting on the laxity of graduation requirements, it is necessary for 'possible' to be interpreted in this way, i.e., as a formal matter capable of objective demonstration: the requirements do not logically entail the taking of math. Sentence 2, by contrast, refers implicitly to one's state of knowledge, and must be so
understood; its truth depends on who makes the statement, and we may know the hostages are not spies and yet make use of 2, to assess the speaker's grasp of the situation and likely response to varying situations. Finally, sentence 1 must be understood as calling attention to something that may actually occur and might conceivably be prevented; we might be able to stop Skylab from failing, but we cannot stop the hostages from being spies although we may be able to contend with that eventuality; and we cannot stop the graduation rules from being consistent with no math unless we change the rules.

These several senses of possibility are often identified with the modal (formal, possible worlds) one. Yet despite the heavy reliance on the possible-worlds semantics, almost no serious work has been done to explicate the nature of these 'worlds' (as opposed to their formal properties). The general view seems to be that they are just formal models, and indeed proof-theoretic results in modal logic depend on just this interpretation. However, this very attitude seems to reduce the validity of such models regarding any question of reality, and it is this that we must address in attempting to catch the proper senses of possibility.

Let us call types 1, 2, and 3, factual, epistemic, and formal possibility, respectively. It is easy to show that properly distinguishing them is essential for effective use of language. The tendency to collapse these three uses of possibility into one, namely that of formal consistency, has been particularly troublesome regarding belief. For if we regard sentences consistent with our beliefs as precisely those we don't believe false, then we are led to the conclusion that we believe all formal consequences of our beliefs. This strange and clearly false supposition actually follows from the conflation of epistemic and formal possibility. Let us examine this. Writing $\text{Poss}(x)$ and $\text{epi Poss}(x)$ for epistemic and formal possibility, respectively, we have the definitions

$$\text{Poss}(x) \iff (\neg x \notin \text{Belief Set})$$
$$\text{epi Poss}(x) \iff \text{Consis}(x + \text{Belief Set})$$

Here '−' stands for negation, and $\text{Consis}(x + S)$ means the adjunction of the element $x$ to the set $S$ results in a consistent set of statements.

Pinning down the nature of the belief set of a reasoning entity is part of the aim of Chapter III. But for now we need no more to make our argument. For if
$B, B, \ldots, B$ are beliefs and if $C$ is a logical consequence of these, then not-$C$ would not be formally possible:

- Poss $(-C)$.

Now passing to epistemic possibility we would find

- Poss $(-C)$

which is defined as

$$-C \in \text{BeliefSet}$$

i.e., $-C$ is believed! Assuming standard collapsing of double negations leads to our promised result: any logical consequence $C$ of beliefs $B, \ldots, B$ is itself a belief. This then allows us to identify all formally equivalent statements as far as belief goes, an unfortunately popular approach in semantic studies.

A major concern of ours then is the proper delineation of epistemic possibility, i.e., a psychologically and computationally plausible model of belief (and inference, which forms an essential part of the growth and change of our beliefs). In particular this must allow for belief of a sentence without the simultaneous belief of all equivalent sentences, and yet be closed under sufficient inference rules to afford content to the beliefs.

This will involve treating a preliminary theme, namely, the differentiation of syntactically distinct but logically equivalent statements, descriptions, etc., so that we can make statements about our beliefs. This will give our formalism vastly greater expressive power than is usual, though keeping it free of contradiction will require care. We claim that natural language is unavoidably self-referential in that we can and often do speak or reason about our capabilities including our means of delineating and communicating, and this itself must be done within those very means. We will argue in fact that this is the rule rather the exception.

Our FIRST PROBLEM (Chapter II) then is that of devising a suitable formal language that is able to express its own syntax and semantics. This has been recognized as desirable for purposes of natural language, but due to the familiar difficulties of self-referential paradoxes little has been done along these lines. Recent work of (Gilmore 1974) and (Kripke 1975) will be helpful here, especially in giving a computational outlook on meaning that will fit in well with our goals regarding belief and inference.

Chapter III treats our SECOND PROBLEM, namely, the above-mentioned delineation of epistemic possibility. This is placed in the context of a rather broad belief and inference system, with sample applications given in Chapter...
IV.

Factual possibility also can be seen as growing out of the issue of belief and inference, for we often try to make our beliefs conform to reality by scientific reasoning. This involves both deductive and inductive inference, and can be carried to arbitrarily great extremes, so that no a priori formal scheme can codify to what extent individuals will avail themselves of this. Nonetheless, when we speak of factual possibilities we intend that they be taken as real in a sense beyond that of our relative knowledge. Appendix B contains some discussion toward a plausible semantics for factual possibility. This will be of a brief, highly informal, and speculative nature, not worked out in any technical detail.

Thus the material falls into two major parts: (1) a suitably realistic formalism and (2) epistemic possibility, belief, and inference. We will now describe in some detail the particular issues to be treated in these areas.

*   *   *

First, we wish to devise a precise and broad formal calculus (language and inference schemes) for the study of natural language, in a psychologically plausible manner. This in particular must deal with punning, a virtually universal phenomenon in language and yet rarely acknowledged in formal studies. We refer to the fact that usually part of what an utterance is to convey is the fact that it is being used to convey something. We don't only get the 'message', we also get the syntactic medium and use it for various ends, such as in framing a reply in similar words. Thus 'John is here' is more than an equivalent of John's presence, for our attention is called to that presence and to John's name as well, and in a readily grasped manner. More striking, 'John's height is impressive, isn't it? Well over six feet for sure!' puns on the expression 'John's height' as both a number and itself as a string description that tags John and the idea of height. In fact, the latter seems the more fundamental notion: we first realize we are focusing on John and his height, and then consider using this numerically. This 'tag' information is contained in the verbal message and not an abstract proposition.

(Another example is the facetious subtitle of this essay, which to be properly read must be seen both as a propositional question AND as a play on (the words of) the title of (Putnam 1977).)

The same kind of analysis can be applied to so-called 'non-existent' entities, e.g., the golden mountain. Here we have just punned on 'the golden mountain' by stating the description as a tag that seems to refer to something but we quickly realize it does not. We do not need to postulate a world of non-existent objects to correspond to such descriptions: we take them for what they are, syntactic
entities that can be used as if they were referential in some contexts when properly agreed upon, but otherwise they remain as dangling tags. Thus we needn't attribute meaning to every grammatical string of words, and this will be a key point of our formalism.

However, if we then agree that language usage tends to be orthographic in that we frequently pun on syntax and reference, and so must preserve the mode of expression in semantic analysis, we find that our formalism must admit the explicit representation of syntax as an object formally expressible. This of course is true, and desirable, in natural languages, e.g., we discuss English almost all the time in English. We designate as 'syntaxal' any formalism capable of expressing its own syntax. This leaves us with the delicate problem of deciding when a description does refer to something and when a sentence has a definite meaning. This is delicate simply because there seems no rule that would single out such descriptions and sentences in advance, due to the familiar paradoxes of self-reference that are expressible in a syntaxal system. This seems to have been a key factor in the avoidance of string-oriented approaches to the so-called propositional attitudes such as belief; however the alternative modal efforts have not generally been at all in line with intuitive usage here. (A refreshing counterexample is (Eberle 1974) which we shall note later.)

It turns out though that the consistency of our formalism can nonetheless be assured by using recent ideas in (Gilmore 1974) and (Kripke 1975). This will involve viewing meaning as having to 'prove' itself in some algorithmic sense, e.g., by unpacking a string to its 'source meanings' if possible; such a test may loop forever in which case the string remains as mere syntax. Thus in terms of our beginning remarks, representations of the world can't be guaranteed perfect at all times, and correct usage consists of carefully tracing the line between sense and nonsense.

To briefly recapitulate: the strings are the message, quite literally, being the actual input from which we must sort out meanings. Part of the meaning consists in knowing that the given strings are the ones used (this is so trivial and automatic that we easily lose sight of it). And it is clear that in many cases the literal message is essential. (This can even include intonation and stress. (Parker-Rhodes 1978) has provided an interesting semantics for spoken sentences that nicely handles certain conversational episodes seemingly out of reach of more customary formalisms.)

Our view of language as part of reality and tied to our theories of reality, leads us to identify symbols with parts of the world, and conversely to regard any (identifiable)
part of the world as potentially allowable for use as a symbol for something or other.

This means that quotation and un-quotat Ion are essential for us, since we need to pass from a symbol to its meaning and back, in order to address this relationship within language itself. The concept of truth will be seen to play the role of un-quotat Ion, which we will develop in a first-order setting despite the familiar negative result on truth definitions. As a consequence, Leibniz' Rule of substitutlvlty, a standard tool of first-order logiC, will remain valid for us. If a term appears in a context where its use depends on syntax, then it will of necessity lie in a quotation context putting it out of the range of substitution. Language by its very nature involves symbolism, yet symbols are used both to stand for a denotee and to call attention to themselves, so that substitution corresponds precisely to contexts in which we don't care about how something is represented; this is precisely the intended sense of variables in first-order predicate calculi.

Subordinate questions, troublesome In propositional treatments, here become straightforward. For instance, 'John knows what Bill said' in its two standard meanings is expressed by

\[
(1) (\exists x) (\text{SAID}(\text{Bill}, x) \land \text{KNOWS}(\text{John}, x))
\]

\[
(2) (\exists x) (\text{STATED}(\text{Bill}, x) \land \text{KNOWS}(\text{John}, \text{Concat}(\text{STATED}(\text{Bill}, x), x)))
\]

Here STATED is used for directly quoting Bill, while SAID corresponds to any 'reasonable' paraphrase. This latter notion of paraphrase is highly non-trivial and will be addressed briefly when we analyze belief. 'Concat' is a function symbol that the formalism will use for concatenation of strings.

A first-order treatment of quotation and un-quotat Ion will be given that allows broad and paradox-free expression of syntax and semantics. It is argued that this makes unnecessary the usual hierarchical constructions for notions such as default reasoning, theory subsumption, concepts, beliefs, and self-reference, and indeed that even greater expressive power is achieved than in those treatments, with reduced complexity of notation.

Belief, growth (and change) of knowledge, and paraphrase, form the central issues deriving from epistemic possibility. We will argue that human inference is expression-oriented and syntactically computed, and that our beliefs are of expressions encoded in some internal 'language' in our heads (see (Fodor 1975) and (Moore-Hendrix..."
Inference provides a kind of housekeeping for these expressions, and is strictly bounded by resource limitations. We will make deliberate use of an analogy with paging in operating systems, to aid us in devising suitable and testable models. Only a modest amount of information is 'in focus' at a given time, and a paging strategy such as LRU (least recently used) seems one in rough accord with naive introspection of thought processes. Knowing when to call a new page (or expression in memory) into focus is a vastly more tricky matter that we shall address only peripherally; in general such would involve the whole range of associations present in human knowledge. However, we will need at least a rudimentary model of long- and short-term memory, where focus is identified with the latter.

The inference rules will be intended to model what processing is done automatically when focus is updated. This is not goal-directed; we seek to characterize unprompted inference, beliefs that automatically arise from given information without being presented as matters at issue. The point of making this distinction is that given a goal or question, an individual can spend arbitrary amounts of time and ingenuity (such as running a theorem-prover, event) to examine various lines of inference; it seems hopeless ever to give a characterization of what one or another individual might do regarding a particular goal in a particular time span. But it is more plausible that certain elementary deductions are either always carried out (even unconsciously) or are so easily done that for purposes of natural language communication each participant can safely assume the others will follow reasoning by filling in these small steps. Furthermore, such a weak inference scheme presumably is all we have recourse to in everyday situations when confronted by a bewildering variety of facts: we ignore familiar repetitions and have no time or inclination to think about the rest most of the time, except when a fact automatically jars us as not right - it seems to contradict something we believe. It is this sense that we seek to capture here.

A desired consequence of this should be that such processing will terminate in a finite (and hopefully short) time; not all logically valid conclusions will be drawn, although we expect certain conclusions to be virtually guaranteed, such as recognizing suitably blatant contradictions. (As a result beliefs can easily be contradictory, but not blatantly.) This is precisely what a speaker must do to reach his audience: neither belabor the trivial nor skip the non-trivial, and so it seems a significant aspect of correct use of natural language. Our approach here is similar in spirit to that of (Eberle 1974), except that he aims at broadly delineating a framework for any level of inferential competence rather than
characterizing a particular level.

The same effort above will bear on 'reasonable' paraphrases. For we want the paraphrase to be one that will be recognized as equivalent to the original by essentially all listeners without special prompting. (E.g., the subtitle of this Overview puns on the title of (Putnam 1977) but in a non-automatic manner.) Also, epistemic possibility can now be defined as 'possible as far as one knows', i.e., 'not believed false'; this of course is the automatic-inferential closure of what one already believes, and is then time-dependent, as is natural and in fact essential in planning, where one is constantly making new observations and updating information as the world changes. The growth and change of knowledge will then be a theme in this section, which we hope to partially characterize to afford comparison with a similar notion of the next section. Finally, we will develop axioms for belief sentences so that each 'believer' will, according to the formalism, believe the automatic conclusions of her/his other beliefs - this is an application of the earlier model of memory and inference.

CHAPTER II

TRUTH AND SYNTAX

1. Introduction

Much work in knowledge representation can be seen as dealing with a trade-off between more expressive power and more tractable notation (e.g., (Creary 1979), (Doyle 1980), (Winograd 1980)). In particular, (unsorted) first-order logic has been widely regarded as very nice to use but too weak for many notions generally treated in what are essentially hierarchical (sorted or higher-order) formulations, e.g., frames, modalities, belief revision, concepts, defaults, quotation, self-reference, theory subsumption. We argue here that, given a suitable treatment of negation inside quotation, this verdict is unnecessary, and in fact much greater expressive power is then available in first-order logic than in the usual languages relied upon for these concerns.

- 18 -
In particular, variables in such hierarchical languages have a limited range of application, so that fully general schemata such as $P \land Q \rightarrow P$ aren't readily expressed in one axiom. This has repercussions in AI specifically regarding the above notions, for they relate syntax to semantics. One has to look at the representations themselves and state their connection to their meanings. A hierarchical syntax prevents one from making this fully general, which latter would be not only philosophically desirable but much simpler notationally. Thus if I claim that I have no belief in a god, then the present belief I am asserting is a belief about (all) my beliefs and therefore also about itself. Putting a hierarchy on beliefs removes the possibility of expressing such relationships. (Similar observations apply to meta-knowledge, concepts, etc.)

The obviously desirable situation then is to allow the representational system to get its hands on the syntactic elements themselves, with free-wheeling quotation and un-quotation mechanisms, much as we allow ourselves in natural languages. Then it would appear that the whole world is at our disposal, for these syntactic elements literally are the medium of discourse. For instance, we can then refer directly to axioms as terms, or to theories (sets of axioms), or to a formula's being true, or being believed, or no longer being believed, or to a term's standing for some object, all in a first-order setting.

Now, this is not so surprising. The big trouble is that then we are confronted with paradox, even from the apparently very simple and natural schema

$\text{True}(\text{'A'}) \leftrightarrow A$

where $A$ is a formula with no free variables. (Here 'True' is our un-quotation device.) This is essentially the comprehension problem of set theory, with the attendant paradoxes of self-reference; we won't detail for the moment how such arise in the present context beyond mentioning that it is possible to construct a sentence $R$ such that

$\text{True}(\text{'R'}) \leftrightarrow \neg R$

is provable, where '$\neg$' indicates negation.

Hierarchical approaches, as in (Russell 1908), (Creary 1979), and (Doyle 1980), do seem to remove paradox, but as we have indicated at the expense not only of cumbersome notation but also of significantly reduced expressive power. It is even unclear how these approaches would express such a simple notion as "John has a false belief." For this seems to require a notion of truth (or un-quotation) that is independent of any hierarchical level. Moreover, the intuitively straightforward approach is simplicity itself:

$(\exists x) \text{Believe}(\text{John},x) \land \neg \text{True}(x))$

We will present a way out of this that saves the simple semantic and notational features of (unsorted) first-order logic yet has the power to express virtually any of the
standard concepts of AI, without paradox or strain on intuition.

The trick follows (Gilmore 1974) and (Kripke 1975). In place of the earlier inconsistent schema, we simply posit
\[
\text{True('A')} \iff (\text{A}^*)
\]
where the * operator replaces each connective occurrence of the form \(-\text{True('...')}\) in A by \(\text{True('-(...)'})\). In most cases, this reduces immediately to the former schema, and when \(-\text{True}\) does appear in A, the new schema is still trivial to apply and intuitively sensible. For example, \(\text{True('-\text{True('1=2')}')}\) is equivalent according to the above schema to \((-\text{True('1=2')}\))*, which is just \(\text{True('-(1=2)'})\). (This in turn is equivalent to \(-(1=2)\).) Actually a slightly different quotation mechanism will be used, namely Hollerith quotation, since then there is no scoping ambiguity, but this isn't necessary for purposes of illustration here.

We will content ourselves in the rest of the introduction with a very brief illustration of the power of this approach.

It is often useful to have generic or prototypical cases of classes or concepts (see (Winograd 1980)). This is frequently related to non-monotonic logic in which generic or default assumptions may be overridden. But if we can refer directly to beliefs as terms, there is no need to depart from classical first-order formulations. Namely, we write them in quoted fashion, e.g., inside Belief predicates:

\[
\text{Believe(me,'Canaries are yellow')}\]

Then whatever reasoning people do (correct or incorrect) is analyzable by us as scientists in our first-order language. This is especially convincing when we introduce time parameters into belief assertions, for then we capture the process of revision itself, and non-monotonicity doesn't arise in our logic. Beliefs simply have temporal ranges outside of which they are inactive or untrustworthy, this too being described by suitable axioms.

This requires hard work, but it is distinct from finding a representational medium. This is largely the so-called control issue, and is dealt with to some extent in Chapters III and IV. It means we can concentrate on the facts we wish to express regarding thought and action, and not be so concerned with novel mechanisms for expressing them. For in reality what we wish to say about intelligence has straightforward expression; the difficulty is discovering what it is.

Thus compunctions about copying free-swinging notation fairly directly from natural language, with its self-reference and relation-objectification (quotation), have kept us tied to overly weak and cumbersome representations, ever since Bertrand Russell's discovery of
paradox in Frege's theory of sets. Given our quotation mechanism, it is hard to see what serious restrictions are placed on knowledge representation by the requirement of first-order formalism. For in any language amenable to computation we are restricted to discrete notations, and even the most complex concepts rely on object- and relation-terms.

As we have already suggested, truth can be regarded as the cement between sentences and their meanings, since we often indicate a relation between a sentence s and the proposition it expresses by saying True('s') ↔ s. Thus truth goes hand in hand with the expression of syntax. It is central to our approach that all syntactic elements be expressible, as is the case in natural language. Only then can we carry out broad and effective (verbal) reasoning about our own (verbal) reasoning, such as seeing that some utterances are ambiguous or that some lines of thought aren't practical. This is precisely what makes an ambitious representation of truth and syntax so problematic: self-reference becomes possible, and with it, paradox. In formal systems this is closely related to undecidable sentences, whereas informal reasoning can often decide such sentences effectively. One aim here will be to clarify this apparent gap between formal and informal reasoning, especially as found in languages for artificial intelligence systems. (See (Ballard & Brown) and (Nilsson 1980) for discussions of self-reference in artificial intelligence.)

Anticipating a complicated set of issues ahead, we should ask why a truth predicate is useful at all. Why not just use mere assertion, e.g., x and not True('x')? We can do this of course, and if we then choose to write True('F') as a rewriting of F, this in itself is harmless. What we lose is really the use of variables ranging over strings so that substitution of one formula in another becomes possible. If this is not required, all is fine, and we can discuss belief, etc., with modal logic. E.g., we can write Believe A → A where Believe is a modal operator, thus avoiding both True and syntactic variables, but then we lose the ability to quantify over sentences. Moreover, when such quantification is allowed, modalities no longer seem to be of any great advantage.

We shall examine here various formal ingredients in paradoxes of self-reference, with an eye to providing a treatment of truth that is as natural and powerful as we can make it without inconsistency. We shall not begin with a criterion for a satisfactory truth definition, but rather hope to tailor it to what proves feasible as our study proceeds. Our first requirement is a broad symbolic language in which to work. This is presented in ensuing paragraphs. It is seen that the language there is what we shall call syntaxal, i.e., syntactic elements are
represented as terms and there are operators for concatenation and substitution. In particular, any formalism allowing Gödelization is syntaxal, although we shall not need the recursive nature of the latter case.

Very imprecisely, our main result is that, self-reference notwithstanding, as in 'This sentence is false,' the truth of a sentence can be identified with the meaning of that sentence precisely when that meaning doesn't depend on whether we designate it as true. Furthermore, this can be formalized in a consistent first-order theory, building on work of (Gilmore 1974) and (Kripke 1975).

2. Syntax

As hinted above, a variety of considerations lead to the desirability of including syntax as formally expressible within our language. Indeed, we eventually would like to model as well as possible the actual context of natural language, including its tentative and developing character.

Thus, symbolism is seen as a learned agreement, and as such it can be altered. It is argued in (Fodor 1975) that there is an unlearned private language 'behind' public (natural) language; we shall not contest this view, and in fact it is quite in line with our development here. However, it is the public language with which we are primarily concerned, whose syntax is freely discussed within that same language.

To make our approach fairly general, we could use a context primitive, so that in effect any utterance could in suitable circumstances have any given meaning, as in the case of a code-book. However, there is also an implied standard context, without which normal communication would be impossible. The generality afforded by context will be important for us in illustrating the difference between punning (where syntax plays a semantic role) and what we shall call 'glibness', i.e., straight substitutional symbolism. This is especially the case in regard to belief, which involves private contexts. There is a strong and misleading appearance of glibness in everyday speech, whereby we rush through utterances while thinking in images 'behind' them. That is, we tend to imagine that our words are a straightforward encoding of those images. But in fact more careful introspection and examination of particular sentences reveals that the sentences contain much additional information, mainly about ourselves, our concerns, and our direction of thought, and also may fail to contain some
essential information of the images. The keys here for us are the 'concerns' and 'directions.' Although we often don't notice it, we communicate a great deal of this in what we say, and even expect others to pick it up (arguments are often the result of failures of this mechanism). There is a clear-cut formal distinction here which is captured by first-order predicate calculi. Namely, a glib context, in which we suppose an expression to have a clear-cut meaning independent of its syntactic form, can be identified with that of a first-order variable or term: we simply use it as if it 'were' its meaning. This is the sense of Leibniz' Rule of Substitution.

Passing to a concrete example, viz. the sentence 'John is here,' we note that if the listener did not know John's name when the utterance was made, then this contains the significant information that someone who is present is named 'John.' We could formalize the utterance as

$$(\exists x) (\text{Here}(x) \& \text{Name}(x, \text{'John'})$$)

but this loses the probable glibness intended by the speaker: 'John' wasn't intended to convey information about names at all, just as we don't intend to define 'apple' when saying 'have an apple' while professing an apple (except in deliberate teaching situations). It is sometimes with surprise we learn that we have revealed a great deal we hadn't intended since we thought we were speaking directly to the issue and not intruding additional comment via our particular choice of words. By the same token, reading improves one's vocabulary even without using a dictionary, since context usually is an excellent indicator of meaning. It appears that children learn language this way, without asking the meaning of every word. If we view all words as defined by convention and context, then there is no fixed natural language, except as a learning process, and language acquisition and language processing should become nearly the same study. All this we take as further evidence that syntax should be part and parcel of semantic analysis, rather than weeded out in a separate parsing phase to leave a ghostly residue of propositions. Our claim is that we must focus on represented knowledge, as for instance string formulas which are the appropriate structure for first-order languages.

Thus when doing arithmetic reasoning, we allow ourselves to interchange 771 + 229 and 1000 at will, using the former as if it 'were' the latter. However, if that were the case then we'd never bother to use the complicated sum when merely 1000 would do. So 771 + 229 is not merely a name for 1000; 771 + 229 is an expression that simultaneously tells us to consider 771 and 229, and to consider the operation of forming from them a sum. In other words, in arithmetic we allow glib use of terms, even though
our reasoning is not correctly represented this way. What is preserved under 'glib' substitution is not meaning but validity, and if we care only about a numerical or logical 'answer' then indeed we are on safe ground.

We claim that some glibness is essential in all communication. Although we may pause to allow a given utterance to reverberate and take on a punning character of its own, this is in a sense a further message which then in turn is glib until it too is commented on in a further pause or other indicator. If we specifically want to call attention to the medium, we usually do it with quotes or other emphasis. Thus it is in the proper and normal usage of language that we have certain glib syntactic elements, such as variables. We either don't pay attention to them beyond noting their reference, or we specifically call attention to them via quotes. With this understanding, we are free to indulge in arbitrary conventions and inventions, as long as they are duly noted.

We note that various problematic utterances, viz., 'the golden mountain,' are very unlikely to be made glibly, i.e., naively assuming a direct reference to a well-defined entity. Indeed, the very recognition that there is something odd about such expressions depends on recognizing their parts. Perhaps more striking is the expression 'the round square' which, although in usual modal non-syntactic treatments it would reduce to nullity, should be recognizable as conflicting on points of roundness and squareness. Thus we should be able to say that round squares are impossible because they need to be round and square at the same time. In particular, round squares will have to be round, and NOT because of their nonexistence. One can argue that round squares are also green and colorless by vacuity, but such reasoning is both unintuitive and uninformative. Indeed the observation that if x is both round and square (where we here interpret these as strictly incompatible geometric properties, rather than a way of indicating something roundish-squarish) then any result is derivable, is perfectly sound. But that very reasoning is lost if we cannot pick out the words 'round' and 'square' from a sentence about something being round and square. Thus the ability to see why a description is null seems to require maintenance of sentential units rather than a monolithic propositional truth.

Additional comments apply to utterances such as 'sky' or 'Santa Claus' where it is difficult to point to a physical referent even though we use these expressions as if they so referred. One approach here is to say that we switch back and forth between mental contexts, or 'stories,' often unconsciously. Alternatively, we might regard Santa Claus as a complex set of associations triggered by the words 'Santa Claus.' This will have some relevance in later
Let us now proceed to some technical details. We will describe a first-order language $L$ allowing full expression of its own syntax. It is our intention to view language and symbolism as part of reality rather than something superimposed on reality; this will bear not only on the present topic but also on belief and other issues. Our approach is similar in spirit but broader in scope than that found in (Creary 1979). We select a domain or world $W$ of objects of discourse. This may include any recognizable aspect of reality, either specific to a particular topic or (in principle) an exhaustive accounting of reality at some suitable level of scientific discourse. We will not get deeply involved in issues of whether an entity may have distinct instances (e.g., the letter 'x')—this is an area which seems not to bear in a crucial way on our treatment. We do however wish to insist that $W$ contain a subcollection $A$ of units, essentially our alphabet for writing expressions, which is to include a first-order vocabulary: quantifiers, constants $a,b,c,\ldots$, variables $x,y,z,\ldots$, connectives $\rightarrow,\cdot,\&,\ldots$, function symbols $f,g,h,\ldots$, and predicate symbols $P,Q,R,\ldots$. We also insist that any finite number of elements of $A$ may be considered as forming a string (in any order) which itself is an entity in $W$: $e\ldots e$ is in $W$ if $e,\ldots,e$ are in $A$. Finally, we insist that among the units of $A$ are the decimal digits $0$ through $9$.

Now we can introduce a fundamental requirement, corresponding to quotation. Essentially, we want utterances to be quotable as they are, i.e., as the very strings of characters that form them, marked only externally to delineate them as strings rather than as their intended meanings. This is perhaps not of theoretical interest, but it certainly corresponds to quotation in natural language and makes for greater readability than certain other methods. For instance, putting slashes before blanks and before other slashes, as in LISP, is very unappealing. SNOBOL is closer to our treatment, but even there a special mechanism is needed for quoting quotation marks. In English, it is common to italicize or use a tone of emphasis to indicate when a quotation begins and ends, and we will employ an analogous device which however is completely describable and 'quotable' in the same way (unlike italics). Namely, for any utterance formed of the characters $e\ldots e$, we write

$$n: e\ldots e$$

$1\ldots n$

as a name for the utterance (string). The number $n$ will always be written in its standard decimal representation in
such contexts. We require each such name to be a constant of the language. Note that here $n$ gives the number of units out of which $e \ldots e$ is formed, $n:e \ldots e$ itself being formed of $n+2$ units (if $n$ is a single digit, otherwise more). It is essential that no unit be a string (of more than one unit) so that we can give an unambiguous reading to any string. Thus $n:e \ldots e$ is not a unit, although it is a constant. As an example note that 3 gives the length of the string $x+y$, and $3:x+y$ is its name. (The FORTRAN aficionado will recognize our quotation device as essentially that of the Hollerith statement.)

The reason we introduce $n:e \ldots e$ instead of simply using $e \ldots e$ in all contexts, is that often $e \ldots e$ itself have a complex meaning beyond its mere string character. For instance, $771 + 229$ will be named by $7: 771 + 229$.

Then the former can be used as a term, i.e., glibly, at least in ordinary arithmetical contexts, while the latter allows us to unpack the term into its syntactic pieces, and thereby recognize that a sum is indicated, etc. Note that $7: 771 + 229$ has a second reading, at least at first glance: as a string of nine symbols! Nothing seems to tell us whether to read the initial '7:' as a length marker or simply the first of several symbols. But in fact these are the same reading! The name $n:e \ldots e$ is precisely the string given, of $n+2$ or more symbols; but it is not a unit.

Our notation allows us to distinguish neatly now between glib and quoted contexts: $3:4+1$ names the string formed of 4, +, and 1, while $4+1$ is simply 5. To be extra cautious, we should perhaps write $1:4$, $1:+$, and $1:1$; however, since units always have length 1, this is redundant, and we will make the convention that $1:a = a$ for all units a other than variables. As a consequence, each such unit a names itself, essentially 'telling by showing.'

Lest the reader think we could get away with the more familiar device of straightforward quoting, we observe that quotes are ambiguous. For instance,

$$A("4") = A("5") = A("6")$$

has two possible readings as a first-order formula, one with string expression

$$4") = A("5$$

as argument to the first instance of $A$, and one with
as argument to the second instance. The problem is that quotes don't uniquely pin down their scopes, unless we do not allow quotes in string expressions; however, this would violate our requirement of complete syntax expressibility. So we use length prefixes as in n:e ... e. The same applies to the quasi-quotes of Quine with reserved variables within quotation contexts; this amounts to prohibiting the complete expression of syntax since a reserved variable can never be quoted in such a context. Of course the EFFECT of quasi-quotations can be had by a suitable (and fully intuitive) combination of concatenations and ordinary variables.

There is a well-known quotation technique for representing syntax in formal languages, namely Gödelization. We are not using this approach here, convenient as it is for proof-theoretic purposes, because we shall not need the powerful arithmetic structure it rests on, and also because our simplified version is vastly more readable and more in line with intuitive usage of quotes. (In (Quine 1946) it is shown that arithmetic can be recaptured from concatenation, however.)

Function and predicate symbols are (strings of) units. Among the function symbols we require are Ln, Concat, Sub, and Entry. (Actually Ln and Entry are sufficient by themselves, as we will see.) Ln(x) denotes the number of units in the string represented by x. Thus Ln(4) = 1, Ln(1:4) = 1, Ln(3:1:4) = Ln(3:1+4) = 3, and Ln(n:e ... e ) = n, while Ln(1+4) = Ln(5) = 1. Note that the argument of Ln in Ln(1:4) is the string denoted or named by 1:4, i.e., the string of the one unit, 4. It is as if we had written Ln("4"). Note that we write 1+4 as a term which is formed with the function symbol +, hence 1+4 is interpreted as the value of the corresponding function (addition), just as if we had instead written 1:5 or simply 5, whereas Ln(3:1+4) = 3, since 3:1+4 names the syntactic object 1+4.

Confusion is easy here since it isn't clear whether 4 as a symbol is distinct from 4 as a number. Essentially this is a 'de dicto - de re' issue, in which 4 can characterize some number (e.g., the successor of the first odd prime) in a de re reading of the symbol in its standard reference, or simply name the graphical structure of the written character "4" in its (de dicto) instantiation without any semantical reference in terms of its usual sense. Of course, this is a distortion of the customary version of these terms in which a general substantive is used (such as 'the president'), but we claim the basic issue
is the same. One can imagine other symbols being used for \(4\) (the number), or "4" (the character) being used for some other reference. But then we must postulate a 'real' number 4 behind the various names for it, one which upholds its four-ness as an essential (not contingent) property: the very idea of four-hood. So why not select one such symbol in universal use (4) and treat it as this de dicto entity? Whether or not one wants to take this as a satisfactory definition of four, it seems perfectly adequate in terms of service: the object of writing, 4, as well as the other number symbols, form a collection with a well-recognized ordering starting with 1, then 2, etc., so we might as well use these as 'the' numbers. Then \(1:4 = 4\) becomes perfectly sound, and also \(1:4 = 1+3\), since 1+3 is used as a term formed with the function symbol + and more properly written \(+(1,3)\). However, 1+3 has no de dicto reading, except possibly as the result of the operation of adding 1 and 3, which however will always turn out to be 4, i.e., the de re meaning as well. This is always the case with first-order terms in the way that they are intended: we can see them via punning, but their use formally is glib. To indicate a different intent in 1+3 and 4, we should write \(3:1+3\) and \(1:4\) (or just 4) and then pull apart \(3:1+3\) to capture its internal sense. Similarly, \(2 = 1:2\) but \(3:1:2\) is not equal to \(1:2\).

We note that LISP uses much the same convention for numbers (constant units): both 2 and '2 have the same evaluation, namely, 2 itself. This does not mean the strings 2 and '2 are indistinguishable, but simply that to distinguish them we must refer to them by name rather than by glib utterance. We will depart from LISP though in a more iconoclastic reading of the above paragraph, regarding numbers greater than 9. For instance, we write twelve as the string formed of 1 and 2, i.e., we regard the concatenated 1 and 2 as BEING the number twelve, as well as being a name for twelve. In LISP this also holds in a sense: '12 = 12. But the difference is that 12 and '12 are not recognized by LISP as two-character strings (the apparent decimal representation is only for reader convenience). To properly refer to a sequence of the units 1 and 2 in LISP, we would normally write '(1 2), which is a list and NOT evaluated to twelve. Still, our language shall make such an identification, e.g., between 2:12 and 12. In effect, 2:12 and 'twelve' and 'XII' all are names for the same object, 12, which also names itself. The advantage we claim here is that, first of all, this is already in tacit worldwide usage, and second, we then do away with an infinitude of units (one for each number) which no one ever writes or speaks or ponders directly anyway.
To forestall an objection: if $2:12 = 12$, then perhaps we would find $2:12 = 2:2:12$ from simple substitution of $2:12$ for $12$ in $2:12$. But this assumes '2:' operates as a function symbol, i.e., that the appearance of $12$ in $2:12$ is as a term, and it is not. The whole expression $2:12$ is a single term, indeed a constant. The proper way to refer to the appending of $12$ to the character string '2:' is via the function symbol Concat, and then no problem occurs, as we shall shortly see.

We have indicated the sense of the symbol $\text{Ln}$, at least sketchily. Now we turn to other primitives and relate them to $\text{Ln}$. Note that since $\text{Ln}$ can take any term as argument, then intuitively any object at all in our formalism will have a length; this simply means that fundamental notions or entities will be constant symbols (hence of length 1). So to completely characterize any object in our formalism, it is enough to give the units in each position in that object (as a string). For this purpose we have the primitive $\text{Entry}$, taking two arguments, both necessarily strings: $\text{Entry}(x,y)$ is a term whose value is the unit in the $x$-th position in the string represented by $y$. Of course, here we intend $x$ to be a number; if it is not then the term will still appear in well-formed formulas but will be of no interest. Thus $\text{Entry}$ and $\text{Ln}$ are clearly related since both are tied to the enumerating of units in strings. To make this precise, we use Concat, which is a function symbol of two arguments also: $\text{Concat}(x,y)$. This denotes the string formed by concatenating $x$ and $y$. However, we will also allow the function symbol Concat to take an arbitrary number of arguments in the obvious convention, namely, $\text{Concat}(x_1,\ldots,x_n) = \text{Concat}(\text{Concat}(x_1,x_2,\ldots,x_n))$. The following postulates are clear from the above intuitive definitions. Note that we have full expressibility of syntax, including numbers: $3:1:1$ names $1:1$, $1:2$ names $2$, and indeed ANY string can be written and prefixed by a length numeral to form a term.

(1) $\text{Ln}(\text{Concat}(x,y)) = \text{Ln}(x) + \text{Ln}(y)$

(2) $\text{Entry}(k,\text{Concat}(x,y)) = \text{Entry}(k,x)$ if $k \leq \text{Ln}(x)$

(3) $\text{Entry}(k + \text{Ln}(x),\text{Concat}(x,y)) = \text{Entry}(k,y)$

(4) $x = y$ iff $(\text{Ln}(x) = \text{Ln}(y) \& \&(k)(k \leq \text{Ln}(x) \rightarrow \text{Entry}(k,x) = \text{Entry}(k,y)))$

Here we cannot simply define 'non-existent' entries to be uniformly equal to some default value, say $d$, since this amounts to saying that a string $e \ldots e$ really is infinite: $1 \ldots 1 \ldots e \ldots e dddddddd\ldots$, and we have no way of knowing which if any of these $d$'s are intended as genuine except by already
knowing the length $L_n$. Thus a notion of ordering is needed for values of $L_n$, as well as a sum. The sum can be very simple, using only the addition table on digits 0 through 9, and a carry rule.

Returning to the earlier issue of terms and function symbols in names, as in $2:12 = 12$, we observe

(5) $\text{concat}(2,:,12) = 4:2:12$, and

(6) $\text{Concat}(2,:,2:12) = 4:2:12$.

These are correct by our conventions, as indeed they must be if we are allowed Leibniz' Rule of Substitution. In line (6) above, the first instance of $2:12$ is interpreted as the string it names, which is precisely $12$. For us, every (closed) term is read as a string-name and then interpreted as that string. (Note that $L_n(2:12) = 2$, while $L_n(4:2:12) = 4$. It is far more natural simply to write $L_n(12) = 2$, however, which also shows $L_n$ to operate in part as a logarithm (base 10).)

We point out that there is no ambiguity in the reading of terms despite our double naming for non-variable units. That is, for any function or predicate symbol $f$, there can be at most one correct reading of $f(t_1, \ldots, t_n)$. This is so since each term $t_i$ must either be a single unit, a sequence of digits, or a sequence of digits followed by a '::' and the appropriate number of further units. Thus if no digit is scanned, we have a unit term; if digits are scanned with no following '::' then we have an integer term; and in the remaining case, we have a (Hollerith) name as a term.

A syntaxal language then is one exhibiting the basic features we have been describing, to wit, a distinct constant (Hollerith) term for every sequence of symbols, and function symbols for length and entry. Concatenation and equality can be defined in terms of these (essentially as in (1), (2), (3), and (4) above), and substitution can as well (this will be used in Appendix A). In effect, both concatenation and substitution depend on an implicit postulate of string formation, that we can adjoin any unit to the end of any string thereby forming a longer string; this also implicitly contains the ingredients for ordering and sum of lengths.

3. Truth

Now we come to truth, the central semantical issue. We wish to give a connection between a string and its meaning, which we shall do with the primitive $Tr$. We will say, as a
first approximation, that a string is true iff its intended meaning holds. Let us call this the presumptive concept of truth, since it assumes there is a prior intelligible sense to truth (or meaning) that automatically comes with each sentence. Such a view means that sentences referring to truth or meaning have to be 'unpacked,' and this opens the door to possible circularity, as we shall see. This will be a key point in our criticism and redefinition. The presumptive view formally then is:

\[ (7) \text{Tr}(n: e \ldots e) \rightarrow e \ldots e \]

\[ (8) \text{Tr}(n: e \ldots e) \leftarrow e \ldots e \]

Traditional problems with this approach include the famous sentence 'This sentence is false.' How such sentences arise in a formal setting, and what can be said about them regarding truth, will be our principal concern in much of what follows.

We assume meaning already to be clear in ordinary contexts, in which we make some utterance "x" to get across some meaning; the presumption is that the meaning doesn't depend on the means of expressing it; later we may comment on our manner of expression. In such cases truth is indeed very simple, once a domain is clearly specified. This is not to say that determining meaning from utterances is simple, and in fact this is a fundamental research area in natural language; but it is not our present concern. Rather we are concerned with problematic cases in which meaning depends on the very utterance in question and/or its relation to other utterances, in a way that may vitiate naive decisions about its truth.

In (Kripke 1975) this problem was addressed, and impressive gains were made, to the effect that within a given domain we can describe meaning or holding or truth in a consistent and intuitive manner. However, since he assigns no meaning to -Tr(x), his approach does not allow certain formulas to be seen as valid which we intuitively realize to be so, and furthermore there is no excluded middle in his semantics. Thus Kripke has proposed a variation on Tarskian truth to get at a broader view of meaning in a model. He comments that his method can be extended to modal systems. Our goal, on the other hand, is to put all the 'truth semantics' in the formalism and then interpret it with Tarski's standard semantics. This mechanism, including essentially unrestricted quotation and un-quotation, makes unnecessary any modal (or higher-order) formulations, however, since these are readily re-written via predicates over strings. standard semantics. We borrow ideas of (Gilmore 1974) for set theory, to accomplish this.
Of course we should restrict (7)-(8) to (well-formed) formulas $e_1 \ldots e_n$ so that (7)-(8) is itself well-formed. However, there will still be many difficulties, and we will modify and qualify (7)-(8) in several steps, so that eventually we will be able to establish a provably consistent schema for truth. In (7) can be seen the key concept of 'groundedness' or 'priority,' i.e., supposing a prior meaning of $x$ BEFORE ascribing meaning to $\text{Tr}(x)$. This will appear repeatedly in the sequel. Modifying Gilmore, we consider a positive formula to be one in which $\text{Tr}$ is not in the scope of (an odd number of) negation signs, and we will see that this is intimately related to the above priority condition (7). More precisely, call $A$ positive if $\text{Tr}$ is not in the scope of negation in the result of passing negation signs in $A$ through to predicate letters, following the usual valid rules for this regarding quantifiers and connectives. (It is important for these purposes that conditionals $A \rightarrow B$ be written as $B \vee \neg A$.) Again borrowing from Gilmore, let $A^*$ be the result of passing $\neg$ through $\text{Tr}$'s as well, so that $(\neg \text{Tr}(x))^*$ is $\text{Tr}(\neg x)$. Then $A \leftrightarrow A^*$ in general only when (7)-(8) holds. Of course, if $A$ is positive, then $A \leftrightarrow A^*$.

Before calling $x$ 'true' we had better see what $x$ means. In effect we are saying that first we have sentences, next we look at their meanings, and only THEN can we consider whether to regard a given sentence as 'true' or not. Once a sentence $x$ has been found to have a meaning that is the case, we then normally wish to record this fact formally by a further sentence, $\text{Tr}(x)$. Finding, as we shall, that in some cases this is not possible without formal inconsistency, shouldn't make us think that $x$ doesn't hold, or even that it isn't 'true,' but quite simply that the sentence that states that $x$ is true conflicts formally with other sentences. We want to be able to say $x$ is true, to assert $x$ in a formalism, to say $\text{Tr}(x)$ is contradictory, and therefore to assert $\neg \text{Tr}(x)$ also in the formalism. Roughly, the idea is that $\text{Tr}(x)$ should represent a decision made AFTER $x$ has been established and so $\neg \text{Tr}(x)$ merely means no such decision has been taken, although $x$ may still be the case.

We can amplify on this as follows: Writing in a grammatically ill-formed manner for convenience of illustration, the 'formula' $\text{Tr}(x) \rightarrow x$ can be wishfully interpreted as saying that if $\text{Tr}(x)$ holds, it can only be because $x$ already held before it was labelled with $\text{Tr}$, and that labelling itself doesn't alter the holding of $x$, i.e., $x$ persists in its meaning and correctness with or without $\text{Tr}(x)$. Positive formulas will turn out to persist in this sense. Roughly, the reason for this is that a positive formula, once true, makes no claims about non-labelability which could undo its own truth if we decide to label it with
Tr. On the other hand, when meaning depends on the method of recording an expression, then that very recording can cause a shift in the outcome, and naive intuition becomes inappropriate. So formulas $x$ that express something about their own labelability with $\text{Tr}$, are of a suspect nature, and conversely, if $x$ holds AND $x$ is consistent with $\text{Tr}(x)$ then it seems justified to so label $x$. This will appear often in the sequel as motivation. We also see then that it is (8) that presents conceptual difficulties, what we might call the 'presumptive fallacy,' for it is (8) that says given $x$ then '$x$' must already be granted true, i.e., not on the prior basis of $x$. It is 'presumed' a priori that such a conclusion is always warranted.

The point though isn't that truth is trickier than mere 'holding,' but rather that meaning in general isn't well-specified. Thus while we may accept $A \lor \neg A$ in general, to the effect that in any given interpretation or momentary criteria of meanings either $A$ will satisfy the criteria or it will not, still this outcome can vary from one such context to another, and may even vary as a result of using those criteria! This is precisely what happens when meaning itself is part of the formula in question. Attempts to single out formulas with or without ambiguous meanings run afoul of the same confusion, since ambiguity of meaning is itself capable of self referral. It is possible to isolate an 'absolute' fragment of meaningfulness, as was shown by (Kripke 1975), but beyond that we fall again into the presumptive fallacy by choosing arbitrarily one branch of many equally valid (or invalid) ones. Here we will eventually show that this can be formalized without giving up excluded middle and full first-order logic. We do lose the inference of $\text{Tr}(A \lor \neg A)$ from $A \lor \neg A$, but this is because now $\text{Tr}$ will mean there is no meaning shift within the argument from one labelling to another. (Note this again points up a distinction between the mere outcome (true-false) of a proposition, and its sense, for if $A$ can shift its meaning, then so does $A \lor \neg A$ even though no record of this shows in the propositional outcome.)

To further emphasize that 'truth' isn't the real issue so much as clarity or non-variability of meaning, we point out that the same 'shifting' of word sense as we speak occurs even by the apparently innocuous affirmation of a sentence by stating it and then saying 'yes,' or the reverse: 'I went to town, yes,' or 'Yes, I went to town.' This can be expressed as a rule: $(x \rightarrow \text{Yes, } x)$ where $x$ is any sentence. Now we can consider the possibility of a sentence referring to its own affirmation with 'yes': 'Yes, this sentence isn't begun or preceded by affirmation with "yes."' All this can be seen essentially as a matter of good housekeeping, getting our meanings straight before making rash claims. But it is non-trivial technically to see how to accomplish this in a general way.
Looking ahead, we consider the possibility that some formula $F$ may formally refer to itself or even to its own labelability, in such a way that

$$F \leftrightarrow \neg \text{Tr}(F).$$

Then it will be impossible to check $F$ before deciding $\text{Tr}(F)$, so that in fact $F$ is not labelable by the above strictures, i.e., $F$ is the case. Thus we have the peculiar situation of concluding both $F$ and $\neg \text{Tr}(F)$. Clearly this must be broached in a more precise manner, and much of our work here will center on just such a task, including a formal version of Russell's heterological paradox.

In recognizing a non-trivial distinction between $x$ and $\text{Tr}(x)$, we create implicitly several types of negation. That is, $\neg x$, $\text{Tr}(\neg x)$, and $\neg \text{Tr}(x)$ all say something about $x$ that opposes at least a very strong affirmation of $x$ that could bring $\text{Tr}(x)$ in its wake. But $\neg x$ denies $x$ directly, and $\text{Tr}(\neg x)$ goes even further by saying that denial itself can be codified in the formalism; however, $\neg \text{Tr}(x)$ merely denies that $x$ can be so codified. Thus in first-order logic we will have $x \lor \neg x$, $x \lor \neg \text{Tr}(x)$, and $\text{Tr}(x) \lor \neg \text{Tr}(x)$, but not $x \lor \text{Tr}(\neg x)$, nor $\text{Tr}(x) \lor \text{Tr}(\neg x)$.

Negation is a key concept for us, since when it is combined with $\text{Tr}$ we get 'lack of decision' as a kind of decision itself: $\neg \text{Tr}(x)$ and $\neg \text{Tr}(\neg x)$ say we are not allowed to label $x$ and $\neg x$ with $\text{Tr}$, respectively, not necessarily because $x$ itself is undecided, but because the syntactic labelling produces problems. This is the price we pay for allowing complete syntax expressibility in our language. Now, as long as $x$ can be decided independently of $\text{Tr}(x)$, this doesn't arise. But given $\neg \text{Tr}(s)$, the formula $\text{Tr}(\neg \text{Tr}(s))$ is another matter, since $\neg \text{Tr}(s)$ could conceivably imply that it itself is not $\text{Tr}$-labelable; we shall in fact eventually see this can happen. On the other hand, if we are able to find $\text{Tr}(\neg s)$, we have a much stronger base to work with since we should be able to require $\text{Tr}(\neg s) \rightarrow \neg s$ and $\neg s \rightarrow \neg \text{Tr}(s)$. Thus, whether or not we can 'justifiably' pass '-' through $\text{Tr}$ in a given instance, seems to be an issue on which hinges the meaning of $\text{Tr}$. We will develop this idea now.

If $x$ doesn't refer to labels or to non-labelability in particular, then it seems plausible that $x$ won't undo its own labelling by its affirmation. Thus we should avoid subformulas '$\neg \text{Tr}(s)$' within $x$ if we seek to identify $\text{Tr}(x)$ and $x$. So, we should be able to require $\text{Tr}(x) \leftrightarrow x$ for positive $x$. This is a good start. But what about other formulas $x$? We don't wish to remain completely silent in those cases, for $\neg \text{Tr}(x)$ should have meaning in any case.
Now, since the presumptive view identifies Tr(x) and x, and since our wish is to use Tr(x) when it can reasonably have that intended sense, it seems that this may provide its own interpretation of Tr. That is, we can regard Tr(x) as saying not only whatever x says, but also that within x presumption reigns: Tr(x) $\leftrightarrow$ x, Tr(-s) $\leftrightarrow$ -s $\leftrightarrow$ -Tr(s) for subformulas s of x, etc. In other words, x is intuitively safe.

From a slightly different angle, the general issue of when Tr(x) $\leftrightarrow$ x is to hold, is essentially the same as the issue of when Tr(-s) $\leftrightarrow$ -Tr(s) is to hold. For if the latter cases are known, then from Tr(-s) $\rightarrow$ -s we get the former cases. Now, in general we have Tr(-s) $\rightarrow$ -s $\rightarrow$ -Tr(s), showing Tr(-s) to be stronger than -Tr(s). If we regard Tr(-s) as saying that '-' can be switched past 'Tr' then we arrive at another formal contention, via x*. For recall x* moves '-' past Tr, so that (-Tr(s))* is Tr(-s). Thus it appears plausible that truth of x persists through labelling precisely when x* $\leftrightarrow$ x. If so then we will also have Tr(x) $\leftrightarrow$ x* in such cases, and in general Tr(x) can be taken to mean x & (x* $\leftrightarrow$ x) which reduces simply to x* since we will require x* $\rightarrow$ x. That is, we now have (as a plausible line for investigation) a formal definition of Tr(x) as simply x*! Whether this can hold up to the test of consistency and naturalness remains to be seen. (Below we will give a partial criterion for naturalness via the concept of 'distributivity. ')

Our tentative view then is that Tr(x) is very strong and resolves questions about x presumptively. However, we will still have the other cases to interpret, i.e., finding s and -Tr(s). But our interpretation will be that s is true and Tr(s) is not true, i.e., s is not consistently Tr-labelable since its sense depends on the non-switchability of '-' within s. This peculiar flavor of truth depending on how we write assertions is a necessary consequence of our requirement of full syntax expressibility, since we then let in meanings based on representation. However, although this means we must use care in making assertions, it does not mean we lose the ability to note formally the details of which formulas hold, which are labelable, etc.

We now take a broader look at various possibilities and desirabilities regarding the concept of truth, with an eye to the devising of a version of truth based on persistence.

The first trouble we note is that variables inside quoted strings do not have their usual sense. The formula

\[ (9) \quad \text{Tr} ( \ 3:x=y \ ) \quad \leftrightarrow \quad x=y \]

combines a closed formula, \( \text{Tr} ( \ 3:x=y \ ) \), and an open formula, \( x=y \). This means that (9) will be true in the usual
first-order semantics only if every substitution instance of (9) for FREE occurrences of \( x \) and \( y \) is true, and in particular then

\[
\text{Tr} (3; x = y) \iff y = y
\]

will be true, although this is not at all what we wish to be the case. We conclude that open formulas need a special treatment regarding \( \text{Tr} \). Indeed, we really want the effect of symbol substitution, e.g., the result of substituting a string \( s \) in the string \( x = y \) for all free occurrences of the variable \( x \). This will be crucial in future development. We shall eventually see however that even for closed formulas there is difficulty with presumptive truth as we already suspect from earlier discussion regarding the schema

\[
(10) \quad \text{Tr} (n; e \ldots e) \iff e \ldots e \quad 1 \ n 1 \ n
\]

To abbreviate, we may write \( \text{Tr}(\text{'F'}) \iff \text{'F'} \) though here \( \text{'F'} \) is not an adequate quotation device, and \( \text{'F'} \) is not a term.

To make reading easier, we introduce a special notation for quotes, which can be defined within our formalism. We define \( \text{quote}(x) \) to be \( \text{Concat}(\text{Ln}(x), :, x) \). Thus

\[
\text{quote}(n; e \ldots e) = \quad 1 \ n
\]

where here \( k \) is the decimal integer value of \( \text{Ln}(n) + n + 1 \). This may seem peculiar, but recall that \( \text{Concat} \) returns a string, which has as NAME itself preceded by its length; also recall that the argument \( n \) really has \( \text{Ln}(n) \) symbols. So \( \text{quote}(\text{\textit{x}}) \) gives the (standard) name of the (standard) name for the string \( \text{x} \), and \( \text{Ln}(\text{\textit{quote}(\text{\textit{x}})}) = \text{Ln}(\text{\textit{Ln}(\text{\textit{x}})}) + \text{Ln}(\text{\textit{x}}) + 1 \).

Now we can extend (10) above to more general formulas via

\[
(11) \quad \text{Concat}(2; \text{Tr}, 1; (\text{\textit{quote}(\text{\textit{x}})}, 1;), 1; \iff , x)
\]

where we consider the sign \( \iff \) to be a single symbol of one unit, as we also do for \( \iff \) and \( \iff \). Note that for each \( x = n; e \ldots e \), where \( e \ldots e \) is a formula, (11) actually names a formula in our string notation. It would be more readable to write something like \( \text{Tr}(\text{x}) \iff \text{x} \), or \( \text{Tr}(\text{'x'}) \iff \text{x} \); however, these are not well-formed formulas, even if \( \text{x} \) is a well-formed formula. Still, we shall often use these notations for convenience, and the understood formula is that named by the appropriate construction with \( \text{Concat} \) and \( \text{quote} \). A further such notation will be \( \text{Tr}(\text{\textit{\textit{-x}}}) \) for \( \text{Tr}(\text{\textit{\textit{Concat}(\text{-x}, \text{x})}}) \).
These considerations suggest a tentative treatment of Tr that serves for open as well as closed formulas. We have only to note that Concat is a function symbol and hence can take variable arguments which will not be evaluated unless an explicit substitution is made. Thus Concat(x,=,y) is not 3:x=y, and it becomes more reasonable to consider then

\[ \text{Tr} (\text{Concat} (\text{quote}(x),=,\text{quote}(y))) \iff x=y. \]

This says that for any x and any y, i.e., for any terms (string-names or functions) we may assign to x and y, the new string stating the equality of the two named strings is itself true just when those strings are indeed equal.

Thus we are led to the next stage in our (still tentative) approach to truth:

\[ (12) \]

\[ \begin{align*}
\text{Tr} (\text{Concat}(b,\ldots,b)) &\iff A, \text{ where } A = a_1 \ldots a_n \text{ is an atomic formula and } b_i = a_i \text{ unless } a_1 \text{ is a (free) variable, in which case } b_i = \text{quote}(a_1); \\
\text{Tr}(\text{Concat}(x,\&,y)) &\iff \text{Tr}(x) \& \text{Tr}(y); \\
\text{Tr}(\text{Concat}(x,2:\text{or},y)) &\iff \text{Tr}(x) \text{ or } \text{Tr}(y); \\
\text{Tr}(\text{Concat}(\neg,x)) &\iff \neg\text{Tr}(x); \\
\text{Tr}(\text{Concat}(\exists:(x),b,\ldots,b)) &\iff (\exists x)\text{Tr}(\text{Concat}(c,\ldots,c)) \quad \text{where } c_i = b_i \text{ unless } b_i \text{ is } 1:x, \text{ in which case } c_i = x; \text{ this device produces the effect of substitution for } x \text{ in a formula, and will be used again in Appendix A for a technical construction}; \\
\text{Tr}(\text{Concat}(3:(x),b,\ldots,b)) &\iff (x)\text{Tr}(\text{Concat}(c,\ldots,c)) \quad \text{with again the above condition on } b_i \text{ and } c_i.
\end{align*} \]

Note that in a sense we are not defining truth of strings, but of string types: Concat(x,=,y) for instance is not a particular string and we are not assigning truth to 3:x=y so much as to any instance of x=y. Note also that (12) gives us a technique for 'unpacking' formulas with Tr into more primitive ones, at least in many cases. This will figure in a prominent way as we proceed.

The converse of the fourth implication of (12) would violate our expectations of groundedness or positiveness, and will be left out. Let us call a formal theory
Our main finding (based on work of (Gilmore 1974) and (Kripke 1975)) is that CONSISTENT DISTRIBUTIVE THEORIES EXIST, i.e., a formal truth predicate can be made to conform to the six rules in (12). In fact, we will find that this is the case in any theory having the axioms

\( (13) \quad \text{Tr}(F) \leftrightarrow F^* \) for all formulas \( F \), and

\( (14) \quad (x) (\text{Tr}(x) \land \text{Tr}(\neg x)) \) .

Furthermore, ANY consistent first-order theory \( T \) can be given a truth predicate \( \text{Tr} \) and extended to a (consistent first-order) truth system \( \text{GK}(T) \) by the adjunction of the above axioms. It will then be the case that in \( \text{GK}(T) \) we have as much 'truth presumption' as possible, in the sense that

\( \text{Tr}(x) \rightarrow x \)

holds, but

\( x \rightarrow \text{Tr}(x) \)

does not, nor even is the rule \( x / \text{Tr}(x) \) (infer \( \text{Tr}(x) \) from \( x \)) permissible.

These ideas can be related to mechanisms for reasoning as found in artificial intelligence systems and languages. Both (13) and (14) above play key roles here. First, (13) amounts to saying that \( \text{Tr}(x) \) acquires a definite meaning when repeated reduction to \( x^* \) leads to an already known result not involving \( \text{Tr} \). Cases in which such reduction leads nowhere, as we have hinted at via the concept of non-grounded formulas, can be viewed as the looping of a non-terminating algorithm. Thus in, say, PLANNER (Hewitt 1971), a procedure call may fail to return due to looping, and then there is lack of a result rather than a positive or negative result. Now comes an interesting point: often \( \text{WE} \) can see that a call will loop, or that a formula can't be labelled true or false, even though no such result may be expressible or derivable in many languages or systems for reasoning (or making plans). But in the truth theories we have devised, such results are expressible and derivable, and indeed (14) itself is of this sort. In fact, (14) is a kind of consistency statement, usually regarded as unprovable within the system in question. Of course, (14) isn't really a statement of consistency of the given system since it doesn't refer formally to the theory and its axioms or proof mechanism. Still, we shall refer to (14) as 'Consis' since there is a formal similarity to the familiar arithmetical consistency statement \( (x) (\text{Thm}(x) \land \text{Thm}(\neg x)) \) of recursive theories, and since Consis indeed does say our
usage of Tr is self-consistent.

There is a technical subtlety occurring here. Whereas in any recursive theory we can formulate the predicate Thm that applies to (Gödel numbers of) theorems, and whereas if the theory is designed with an underlying standard model in mind then we know Thm(x) → x is true, still this will not be a theorem of the theory. Similarly, we may know a procedure will never halt, even though PLANNER may not be able to tell us this. We can make this argument more precise, simply by the earlier observation that Tr(x) → x is a theorem in our systems but Thm(x) → x is not, and similarly that Cons is a theorem but (x)(Thm(x) & Thm(¬x)) is not. Our point is that we reason about halting or consistency from general principles and not from procedural code or Gödelizations of axioms. That is, we carry out reasoning about a general notion of provability or truth or halting, in the vein of Tr rather than Thm. It has been suggested (in various guises by various authors -- see Cherniavsky 1978) for a recent effort) that this shows human reason to be more powerful than, or at least different from, mechanical proof devices. But it appears that the difference is between various types of systems (formalized or not) rather than between formal and human (informal) reasoning. This bears amplification.

Basically, we have a choice of either a 'universal' but never fully specified sense of Tr (or halting, etc.), or of a fully specified but (provably) incomplete one (e.g. Thm). Of course, either way the concept will be incomplete, but the advantages of the former are that, first of all, we can take the concept (or symbolic usage) to be complete in our intended interpretation; secondly, we can even incorporate this as an axiom of our theory without risking inconsistency; and finally, any particular specification of a Thm-type predicate for truth can be viewed as contained within the more general sense of Tr. It is precisely the lack of full specification of Tr arithmetically that allows us to avoid the usual diagonal arguments leading to paradox. It is this same lack of specification that allows us to observe results outside a given system; in a sense Tr 'grows' whenever we see some new way of thinking.

Thus Thm is relative to a particular theory, while Tr is not. When we assert our reasoning to be correct or consistent, either we refer to a designated and formal type of reasoning not including the present assertions, or we speak generally about our expectations of thought, without knowing precisely what they are: we can reason about reason coherently in the manner of Tr. But nothing can provide us with a complete specification of our reasoning (à la Thm) without thereby extending that very reasoning beyond that specification. We cannot by (whatever current) reasoning
methods (or beliefs) at our disposal arrive correctly at a
discovery of those same methods and their consistency, even
though we can consistently suppose there to be such a set of
methods and that it is consistent, and this itself can be
part of the set.

Thus I may believe that my beliefs about arithmetic are
true, and this very belief (if true) is an arithmetical fact
of consequence if it is applied to any specific belief set;
but then it lies outside that set. Alternatively, if I use
it only as a general principle about whatever beliefs I may
have about arithmetic, then it is useful for general
reasoning about reasoning. If outside information comes to
our attention, it may then be possible to observe the old
set, but now we have a broader set of methods at our
disposal. We would differ from a fixed formal system in
that we can accept new information, or that our axioms may
change. But this simply suggests analogy with a Turing
machine with variable or potentially infinite input tape,
still a mechanical device.

There may be a worthwhile distinction here between
various kinds of (formal or informal) systems, in terms of
their self-modelling in the above weak sense. It is
interesting that (Minsky 1968) suggests a mechanistic view
of consciousness much along these lines. One can imagine a
token 'Self' that is used much like Tr, in that it is not
fully specified yet enormously useful for general reasoning.
In fact our brief discussion of belief above bears close
resemblance to such usage.

Arguments to the effect that our minds are not
computationally equivalent to (fixed) formal systems are
probably correct: our axioms come from interaction with the
outside world and so aren't preset within our own brains.
(Also a formal logical system doesn't specify any sequence
of outputs, though this may not be a major point.) Thus
Godel's theorem doesn't apply to us in a general setting,
any more than to a Turing machine. For the latter can
indeed produce precisely the truths of arithmetic if its
input tape has enough information (an infinite amount). On
the other hand, in a practical setting where we may assume a
closed finite realm of inputs, a person (and a Turing
machine) will indeed fail to produce all arithmetical truth,
even though that very fact may be recognized by the same
person or machine! Godel's argument can be given with
reference to a generic or abstract sense of one's axioms
without our knowing just what they are, and still we can
reason precisely in this vein.

The fact that we can do such reasoning, or that a
formal system such as we have devised can represent such
reasoning consistently, does not mean there is something
here outside arithmetical or algorithmic behavior. This
behavior is completely arithmetical in the formal sense, even though it is 'about' non-arithmetical constructs. Were we ever to discover a complete specification of all reasoning methods possible for humans, and were we to mistakenly take this to categorize Tr, then we would fall into contradiction. The same applies to the systems we devise here. However, the assertion of consistency of these methods would then become a further method, so the set of (potential) reasoning methods would seem not to be recursive, i.e., our minds may in principle be able to grow arbitrarily complex. However, we have no such complete power of reasoning, we only grow little by little, as better representations become available to us through outside information.

Let us see how the usual diagonal 'paradox' is avoided here. Imagine, in a philosophic reverie, that a fancy Turing machine, TM, of roughly the complexity of a human being, say, Helena, has on its input tape precisely a description D(TM) of itself, and that TM's operation is to try to see what the described machine will do on that same input and then act differently. If TM simply simulates the described machine (itself) naively, running on the same tape D(TM), then an infinite loop occurs, and TM will not notice this. But if TM can reason enough about D(TM) to see it is looping if it starts to do so, then of course in simulation D(TM) will do this too! But if TM considers the possibility that D(TM) describes itself, TM may then conclude that the behavior of D(TM) under simulation can never be determined by TM, and this is true without any special checking of the detailed nature of D(TM): TM may see this by fully general reasoning just as Helena does. Both TM and Helena can state and use the fact that their reasoning techniques are consistent and correct, even though they have no precise representation of those techniques, and were they to do so, they could not show them to be correct, unless the 'experience' of being presented them altered their techniques themselves. Now, Helena and TM may each discover Godel's theorem, and each may surmise or assume it applies to herself and the world of inputs. But equally may they be able to discover that no specification of those inputs can ever be completely known to them since this would amount to a further independent input.

The above considerations can be seen in terms of the halting concept. We can reason in general terms about halting problems, and we can also consider the halting of any specified procedure. Regarding a particular language and planning system, such as PLANNER, we can state a theorem that summarizes the above windy paragraphs: There is no (correct) procedure implementable in PLANNER corresponding to a procedural description for looping, i.e., there is no procedure LOOP(p,q) which returns true iff the passed procedure p loops when passed q. Yet we can often play the
role of such a procedure, not by simulating procedure calls but by doing general reasoning with a tentative such procedure name to see what might ensue. Then we can even prove LOOP has no implementation. Now all this can be done within PLANNER, but by implementing the kind of general reasoning we are urging. A procedure 'DEDUCT' cannot refer to itself and avoid contradiction, unless it is very weak indeed, and in particular it then won't be able to establish the very looping and non-labelability results we are concerned with. What is needed is a non-procedural construct such as Tr. Our result on the realizability of such a construct shows that in PLANNER a theorem-prover could be written so as to achieve this. As an example, SHRDLU of (Winograd 1972) cannot deal with sentential variables or do reasoning about reasoning, but the underlying logic used there can be extended in an automatic manner to allow this. (The parsing aspect of this is another matter, as is the knowledge representation issue.)

The program FOL of (Weyhrauch 1978) is another system worth considering in this regard, for it embodies seemingly self-referential semantics. However, the current description of FOL is too non-technical to make precise comparison possible.

It is striking that adjunction of precisely the formulas $(\forall x)\text{Thm}(A(x)) \rightarrow (\forall x)A(x)$ was found by (Feferman 1962) to suffice in iterated extensions of Peano arithmetic to capture (rather nonconstructively) all true first-order formulas of number theory. It seems then that this distinction between Tr and Thm is not whimsical, for Feferman has shown this to be the very crux of the matter, at least regarding arithmetic. Tr has the above property, and Thm does not; yet its iterative introduction (in a sophisticated manner) takes us from Thm to (the intended sense of) Tr!

Apart from such philosophical concerns for 'the whole truth,' what practical conclusion can be drawn from all this? First of all, we are guaranteed a contradiction-free and easily described method for discussing the syntax-semantics bond of truth in any domain already expressed in logic. Second, as a matter of computational practice, we can use $\text{Tr}(x) \leftrightarrow x^{*}$ to 'reduce' formulas with Tr to ones that may be simpler, and thus we may reach a 'ground' instance whose holding is already known. Now, the whole difficulty we have been addressing arises because this does not always occur, even though it is counter-intuitive that the number of Tr's could fail to decrease with such a reduction. (Appendix A details this odd circumstance.) Nonetheless, we can adopt for any given applied domain a rule of thumb to the effect that if repeated application of
Tr(x) \iff \neg x^* does not lead to a ground formula in, say, ten steps, then for all practical purposes we don't care to look further. Here we use our domain knowledge to judge in advance whether many nested Tr's are going to be of interest -- in most domains that come to mind this will not be the case.

4. Formal results (proofs in Appendix A)

We shall define a class of theories of truth. Since a key question for us will be whether and to what advantage the inconsistent scheme (7)-(8) can be replaced by the inference rules

$$\text{Tr}(x) \equiv x \quad \text{and} \quad x \equiv \text{Tr}(x)$$

we will want to allow a relaxing of strict first-order theories for which the deduction theorem holds (we will see that this fails in important cases for the two rules above). Recall that to simplify reading, we often write Tr(x) and

Tr(-x) when a more complicated expression involving quote and Concat is called for.

Define then a **classical truth system** as any syntaxal first-order theory with a predicate Tr and possibly augmented by (non-first-order) additional inference rules, such that for no x is

$$\text{Tr}(x) \land \text{Tr}(-x)$$

a theorem (or axiom). This requires some explanation. Our idea is that there might be an odd case for which Tr(x) & Tr(-x) is undeniable in that the very denying of it is part of its truth so that it is inadvertently affirmed. So we merely insist that nothing force us to affirm a 'truth clash,' rather than affirming that there could not be such a clash. Of course we would be delighted to so affirm should this turn out to be feasible without serious drawbacks, and looking ahead to such a discussion, we recall Consis is the sentence

$$(x) \equiv (\text{Tr}(x) \land \text{Tr}(-x)) .$$

Note though that in Peano arithmetic P, or indeed any recursive consistent theory T, Consis is unprovable if we identify Tr and Thm (this is Godel's theorem on consistency proofs), and moreover any such T forms a classical truth
system with this identification.

If we replace excluded middle (F v -F) by a suitable intuitionist alternative such as (-F --> -F), so that the underlying logic is no longer classical, we call the resulting theory an intuitionistic truth system.

Intuitively, a truth system for us is a formal specification of a distinction or division among formulas, between those marked 'true,' those marked 'not true,' and possibly others. We avoid tying Tr(x) to the assertion of x, since we do not want to prejudge the sense of assertion. Thus although it is intuitively reasonable to require -(x & Tr(-x)), this is equivalent to Tr(-x) --> -x, which we wish to be free to consider without prior commitment. Here we really mean Tr(Concat(-,n:e ... e)) --> -e ... e where n 1 n

e ... e is closed.

We have then the following results, which are established in Appendix A:

In a classical truth system, there is a (non-positive) formula RR such that RR <-> -Tr(RR) is a theorem. From this it follows that not all formulas of the form F --> Tr(F) can be theorems in such a system. Moreover, if the rule of deduction x / Tr(x) is used, then not all formulas of the form Tr(F) --> F can be theorems. Finally, these results also hold for intuitionistic truth systems. (Appendix A, Theorems 1,2,3,4.)

This can be related to the aforementioned result of Godel. Consistency of a recursive theory T can be regarded as the assertion of the sentences

- (Thm(x) & Thm(-x))

for each x. But the sentences Thm(x) --> x are an even stronger assertion about T, namely a 'truth' assertion, and our theorem shows this to be in general not a theorem of T. We note that in particular, Peano arithmetic P does not have (s)(Thm(s) --> s) as a theorem, even though such sentences are true (in the standard model). We can of course have Tr(x) --> x for all (closed) x but without the rule

x / Tr(x) i.e., from x infer Tr(x),

simply by making Tr(x) false for each x, vacuously satisfying the implication. We can take this result as evidence that x / Tr(x) should be dropped since Tr(x) --> x seems to be part of what truth is about. We will return to this. But for now we note that the very oddity of truth in certain contexts, where a decision based on some information can lead to a destruction of that very information and so erode the basis for the decision, makes all our intuitions suspect at least initially. Should we find that Tr(x) --> x can be made part of a consistent framework, and that x /
Tr(x) clashes with some convincingly desirable result, then we will be on firmer ground.

We can construct a more faithful version of Godel's result as follows. Call 'constructive' any syntaxal theory with rule \( x \vdash Tr(x) \) and axioms

\[
\begin{align*}
Tr(x) & \rightarrow Tr(Tr(x)) \\
Tr(x \rightarrow y) & \rightarrow (Tr(x) \rightarrow Tr(y))
\end{align*}
\]

This is intended to capture part of the sense of the predicate Thm of arithmetic P, or any recursive theory. Then we have the following striking result:

In any constructive truth system, Consis is not a theorem. (Appendix A, Theorem 3'.)

In particular, this applies to any consistent recursive theory of arithmetic. In a sense, Theorem 3' says that abstract truth goes beyond constructive deduction, for Consis certainly ought to be true in some sense, yet this is not derivable in a constructive system. This as we have already indicated plays a role in our analysis of reasoning in mechanical systems.

Next we exploit a result in (Gilmore 1974) for set theory, modified to the syntaxal context and similar to the approach in (Kripke 1975) to model-theoretic truth. Our primary concern is to be able to reason formally about truth with full first-order logic, including excluded middle. This means that in particular we will want to have \( Tr(x) \leftrightarrow x \) for 'normal' formulas \( x \), so that truth will have its intuitive sense. The idea is that positive formulas are the ones to focus on; this seems to have first appeared in (Gilmore 1974) in the context of set theory although unmotivated as regards truth or decisions with priority as we have at length discussed.

Now, in (Kripke 1975) a notion of truth was defined for a structure, which in fact implicitly involves the closure of our (12) equivalences, i.e., Kripke in effect constructs a distributive version of truth-labeling. However, he works solely within a given model, assuming sentences not labeled Tr to be totally undecided. Recalling the formula RR with the property \( RR \leftrightarrow \neg Tr(RR) \), this means in particular that the 'meta-fact' we observe that RR and \( \neg RR \) are unlabeled and that RR holds (in the usual Tarskian sense) is unrecorded in his semantics. Moreover, one cannot use excluded middle freely in his scheme, for the same reason. (These are objections made by Kripke himself.) Since a similar iterative definition was used by Gilmore to show his set theory consistent, without giving up excluded middle, our
Idea is to modify his approach in an attempt to codify Kripke's construction in a first-order formalism. In effect, whereas Kripke intended to supercede Tarskian truth, we seek to Tarskize Kripke's approach. (See Field 1972 for a critique of Tarskian semantics.) Our result then is

(Appendix A, Theorem 7): We can consistently assume $\text{Tr}(x) \iff x$ for all positive formulas $x$.

The question then arises, whether we can salvage the best of both worlds, i.e., the schema $\text{Tr}(x) \iff x$ for positive $x$ and the two rules $x / \text{Tr}(x)$ and $\text{Tr}(x) / x$. This would appear to correspond most closely to intuitive notions of truth, in which establishability guarantees truth, truth is equivalent to 'holding' when the formula in question is positive (or grounded, in Kripke's approach) so that a clear-cut meaning is felt to be present, and establishing the truth of a formula guarantees the (meaning of the) formula. (It is essentially Kripke's reliance on groundedness rather than the broader notion of positiveness, that makes his approach weaker than that of Gilmore. Thus Gilmore gives us RR as a formal conclusion, as well as $-\text{Tr}(RR)$ and $-\text{Tr}(-RR)$, while Kripke allows none of these.)

It may appear that we can add $\text{Tr}(x)$ for any theorem $x$, with impunity. But we already know this leads to a contradiction. (We can see this more clearly in the model $M$ of Theorem 7 (Appendix A), for there RR is true even though it is not positive. Recall $\text{RR} \iff -\text{Tr}(\text{RR})$, and since RR is never decided as true, i.e., $(\text{RR})^*$ doesn't hold at any stage in the construction of $M$, then $\text{Tr}(\text{RR})$ is never added as a true atomic formula. So RR holds in $M$, yet so does $-\text{Tr}(\text{RR})$ (indeed, these are equivalent, due to the peculiar nature of RR). Hence the theory of (all truths of) $M$ is inconsistent with $\text{Tr}(\text{RR})$ even though RR is a theorem there. If we add $\text{Tr}(\text{RR})$ to $M$, we inadvertently remove RR, so $\text{Tr}(\text{RR}) \rightarrow \text{RR}$ becomes false. Apparently then our intuition about $x / \text{Tr}(x)$ applies only to cases where this 'double jeopardy' doesn't occur.

It is convenient to introduce some terminology here. A truth system with the axiom schema $\text{Tr}(x) \rightarrow x$ for all formulas $x$ is called definite; one with $\text{Tr}(x) \iff x$ for positive $x$ is positive; and one with $\text{Tr}(x) \iff x*$ for all $x$ is reductive. It is easy to see that any reductive system is also positive, and that a reductive system with Consis is definite. (Note Consis $\rightarrow (x^* \rightarrow x)$.) Moreover:

No reductive truth system can have the rule of inference $x / \text{Tr}(x)$. (Appendix A, Theorem 9.)
It begins to sound tempting to identify \( \text{Tr}(x) \) and \( x^* \), the more so because Gilmore's construction (as modified for our purposes) shows we can do this consistently in a truth system, and furthermore we get \( \text{Tr}(x) \vdash x \) free of charge, i.e., subsumed in a pure first-order logic. Moreover, we also get \( \text{Tr}(x) \vdash x \) for all \( x \), and hence we get Consis too. Finally \( \text{Tr}(x) \iff x^* \) yields a distributive system (see (12)). Perhaps after all it is worth giving up the 'token' adjoining of 'Tr' to theorems simply to sooth our urge to compound confidence with labels. This is especially compelling now that we see such adjunction can violate its very intention as with RR and \( \text{Tr}(\text{RR}) \), i.e., when we are dealing with a non-positive formula.

If we take this approach, i.e., that of requiring \( \text{Tr}(x) \iff x^* \) and \( x^* \vdash x \), we seem to lose no significant expressive power. For instance, we can derive RR, i.e., \( \neg\text{Tr}(\neg\text{RR}) \), as well as \( \neg\text{Tr}(-\text{RR}) \), and these tell us precisely that it is folly to adjoin Tr to those very theorems! What more could we want in this regard? Still the technical question remains, whether a positive truth system can admit the rule \( x \vdash \text{Tr}(x) \), but there is now much less reason to be anxious about such an eventuality.

A serious doubt may arise regarding Peano arithmetic \( P \), where we have (implicitly) the rule in question, and where Consis is not a theorem, but this merely serves to point up a distinction between Thm and Tr. The latter has a broader intended significance. Thus in the standard model for \( P \), we have \( \neg\text{Thm}(x) \) for suitable \( x \), but this will not (always) be a theorem of \( P \); however in a truth system of the sort we now envision, i.e., reductive and definite, there is the possibility of obtaining \( \neg\text{Tr}(x) \) as a theorem whenever this holds in the standard model \( M \) for the system. The difference lies in the tying of Thm to the recursive nature of \( P \) while Tr is taken to have a meaning independent of the axiomatic details.

Lest the above examples with RR and \( \text{Tr}(\text{RR}) \) seem far from ordinary concerns or ideas of importance in concrete reasoning, we note that Consis expresses a very natural and important fact (or desired circumstance that we normally assume to be the case) and that Consis holds in a positive definite truth system. However it is easily seen that \( \text{Tr}(\text{Consis}) \) yields \( \text{Tr}(x) \lor \text{Tr}(\neg x) \), so that \( \text{Tr}(\text{Consis}) \) is disprovable there. Thus the failure of \( x \vdash \text{Tr}(x) \) occurs at a very intuitive and serious level, so that the very notion of systematic correctness and consistency is at issue. We want to be able to assert correctness and consistency of Tr labels, in a correct and consistent manner, as is the case in positive definite systems. The outcome though is that we must give up \( \text{Tr}(\text{Consis}) \).
We may summarize our main findings as

Any consistent first-order syntaxal theory $T$ has a first-order extension $GK(T)$ (for Gilmore/Kripke) which is a reductive truth system that is positive and definite, distributive, and consequently with Consis and the (subsumed) rule $\text{Tr}(x) \vdash x$. Moreover, the rule $x \vdash \text{Tr}(x)$, the last refuge of the presumptive fallacy, is necessarily left out in any reductive system.

Whether $GK(T)$ is maximal for $x \vdash \text{Tr}(x)$ is not immediately clear, i.e., whether $\text{Tr}(x)$ is a theorem whenever this is consistent with $GK(T)$ and $x$ is a theorem; such a result would certainly be desirable and would formally establish our goal of basing truth on priority:

$$GK(T) \vdash \text{Tr}(x) \iff GK(T) \vdash x \text{ and } GK(T) \not\vdash \neg \text{Tr}(x)$$

where $GK(T) \vdash Q$ means $Q$ is a theorem of $GK(T)$. This then remains an open question.

CHAPTER III
THE LOGIC OF BELIEF AND THOUGHT

1. Introduction

Our initial concern in this section is to settle on some terminological usages and provide an overview of the general area of our efforts. Our long-range goal here is to characterize reasoning from beliefs, in a way that allows the construction of a satisfactory concept of epistemic possibility along resource-limited lines. We will see that this impinges on various parts of cognitive science, e.g., knowledge representation, planning, and inferring. Our central contention in this chapter is that attention is what keeps computation tractable in a reasoning being, and that therefore processes by which attention changes should be modelled in systems that hope to address the problem of reasoning in a complex environment.
Three particular problems that will arise repeatedly are the following: paraphrase, focussed inference, and defaults. These seemingly diverse areas will turn out to be closely linked in our work. We shall illustrate them here with sample problems:

1. Sally: 'Uncooked lobsters are green.'
   John: 'These are red, so they must be cooked.'
   Sally: 'Right!'
   Phil: 'Live lobsters are uncooked, so they must be green.'
   Sally: 'Of course, that's what I just said!'

   Why does Sally regard Phil's statement as a trivial paraphrase, and not John's?

2. Psychologists such as (Wason & Johnson-Laird 1972) have provided examples of unsound inference, in which subjects seemingly miss key observations due to not focussing on the relevant cues. What is focus of attention and how is it related to inference? Do we really misuse logic?

3. A meeting-house burns down, killing all within. If I'd been there I'd surely have died. Yet how can I conclude that my (counterfactual) presence wouldn't have prevented the fire or its severity? This is the frame problem, at least in part: how to account for all (relevant?) consequences of some alteration in circumstances, whether real or envisaged?

   These will provide illustration and motivation for the model of belief and inference we will develop in this and the next chapter.

   We wish to be able to view a belief as something taken at face-value, and also as an object of examination, so that a belief can be held and then later questioned in the context of another belief:

   'John is a Russian.'

   'REALLY?'

   'I said so, but maybe I'm wrong.'

   The meaning of the first and third statements can be represented in first-order logic as

   \text{Said('is(Russian,John)')} \& \text{Unsure('is(Russian,John)')}

   This kind of interplay is perhaps the main theme of this thesis. Chapter II has provided formal justification for such notational usage, and Chapter IV will give some
concrete experience. In this chapter we sketch and motivate the underlying model of thought this embodies.

2. Previous approaches to belief and inference

Taking (Hintikka 1962) as a starting point, (Moore 1979) has worked out in admirable detail a treatment of knowledge and inference based on modal logic. His arguments can be viewed as applying equally well to beliefs as to knowledge. He explicitly embraces the well-known feature common to this approach, of requiring all logical consequences of given beliefs (or knowledge) to also be beliefs. He shows that various problems can be stated and dealt with in a clean and plausible manner in such a setting. However, no mechanism of reasoning is postulated or indeed even possible in this context, since conclusions when logically justified appear as beliefs by definition. Thus the entire issue we are concerned with here, namely an inferential procedure that can be examined and made to obey restrictions, is unavailable. In particular, there is no facility for considering time as a modulator of beliefs, in that things once believed can never be forgotten or ignored.

By way of contrast, (Eberle 1974) attacks the problem of finding which inferences might be actually made from a given set of beliefs. He allows for inconsistent beliefs, and does not demand that an inference be made simply because it can be justified logically. However, this simply leaves totally open the question of which inferences in fact people do make (and why and how). Also, he doesn't consider active (currently used) beliefs as distinct from distant (unconscious or forgotten) ones.

(Partee 1973) comes a little closer, taking (Carnap 1947) as point of departure. She is concerned with specific computational restrictions on inference. However, she (like Carnap) sticks with the simpler notion of belief equivalence rather than inference. There again is no connection with time or limited focus of attention.

In (McCarthy et al 1978) an attempt is made to deal with knowledge (or belief) modally so that one person can reason about another's reasoning about yet another's, and so on. The approach is similar to (Creary 1979) in that there is an implied simulation of other's inferential processes. But in neither of these is any mechanism of inference supplied, nor any notion of limited access to beliefs for any given inference.
The work of non-monotonists such as (Doyle 1980) and (Reiter 1980) and to some extent (Weyhrauch 1978), has again this same lack, of any limitation placed on reasoning: all beliefs in existence are available to inferential processes at all times, making consistency much too important an issue, even though these systems allow a kind of contradiction to exist between various moments.

The biggest single common failing of the above approaches, with respect to the goals we are pursuing here, is the lack of changing focus of attention on a limited subset of all the beliefs in existence as inference proceeds.

3. Theoretical restrictions

Suppose an operator Bel is given, either as a modality or as a first-order predicate symbol. What axioms are plausible regarding Bel, and what restrictions can be placed on it from constraints of consistency and of practicality? Note that we are in no way saying that beliefs should be consistent with one another, but simply that our theory about beliefs should itself be consistent (otherwise how can we understand it as an intelligible and testable set of assertions about behavior?).

We can easily see that Bel must obey similar constraints to those found for Tr, most particularly,

\[ \text{Bel}(x) \iff x \]

cannot be the case for any belief system that is syntaxal, i.e., if the range of entities that are potentially believed includes assertions about beliefs, and if we can pick them apart. Thus we CANNOT believe precisely what is the case in such a broad setting. Note that this truly applies to people, not just a formalism. Any entity with a set of 'beliefs' must respect our theorem, no matter the logicality of that entity. Of course, here we ignore subtleties regarding time and change of beliefs.

We can of course identify Bel and Tr, requiring then

\[ \text{Bel}(x) \iff x^* \]

This will then have precisely the nice properties described earlier for truth, but is obviously unsuitable for belief since we may believe false and even contradictory things. In fact, we do not wish to characterize WHAT is believed by anyone, since this may vary in unpredictable ways. Rather
we seek general principles governing interrelations between beliefs and inferential relations. Key points will be how Bel applies to itself (i.e., Bel(Bel(x))) and what we have earlier described as automatic inferences, ones made by virtue of the mere presence of the proper antecedents in consciousness. Toward this end, we will be motivated by practical limitations of computational activity, not only as in computers, but in any resource-limited entities such as ourselves.

We can further push the formal arguments for Tr onto Bel: for the argument in Appendix A giving rise to a formula RR whose interpretation is \(-Tr(RR)\), applies identically to establish a formula BB with BB \(\equiv -Bel(BB)\). Thus, BB says BB is not believed, and if we don't believe BB then it is true so we will fail to believe some true statement (namely, BB). But then in seeing this, we come to believe BB and at that instant it becomes false! And it never settles down, since our attitude toward it alters its validity which then may flip our attitude, etc. However, there's no inconsistency in flipping, especially if time is considered as a parameter to beliefs. For example let

\[-Bel(Sub(x,quote(x)),t)\]

define a predicate B(x,t), and define BB(t) as

Then if we don't believe BB(t) and come to see it is true, we can then believe it without changing its truth at the time indexed. Still, the moral is worth drawing: it is impossible even in principle to believe precisely the true sentences in a language allowing expression of syntax. This is independent of one's cleverness or luckiness or omniscience, for the very meanings of some beliefs depend on what is believed and therefore don't have a truth of their own: they refer to properties of the believing system rather than some outside conditions.

4. Representation and logic

We have mentioned earlier that beliefs can profitably be regarded as sentences. This needn't be interpreted strictly in terms of sentences physiologically stored in the brain, but only at some suitable conceptual level of abstraction. It is sufficient for us to accept that a belief is a mental phenomenon and as such can be described in principle in terms of various physiological conditions and processes which in turn can be written down as strings
in a suitable language, such as English or first-order logic. We are showing a bias here in favor of some kind of mind/brain identity theory or perhaps functionalism. People generally express beliefs in a natural language, whereas we here will usually employ first-order logic, even though passing from a natural language such as English to a formal representation is extremely non-trivial. We will assume such a front end translator already exists, or that whatever information we wish to reason about can be readily expressed directly in our formalism.

Although we shall not develop it in any great detail in this work, we find it fruitful to view images or descriptions as fundamental elements out of which beliefs and memories arise. Thus 'Jim in office' can be viewed as a description of an image (the appearance of Jim in his office) which we may have experienced once (by seeing Jim there, or by thinking about how he would look there, etc), in which case the image/description corresponds to a memory, namely, of a prior time when the image was first invoked. Further, we can form a belief (that the image is 'true' or 'real') if we attribute to it an external source of an appropriate sort.

The ability to deal with sentences (or images) as related to but different from realities seems crucial for thinking, planning, considering possibilities, using symbols as being about other things. If we confused our every image or notion with external reality, we'd get little done. Conversely, our increased versatility and skill in many areas as we grow up would seem to depend in part on our willingness to look more seriously at possibilities different from what most readily comes to mind.

What we will defend now is the implicit claim above that first-order logic is up to the task of describing anything English can describe, or indeed anything we can think about. The essence of our claim will be contained in our General Hypothesis below, to wit, that our thinking is rule-based and involves only extremely simple steps readily expressed in logic. We have already rather extensively presented a theory of truth in first-order logic showing that self-reference (and therefore punning) can be dealt with in a natural manner. To further support our point, we must be more specific about just what we mean by logic expressing something. This will evolve as our discussion develops.

There is currently a plethora of notations and representational systems in artificial intelligence. Recent efforts include (McCarthy et al 1978), (Creary 1979), (Weyhrauch 1978), and (Doyle 1980). We must show that our approach of first-order logic can subsume this work. The main point here is that we want to describe certain features
of mental processes in some formalism, we don't want the formalism's proof-mechanism to be (or correspond directly to) those processes. In effect, efforts of the latter sort restrict the full play of logic rather than augment it. We already know enough about the world to be able to express changing properties with time, processes that interact, and so on, in a perfectly ordinary (formal or informal) language. Yet many systems put time and process in the background and seek instead to capture the complex workings of human thought by a mechanism akin to the proof-theory of formal logic. Our approach, on the other hand, uses logic simply as a highly expressive and semantically adequate tool for scientific descriptions and analyses: there is no implicit assumption that the processes in question are themselves 'logical' or 'consistent.' (They are algorithmic and include logical capabilities, to be sure.) Thus we stay strictly within a monotonic logic.

We will amplify on the above point, for it is easy to become confused by our terminology. First-order logic (FOL) has two quite distinct senses, only one of which is technically relevant to most situations here. These senses are 1) as a formal language with a well-defined semantics, and 2) as a mechanism for generating proofs of theorems. Associated with any first-order theory \( T \) is the set of its theorems, \( \text{Thm}_T \). Moreover, there is an explicit characterization of \( \text{Thm}_T \) in terms of operations defined in \( T \), giving rise to the second sense above. However, \( T \) is virtually never used in this way, as a generator of \( \text{Thm}_T \) by exhaustive application of these operations. The most significant fact regarding \( \text{Thm}_T \) and \( T \) is that all the results of sound inferences that can be made from the axioms of \( T \) are already given in \( \text{Thm}_T \). FOL then admits via \( \text{Thm}_T \) of \( T \) a characterization of all that is true as a result of certain axioms, just the sort of thing that is wanted for a clear technical study of any given area.

But this is distinct from the expressiveness of \( T \), or FOL in general. FOL addresses whatever we tell it to, with appropriate axioms, and this can include process notions such as inferential schemes, whether the exhaustive one giving the theorems of FOL itself, or some other. \( \text{Thm}_T \) tells us what happen to be the consequences of the axioms of \( T \), if \( T \) is a first-order theory; it isn't the arbiter of what the axioms of \( T \) are about, nor of what processes (inferential or otherwise) those axioms are subject to in any given system \( T \) may belong to. Once we have decided what procedure we wish to investigate, it is duck soup to write it down (yes, in FOL). To expect FOL itself to have the inferential processes already determined would be something
like asking English to do our thinking for us. We can think in English, and even express our thought processes in English, but English itself does not express in advance any procedure for inferring, and this is no shortcoming. (We will describe our system (to be presented soon) largely in the same language (FOL) that the system uses, and then it too can have descriptive statements about itself.)

The confusion about FOL seems to stem then from regarding Thm as embodying a kind of internal and fixed sense of inference expressed in FOL, whereas in fact it is quite different: Thm contains the exhaustive truth implicit in a given set of axioms in an instance T of FOL. It does not come expressed automatically in FOL or any specific instance T thereof, nor is there any reason to seize on the proof-theoretic feature of FOL as the place cognitive relevance should be exhibited. This apparently is the source of the oft-heard remark that logic cannot express the notion of process. Why Thm should ever have been linked to thought processes is not clear to me. Perhaps it was hoped we (and robots) could be 'smart' enough to see all logical consequences of our beliefs instantly, and 'sturdy' enough to agree to them. It seems that in reality we may be a lot smarter and sturdier than that, in being able to function so well without being tied down by such exhaustive searches, or the requirement of (global) consistency.

Thus although (Winograd 1980) is right and (Hayes 1977) wrong, in that the semantics of a system can depend on properties of the processes involved, Hayes is right and Winograd wrong in that first-order logic remains adequate to the task of expressing this dependency. As we discover more about the processes, we can express them in first-order logic, using quotation when necessary. The processes need not conform at all to the proof-theoretic mechanisms of logic, but can be whatever we deem appropriate; this has no effect on our expression of them as formulas of logic.

Clearly exhaustive generation of truths is not what artificial intelligence needs, but this bears in no way on the usefulness of FOL to A.I. We need a model of non-exhaustive processes of inference as an expression of relationships between stages of information-processing. FOL will express whatever we wish to say about such. Efforts such as (McDermott & Doyle 1980) simply compound the exhaustive difficulties, for they rely, in addition to the proof-mechanism of FOL, on the concept of consistency, which is even less tractable than ordinary theorem generation. That is, they wish to use information to the effect that a given statement is not present in (or derivable from) some data base, to draw a conclusion, such as that the statement's negation is acceptable (as a default, say).
This is computationally tractable only under severe restrictions, such as a closed-world assumption (Reiter 1980). Our model to be presented shortly divides information into several parts, of which the one available to inferring at any moment (short-term memory) is not only finite but quite small, so that it is entirely feasible to conclude that some statement is not present there. This captures the spirit of non-monotonicity, but as a behavioral principle of mental activity, not one of formal derivability. The principle itself of course is perfectly expressible in a formalism, which is how we shall treat it later.

Now that we have argued that first-order logic doesn't suffer from certain negative features, what positive features can it offer us? To name a few advantages, quotation applied to beliefs allows us to talk (and think) about inconsistency, error and recovery, and others' beliefs. The following concrete illustrations of the power of first-order logic and syntaxality are presented below:

1. theory subsumption
2. prototypical, pseudo-quantifiers, and defaults

3. truth and falsehood
4. word and object

Theory Subsumption

Consider Doyle's example of one theory contributing to another in a descriptive manner as an illustration of a presumed weakness in standard first-order logic. He suggests it may be desirable to establish a theory of horses based on a theory of mammals, by stating that the former extend the latter. It is not sufficient to state simply Horse(x) --> Mammal(x), where x is a string that possibly names a horse or mammal, for this doesn't give us an independent theory of horses that we can then tamper or reason with apart from the mamal theory. We would like to be able to say, in effect, that the 'mammal' axioms give rise to 'horse' axioms by replacing the term 'mammal' by 'horse' in the initial axioms. But in fact we can do this in first-order (syntaxal) logic, and very directly, almost as we say it in English. We simply say

\[(p)( (x)(Mammal(x) --> Apply(p,x))

--> (x)(Horse(x) --> Apply(p,x)) )\]

Here Apply is defined by Apply(y,x) \(\leftarrow\rightarrow\) Tr(Concat(y,1,(x,1:))). Thus \((x)(Mammal(x)-->Apply(p,x))\)
is a condition that expresses that $p$ is part of the mammal-theory. If we wish, we can get fancier and say Theory(mammal) and proceed to do abstract reasoning about theories.

It is unclear how important this example is in ordinary reasoning, and Doyle doesn’t make a solid case for it. It does seem though that theory subsumption is important for lots of common situations, e.g., whole sets of properties are sometimes carried over to another domain and then modified, as when a new School of Humanities is formed within a university and initially ‘copies over’ the graduation requirements from the Science School. Although the copying can be done without reference to propositional variables, it is hard to see how the intention to do so or the fact of having done so can be expressed otherwise.

Prototypical, pseudo-quantifiers, and defaults

Doyle gives a more convincing example of what he calls 'family resemblance,' in which it is desired to specify a prototypical or generic member of a given family. If we can refer directly to characteristics of members of the family as terms, then we can have a predicate saying that those characteristics applying, say, to more than 80% of the members are considered prototypical. And this is indeed feasible in as succinct a fashion as we have just stated, in

\[ \text{Prot}(p, \text{Fam}) \leftrightarrow \text{ApplyMost}(p, \text{Fam}) \]

expresses that $p$ is a prototypical property of the family Fam precisely when $p$ applies to ‘most’ of the members of Fam.

ApplyMost can be defined formally in various ways, one being by viewing Fam as a set and placing conditions on the relative proportions of subsets of Fam to which $p$ does or doesn’t apply; another is via the introduction of a kind of pseudo-quantifier ForMost:

\[ \text{ForMost}(1:x, '\text{Fam}(x) \rightarrow \text{Apply}(p,x)') \]

where this again is defined by subset proportions. Then one can adopt whatever axioms are desired for this regarding distributivity over ‘&’, ‘or’, etc.

In a related fashion we can express default reasoning. Thus we may assume a given family member to have all (or any currently of interest) of the prototypical properties, as a
default. Then if we later see something is wrong with this, we can adopt an appropriate response. Doyle has one approach to this, but it is obvious that in reality many approaches are both used by people and appropriate in different contexts:

1. bullheadedness: go on with the default against all evidence.
2. noncomittal: flip a coin to decide what to believe.
3. surrender: stop working on the problem.
4. appeal: get outside advice.
5. logical: think hard, look for reasoning errors, question assumptions.
6. empirical: run tests, proceed with caution.
7. eclectic: any combination of the above.

For example, an empirical 'Check' could reasonably be taken to override other conflicting information. This itself can be written as additional information, a default-handling rule that is simply one more piece of (first-order) information that happens to refer to (at least part of) the set of beliefs themselves. Thus

isCanary(Tweety) and isCanary(x) --> isYellow(x) conflict with isGreen(Tweety). But this can be resolved if the latter is Checkable by observation. If it turns out that the Check is positive, it still is not necessary to discard the conditional statement isCanary(x) --> isYellow(x). In the future additional Checks may be needed; or we can accompany the conditional by information about its doubtfulness, even using OnlyUsually('isCanary(x) --> isYellow(x)'). This is not intended to provide a system for default handling, for it is too sketchy; but we claim it does show FOL is not standing in the way.

Such a rule itself is subject to being overridden if still further rules are present to that effect. It all depends on what techniques are currently available to the system's focus of attention. It may recall one rule and then another, if its long-term associative memory (to be discussed later) so behaves. Whether this can be arranged (or evolved) in a way that leads to sophisticated and robust behavior is another issue, one way of describing the frame problem. We suggest that it is possible, and try to give some evidence for this in Chapter IV. However, this has apparently nothing to do with the expressive adequacy of FOL that we are defending in this section.
Now, as long as we take our beliefs too seriously, by representing them as facts (axioms), then the issue of consistency rears its head. But if we write them in quoted fashion, e.g., inside Belief predicates, then this doesn't arise, we simply create new beliefs and perhaps mark old ones inactive or untrustworthy. Details will have to remain until we get to the issue of focus in a later section. However, we see here the point that old beliefs don't have to be thrown out or even revised as such, so much as augmented by further beliefs. This means inconsistency of (quoted) beliefs, but matters not at all as long as we have the means for getting the appropriate beliefs in focus at the right times (the frame problem). We suggest that this is done not by reasoning and post facto analysis in the usual sense but simply by weighted associations in our memories, that get built in as we learn. Thus we have belief X, we learn to qualify X by Y, and still belief X remains and is at times invoked, and at times we find ourselves recalling Y too. If we get into trouble a lot due to misusing X without Y, this should stimulate our X-Y associations.

We don't mean to say that this reduces defaults (and more generally, the frame problem) to trivialities. Nor do we pretend to have anything like a definitive treatment of this. However, we hope that the division into (conscious) reasoning and retrieval from (long-term) memory places the frame problem in a perspective that is useful.

The point is that we have many beliefs, that in principle could conflict endlessly. Specific measures are needed for specific conflicts in specific situations of attention-focus. Thus we claim that no one default-handler is going to be satisfactory, and in fact our approach regards defaults simply as further information of no significantly different character than other information that may come into focus. We are saying then that default reasoning, like all reasoning, is a psychological and ecological (survivalship) issue and not one of formal logic. Whether we can discover associative mechanisms that bring suitable (and suitably few) beliefs into focus at the right times, remains a major problem, one involving an interweaving of various techniques. This has been done with some success in very limited domains where an overabundance of information isn't a problem. In Chapter IV, we give a very slight indication of what a more ambitious domain might look like, but a really serious effort will remain for the future.
Truth and falsehood become important when more than one agent is present, for then each has the problem of assessing the veracity or trustworthiness or appropriateness of the others' statements. This then involves relating syntax (what agent x says) and semantics (what x means or ought to mean, etc.). Apparently the well-known paradoxes of truth definitions have successfully scared researchers away from first-order formulations of this via terms for syntactic elements. Our syntactic analysis and consistency proof can be viewed as offering some philosophic comfort here. (But as Andy Haas keeps pointing out to me, the consistency proof shouldn't really matter since any practical reasoning system can't use the full strength of first-order logic in its daily problem-solving anyway. Some other means is needed to limit deduction - e.g. for the frame problem and error recovery - and this may just as well serve to take care of the paradoxical cases.)

Word and Object connections are also handled easily by syntactic languages; indeed this is the paradigmatic case for such languages. Thus, the difference between the concept of telephone, a particular telephone, and the name 'telephone' is readily expressed by passing from predicate to term: Telephone(x) says x is a telephone, whereas Apply('Telephone',x) notes that 'Telephone' can be used to indicate a predicate applying to x. We can also write Is(TELEPHONE,x) and use Concept('Telephone') as a function applied to a term and returning an entity regarded as a concept:

Concept('Telephone') = TELEPHONE

Here we follow (McCarthy 1979) in using capitals for concepts. All these notations are available in first-order syntactic systems, and not only as alternative choices but together. E.g., we will have

Telephone(x) <-> ls(Concept('Telephone'),x)

and even

Apply(y,x) <-> ls(Concept(y),x)

This ends our quick illustration of the expressive power of logic. Now we comment on a possible difficulty of this medium of representation, which however we think can
really be handled in a reasonably straightforward manner: sentential imagery. It is equally clear that lots of information in people is stored as 'images' in some significant sense, and that these images are far more highly processed than mere pixel data. To maintain our contention that beliefs are reasonably viewed as sentences we should be able to account for imagery in terms of sentences in first-order logic.

Now, (Sloman 1971) presents an argument that some kinds of reasoning use information that is represented 'analogically' and not (presumably) sententially; he specifically uses visual images as a key example. But it is not at all clear how the information in an image is non-sentential. Indeed, it is easier to suppose that it is sentential in form, both as a matter of convenience and of intelligibility. For example, if the conjunction of lines to form corners isn't explicitly represented by reference to a stored concept of corners, then it must be calculated at each usage from more primitive data, and this is hardly conducive to the facility Sloman claims for analogical representations.

What is more, such a representation can be no better than the direct perception of what it depicts. But sensory perception may well be sentential (parsed, relational) in form. Now, it is true that features of an image can be extremely useful guides to using the image, such as ready recognition of a sequence of connected objects which can affect one another in turn. But this too is easily expressible in logic. Whether such a fact will be called to attention is another issue. Perhaps our visual system picks up such things readily and forces them into consciousness; but this only says there is a strong associational link between certain visual memories (beliefs, information) and other entities in the brain. Any system that makes inferences from a huge set of information will have to have such associations as a control structure.

Summary

We now conclude this section on representation, with a brief illustrative plea. People may have been trying to make logic do too much. We get along in the world, not so much by logic (correct or incorrect), as by lots of special responses tied to special circumstances arising in the flow (external and internal) of information coming to our attention. The control of this flow of attention then needs addressing for a better understanding of the questions we have been discussing. Logic should be regarded then as providing meaning (semantics) to our ruminations but not so
very much a strategy of inference: we aren't very good at brute force theorem-proving, and such doesn't seem suited for everyday decisions anyway. We use (and need to use) more real-time resource-limited methods.

We do act in certain ways, and this doesn't say we believe our actions to be logical, whatever that might mean, though some logic may contribute to the eventual action. This doesn't mean we misuse logic; rather we try things out if they seem to have a kind of plausibility: we do a perfectly fine bit of speculation.

As an example, I think (Anderson 1976) is in error in regarding the following inference as evidence of illogical thinking:

1. If I was dying the doctor would be frowning.
2. The doctor is frowning.
3. Therefore, I am dying.

Anderson suggests such reasoning may arise because it has worked before by chance, or because it convinces others, or because it is used by others. But if we re-write it slightly, there is no need to regard it as representing an invalid argument at all. For if we don't claim our beliefs to be iron-clad, if we seek merely good hunches that give reasonable pictures of our environment, then we are acting as amateur scientists, trying to fit theories to reality. Let us put his example the other way around:

1. The doctor is frowning.
2. The only interpretation I can think of is that he has bad news for me.
3. Therefore I'll go on the assumption that I'm in bad shape, could even be the worst: death.

Now, we needn't actually state this so carefully, we can jump from the frown to the conviction we are dying (the 'reasoning' may be done by an unconscious process of relaxation to find the best/nearest interpretation). But this still doesn't indicate we believe our conclusion to have logic behind it. That the subject is dying is certainly a reasonable conjecture, and there's nothing wrong with taking a conjecture as true. After all, when do we ever know for sure an attitude is correct? We can admit all our knowledge is imperfect and still use it just the same, without being guilty of faulty reasoning. It would be very striking if Anderson's subjects were to report a firm conviction that the conclusion was not merely a good one that fit the facts and that they had decided to accept, but
that it was in fact the only possible one in any interpretation! That would be illogic indeed!

5. Active belief models

To represent ourselves representing information in a given domain, we need to admit beliefs as viable entities. Furthermore, using beliefs to guide action and to produce further beliefs involves self-regulation and self-reference. This permits us to avoid exhaustive (theorem-prover) analysis of information that is generally impractical in real-time situations (if not outright impossible due to the difficulties inherent in consistency checking). It also avoids use of an arbitrary cut-off point beyond which one looks no further, clearly both unrealistic as a cognitive model and undesirable in a varying environment that requires responses of varying degrees of careful analysis.

Self-regulation is needed to allow us a concept of default or 'tolerance' that conveniently ignores most of the world most of the time. This is not simply resource-limitation, in the naive sense of running out of time or space, for we can spend arbitrary amounts of time and space when we really want to. Rather, our efforts should be viewed as guided by the extent and nature of our concerns, goals, attitudes; in other words, by our beliefs themselves, including those about what we are doing. Thus we postulate a kind of self-regulating reasoner that decides what and how much to analyze. The minimum here will yield our default state in which we notice only what forces itself on us by the very meaning of what we are thinking, i.e. no intermediate logical steps required.

We isolate two cases in which this seems highly plausible: direct contradiction \((A \land \neg A)\) and direct instantiation of conclusion from antecedent in a conditional (essentially modus ponens):

\[
\text{from } Ac \land Ax \rightarrow Bx \text{ conclude } Bc.
\]

It seems unavoidable that we should notice contradiction in the former case once both elements are jointly observed, and equally unavoidable that we should note the conclusion of the latter case once we have jointly understood the premises: no further steps are needed before reaching these results. On the other hand, we claim that (most) other logical rules do require additional mental effort and therefore motivation as well as learning how to apply them. This can be illustrated by various examples of reasoning, such as contrapositives, which are often done incorrectly or
not at all, unless properly set up so as to demand attention to the conclusion sought.

Still, other rules than modus ponens may be desired. Thus from A, B, and A&B --> C we should be able to infer C without having to form first a long conditional as in some treatments of first-order logic. Similarly for A and A\lor B --> C. On the other hand, it is not necessary that \neg A be inferred from A --> B and \neg B, and in fact this will be an important point later.

Most importantly, we must not be ambitious about codifying syntactic rules for belief-inference. Very little seems to be automatically inferred in general, and there may be good reason for this. For example, let us consider a 'stubborn' belief probably not uncommon in people at times, especially children (recall we leave off quotes and Concat when context makes the meaning clear):

\text{Bel}(Q, (\times)(\text{Bel}(Q,x) --> x))

This says that person Q believes her/his beliefs to be the case, i.e., in Q's model of the world, that portion which represents Q's beliefs is true of that whole world. Contrast this with

\text{Bel}(P, (\exists x)(\text{Bel}(P,x) & \neg x))

expressing knowledge of self-failibility in the prudent thinker P, an attitude necessary for lots of planning activity carried on while acting at the same time. Thus, pursuing an effort toward a goal without foolhardiness, involves acting on beliefs while checking from time to time whether they hold up. Here P knows enough about the world to be convinced that there is a difference between current convictions and reality, even though P doesn't know of any particular disparity (that would be ground for changing one's beliefs, after all!). So in P's 'world model,' there is something believed but false (or rather it is an axiom that there is such).

This indicates moreover that the following can reasonably hold:

\neg(\exists x)(\text{Bel}(P,\text{Bel}(P,x) & \neg x))

at least if we interpret for the moment Bel(P,b) as saying P believes b 'in the current time period' and b is readily accessible to P's immediate thought, so that P would respond 'yes' when asked about b's truth. Thus existential quantifiers don't pass through belief predicates. This also holds for universal quantification:

(\forall x)(\text{Bel}(P,A(x))
isn't equivalent to

\[ \text{Bel}(P, (x)A(x)) \]

since the former asserts the existence of many beliefs held by \( P \), while the latter just one (universal) belief of \( P \).

What then can we say about belief-inference? Many candidates for belief schemata come quickly to mind, and are almost as quickly put in doubt. We will soon present a general framework for analyzing belief on a principled basis as part of a theory of mental behavior. But for the moment we shall speculate a bit. The following have at least some intuitive plausibility, although we shall argue that more properly they should be referred to beliefs currently in focus of attention. Here \( t \) is a time parameter, indicating a time at which the corresponding belief is held.

\[ \text{Bel}(Q, x \& y, t) \rightarrow \text{Bel}(Q, x, t) \& \text{Bel}(Q, y, t) \]

\[ \text{Bel}(Q, x, t) \& \text{Bel}(Q, y, t) \rightarrow \text{Bel}(Q, x \& y, t) \]

\[ \text{Bel}(Q, x, t) \& \text{Bel}(Q, x \rightarrow y, t) \rightarrow \text{Bel}(Q, y, t) \]

\[ \text{Bel}(Q, (x)(A(x) \rightarrow B(x)), t) \& \text{Bel}(Q, A(c), t) \rightarrow \text{Bel}(Q, B(c), t) \]

\[ \text{Bel}(Q, x, t) \& \neg \text{PIJ}(Q, x, t) \rightarrow \text{Bel}(Q, x, t+1) \]

where \( \text{PIJ} \) stands for 'perceived in jeopardy' (of being altered).

This latter bears on the frame problem, and depends on formalizing \( \text{PIJ} \) according to our earlier ideas about possibility and limited inference to be developed here, so we shall discuss it at some length. Roughly, unless through experience we learn that certain things are likely to change under certain conditions, then we assume they will not. We don't try to keep track of all changes, even ones we could detail with available knowledge, except when our focus of attention brings particular worries to mind. Thus in apparent contradiction to Hayes' 'connections' for keeping track of what changes, here updating is done on a goal-driven, need-to-know basis, rather than simulating the whole world as far as we know it. We just keep a little bit of the world in focus; indirect side-effects we simply fail to see and so we occasionally blunder and have to learn new 'jeopardies.'

A given \( \text{PIJ} \) will remain essentially forever (no truth maintenance) in memory, but soon finds itself having to compete with other jeopardies and denials of jeopardy. Thus we can recall and use an old and inappropriate belief, and later recall why it was inappropriate. Of course, how to do
this in detail is a very hard problem, but we think this is the problem to address rather than a logical formalism that doesn't account for the complex temporal cataloging of long- and intermediate-term memory, nor the attendant forgetting and confusion and conflicts in beliefs.

An example would be driving into town on a road with a drawbridge, to go shopping for half an hour, but knowing the bridge will be raised at noon for a boating procession that will last the rest of the afternoon. So we might rush to get across the bridge before noon, 'forgetting' that we will be stuck in town far longer than we intend. We might even plan specifically to get milk and be back home by one o'clock for lunch, and thereby have chosen the store in town rather than the one over the county line 45 minutes away. Clearly no highly sophisticated thinking is needed to find the flaw, yet tracing such plan-steps all the time would become impossible in all but the most trivial domains where the world is nicely restricted to a few salient entities and relations. Thus we don't worry about flying saucers attacking us on the road, or running out of gas, etc, etc.

Let us pursue a little further the Jeopardy notion. First of all, it is useful for us as scientists describing a system or being that keeps beliefs around until there is a mechanism activated to change them. Thus, if I saw Jim in his office at noon, I have the belief represented by

\[
\text{Bel}(\text{me},'\text{Jim in office at noon}',\text{noon})
\]

and I might retain this belief in two forms:

\[
\text{Bel}(\text{me},'\text{Jim in office now}',\text{now})
\]

\[
\text{Bel}(\text{me},'\text{Jim in office at noon}',\text{now})
\]

with varying degrees of appropriateness. The latter isn't very much at issue, but the former depends on my view of people's habits regarding remaining in offices. Presumably I will have learned early on that people don't remain in offices indefinitely, and in particular there is little sense in assuming much about Jim or anyone else an hour after seeing him/her in an office, unless there is more information such as that individual's schedule. Thus we can view the development of my thoughts regarding Jim in his office along the following lines: I retrieve the image of him in his office, as an on-going thing, and accept it as long as nothing occurs to me to upset it, such as the realization that a lot of time has passed and he's likely gone, or simply an introspective realization that the image is from the past and may no longer obtain ('well, I saw him there 30 seconds ago...').

In general, a belief that an image obtains in reality may be associated with a starting time (e.g. of observation) and granted open-ended extension until such
time as we see some reason to curtail it. Of course, with many things, we do have well-learned reasons for such curtailment. It may be more in line with our computational perspective to regard PIJ as meaning 'proved in jeopardy' so that -PIJ means a certain computation did not occur. However, although this clearly resembles fail-to-prove schemes as in PLANNER, it is crucial to note that here there is no implied effort to prove the item in question. Our application in Chapter IV will make some use of this idea, although we have not represented time explicitly there.

Thus we generally go right on believing x until we happen to find reason not to. We don't search for reasons to go on or not with x. If a 'jeopardy' is perceived regarding x, then we stop to consider it, and so -Bel(x) takes over temporarily, but not Bel(-x).

The following interlude provides a nice example for many of the points we wish to make:

Sally the Robot is to prepare to paint a box.

Her inventor, not trusting the robot's sophistication, insists she be given 'frame' axioms:

\[(x)(t) \text{ (HasPaint}(t) \rightarrow \text{HasPaint}(\text{GoTo}(x,t)))\]

\[(s) \text{ (At } (Sally,\text{Box},s) \rightarrow \text{At } (Sally,\text{Box}, \text{DropPaint}(s)))\]

Here s and t are time variables, x is a position variable, and GoTo and DropPaint are functions that return the time of completion of the indicated action. Sally has certain other information about the situation, defining the CanPaint condition and various operations that can be performed to this end.

However, we wish to argue Sally can get by without the frame axioms. In particular, the second frame axiom above is not something we should have to state: it seems clear that dropping the paint won't affect our position. While the contrary is conceivable, still we see no direct influence there, so we pay it no attention. It isn't that we look for a connection and failing to see one we then go ahead; we don't even consider it. Sally could make the same inference, or rather equally pay it no attention, by using the kind of default we urge here: a belief persists unless called specifically into question by something. Since nothing specifically calls present position into question when paint is dropped, then the position belief remains unchanged. Occasions in which remoter causes are noticed, must be triggered by appropriate memory of past experiences or other signs. This way, sometimes important results will be overlooked, just as in real life.
Now, this doesn't mean checking all possible consequences of given information to see if a belief is called into question. Our contention is that focus of attention is severely restricted, so that most of our beliefs simply don't come into the picture most of the time. The first frame axiom above also fits this pattern. Thus, although the following two beliefs are true when Sally is in the HasPaint condition, namely HasPaint itself and At(Paint,PositionX), only the HasPaint one is of interest to the problem and the other needn't be explicitly removed as a belief even though it contradicts the former when Sally moves (GoTo). Should the two surface together in Sally's attention, then Sally either can check whether in fact she has the paint, or (if this is just a planning stage) she can look for the stored fact about moving when physically connected to light objects. But this needn't be checked in advance!

On the other hand, if there is a NearPaint condition, the same argument could be applied to it, even though intuitively NearPaint(Sally,t) shouldn't yield

\[
\text{NearPaint}(\text{Sally}, \text{GoTo}(x,t))
\]

We can get this from a PIJ rule that says GoTo upsets nearness in general:

\[
\text{Near}(a,b,t) \& \text{Have}(\text{Sally},a,t) \rightarrow \text{PIJ}'(\text{Near}(a,b,\text{GoTo}(x,t)))
\]

Thus we say what (easily suspected) changes there are, but not the steady states (these are implicit and numerous). We note that this seems to cover the 'kaput' construct of (McCarthy & Hayes 1969). Now, we have substituted one axiom for some others but the new one is far more general, and also far less likely to be invoked in inessential situations. Still, the real world will require very many such rules, and organizing a scheme for bringing the appropriate ones into focus won't be simple; to our knowledge this has never been tried in a complex setting, and we will only hint at it in our examples here.

In fact, we often do silly things due to such unnoticed confusion arising from this difficulty. Consider Sally having blundered into the wrong room. If she notices a familiar object from that room and remarks that the room is indeed the one it is, still this may not happen to cause Sally to be upset, since the two 'facts' of the room's being two different rooms may not happen to be in focus together.

Now, once a discrepancy is noticed, Sally can consider what items in her short-term memory are amenable to simple verification, marking as temporarily forbidden those that fail the test. This is clearly very sketchy, and will
remain so until future efforts can implement such a mechanism for detailed study.

It is very striking that with all the interest in belief systems involving belief revision, almost no attention has been given to time, or indeed to any mechanisms for change of belief. However, this seems to be the crux of the matter. We don't simply need to know what to do when a revision is called for (truth maintenance), but rather we need to analyze how we get to know things in the first place: the process of thinking. Beliefs aren't just a large passive set of axioms, they are active at various moments, we reason about particular beliefs at particular times. Hence, we turn to focus of attention as a key point for our treatment of belief.

6. General hypothesis and model

Here we develop a restricted and computationally explicit mechanism giving rise to belief-inference. We don't want to postulate a closure of beliefs, but rather a construction of beliefs.

The model hypothesizes a three-part memory: long-term memory (LTM) for the bulk of stored beliefs, intermediate-term memory (ITM) for a record of recently accessed beliefs, and short-term memory (STM) for beliefs currently being accessed. Thus inference itself takes place in STM, which will be quite small. ITM serves both to keep an implicit stack of goals and to allow some readily accessible feedback about what has been accomplished and in what order; ITM then is larger than STM. Finally, LTM is very large and includes associational information ('Assoc') that largely determines what beliefs will be brought into STM. We will be able to get away with relatively simple and inexpensive inference rules in STM precisely because a great deal of control information is in LTM.

The main inferential mechanism we consider shall be modus ponens, applied in the form of inferring $B(c)$ from $A(c)$ and $(x)(A(x) \rightarrow B(x))$. (Here $A$ and $B$ are to stand for arbitrary predicates.) The usual notion of proof, whereby we may conclude, for instance, that a course of action will lead to a particular result, can involve a subtle chain of reasoning that we learn to perform as an algorithm at our
disposal. Normally we won't derive implications, except from other implications that are even longer.

Our general hypothesis can be stated as follows: Attention is what keeps computation tractable in a reasoning being; goals and associations are incredibly complex phenomena without any overall unique representational structure, but still are largely straightforward as inputs to a simpler ITM/STM mechanism for attention. Essentially all reasoning, planning, and acting stem from auto-inference, very local and undirected steps, whose apparent overall structure is 'guided' by the massive Assoc structure which embodies our technique for analyzing the world. Thus reason is decentralized, and even very specific algorithms are followed by this seemingly haphazard route. Yet auto-inference isn't totally trivial: it embodies some real semantics in the powerful rule of modus ponens. Thus we claim that there is some logic built into our very mechanism of thinking, a notion of implication or causation.

There will be no overall logic for searching proof trees or controlling inference, beyond the 'haphazard' Assoc mechanism of memory in general, or specific built-in or learned routines for specific situations.

The central formal mechanism we adopt for this purpose is the Assoc operator, which relates current focus and long-term memory. Our idea is that focus can be modelled as an extremely simple structure, perhaps with no more than half-a-dozen constituent beliefs at a time, with a modest backlog of previous focus states (ITM). Most of the complexity of mental processing will be conveniently shoved into the LTM whose structure we shall largely ignore. Our purpose is to isolate those fragments of processing that involve deliberate or conscious reasoning.

Most of the familiar kinds of inference will take the form of modus ponens applied to a general learned rule and a particular instance of the conditional hypothesis. We can even offer a definition of understanding this way. Thus we can argue that someone who doesn't protest at being told A, -B, and A --> B, must not have understood the assertions, or forgotten them, or not taken them seriously. But this would not be our conclusion about someone being told the Peano postulates and then also the negation of a non-trivial theorem about the integers.

ITM will be modelled as a queue, in which popping deletes the end altogether. The idea is that we keep track of very recent flow of attention, in temporal order, with an LRU strategy to keep the size reasonable. STM can be viewed as the front portion (incoming) of ITM. The size of ITM
isn't terribly crucial, and we can even think of it as open-ended but less readily accessible further from the front.

However, STM's size is a matter of key importance, for it determines what kind of reasoning is appropriately assigned to the items in current focus. Since we are defining STM (or focus) to consist of those elements that are automatically used in inference by either direct contradiction or modus ponens, it should be possible to present evidence from behavior, especially language usage, to pin down this size - and at the same time increase the plausibility of the STM/ITM distinction we are making regarding inference. Note that our STM is not necessarily directly related to the usual short-term memory of recall, since we are more concerned with inference; but connections are possible nonetheless. Our hope is that STM can be restricted to a very small number of beliefs, and still serve for nontrivial inferences based on automatic (essentially trivial) ones.

We shall use LTM, ITM, and STM as predicate symbols, where for instance, ITM(x) means x is in ITM (at some implicit time).

A rough schematic summary of our above discussion of 'auto-inference' then is:

\[
\text{STM}(A(c),(x)(A(x)\rightarrow B(x)) \rightarrow \text{STM}(B(c))
\]

\[
\text{STM}(A,-A) \rightarrow \text{STM('Jeopardy}(A,-A)')
\]

\[
\text{STM}(y) \& \text{ASSOC}(y,x) \rightarrow \text{STM}(x)
\]

where the latter expresses (clumsily, for time has not been represented) that if x is associated with y, the latter being in the current state of STM, then x is brought into focus in the next 'chunk' of inference. Of course, this hides the frame problem (namely, which are the appropriate things to associate), but in a form that we think is useful. 'Jeopardy' simply means something has gone wrong and some kind of error-recovery is needed; we won't try to present any general method for this, and we even suggest there may not be one beyond lots of special techniques for special situations.

The lack of direct clash with LTM may seem odd: a flagrant contradiction of current focus and well-known memory should provoke something. But not always, indeed only when ASSOC does just this! Consider having lent your favorite pen to John, and then believing he has it, but also having the belief that it is at home where you usually keep it. (This kind of forgetfulness may point to facts about the representation of knowledge, in particular that it may be more efficient not to keep updates in precise order.)
Additional rules need to go here, such as pattern-directed re-invocation of data from ITM to STM. Noticing similarity of elements in ITM by their syntactic form, and in a last-first manner, seems both psychologically plausible and computationally feasible, as well as useful for keeping contexts and goals in rough order. Here we won't go further than recalling goals and achievement of goals.

ASSOC itself clearly contains most of cognitive science, psychology, epistemology, etc, and we won't attempt to formalize it much further except for a few special selected examples.

Focus also drifts, via ASSOC, so that even though we may do just what the rules above say, to conclude 'John is dangerous' from 'John is Martian' and 'All Martians are dangerous,' still we may not go further to infer that 'John has wild eyes' from 'All dangerous beings have wild eyes,' if we haven't kept 'John is dangerous' in focus. We may catch the spirit of the ideas being presented quickly, as in a mental teaser, and so keep in focus those things that seem to allow ready inferences, or we may think of any number of other related things. Thus the effect of modus ponens is diluted by STM being small.

Consider another example: 'There's a green snake!' 'Green snakes are poisonous and terribly bitter to eat. 'I ate a delicious snake once.' So, it wasn't green, if the earlier generalization is correct. But we needn't bother to make this observation, if that isn't of interest, and if we DO then we may say so to point out to others what course our thinking has taken: 'So, it can't have been green...'. On the other hand, we are claiming here that 'that snake is poisonous' is auto-inferred from the first two statements, and indeed it seems highly rhetorical, even repetitive, to state it. It is as if the generalization was interpreted as being about the given snake as it was made, or we place our snake in the role of a generic snake in the generalization. We view this as evidence for modus ponens as an auto-inferential rule, and against contrapositives.

As a final example in the above vein, consider the following two conversations:

1. JOHN'S OUT THERE IN THAT BLIZZARD!
2. He's an Eskimo, and they're OK in blizzards.
3. SO, HE'S OK?
4. That's what I said! He's an Eskimo, and Eskimos can get through blizzards.
5. SO HE CAN GET THROUGH IT?

6. That's what I just said, damn it!

1. JOHN'S OUT THERE IN THAT BLIZZARD!

2. He's an Eskimo, he's OK in blizzards.

3. ESKIMOS ARE OK IN BLIZZARDS?

4. That's what I said! Don't worry about John, Eskimos know what to do.

5. SO JOHN KNOWS WHAT TO DO?

6. That's what I just said, damn it!

These seem again to support the idea that we use modus ponens as if the instance were temporarily in the role of generic, so that 'he's OK in blizzards' above really is read as being about 'John' in role as generic Eskimo. This would seem to explain both that conversation and the first one in which the general case ('they're OK in blizzards') is given yet the instance is naturally understood ('he's OK').

We offer a test for our modus ponens rule as follows: given A, ¬B, and A→B (in order of presentation) and also ¬B, A, and A→B (to other subjects), and similar combinations of two or three of these, is a clash noted, and of so is it between A and ¬A, or B and ¬B? Our hypothesis predicts that in the absence of goals to the contrary, it will be B and ¬B, i.e. we don't invoke arguments with contrapositives (¬B → ¬A) unless that happens to be called to mind for some purpose, such as questioning the truth of A.
CHAPTER IV
GETTING ON IN THE WORLD

1. Introduction

We will make use of Raphael's study of the frame problem (Raphael 1976) with the example of a robot in a room having various objects. We shall define an Assoc operator for the robot (still dubbed 'Sally') and show how Sally's attention remains fairly reasonably directed toward solving problems contained in her LTM/ITM. This has been implemented as a program in QLOG. Some programming details are included in Appendix C.

Our intention is to provide evidence for our General Hypothesis, to wit: ITM/STM as we have outlined it serves as a ready-made general purpose agent, of arbitrary smartness and adaptability according to the LTM information (beliefs and associations) that is its fodder. Moreover, STM/ITM although 'universal' in its logical capabilities, doesn't spread itself too thin; its heuristic is in effect to be rather blind, trusting to experience that the important things to watch for are already known, or will result in errors or even disaster leading either to a new thing to watch for in the future or total destruction. (This has a nice evolutionary ring to it.) We also wish to argue that for this kind of world-dealing, it is nearly indispensable that there be cognition, i.e., aboutness, self-modelling, etc., and there also be a certain puzzlement about goals, i.e., focus isn't training single-mindedly on just one problem but instead can shift around somewhat arbitrarily so that things get done, if at all, only when circumstances don't present too much interruption, within or without, including getting caught up in wondering about one's own goals. This latter feature would seem to be a necessary consequence of allowing self-reflective thought, since one's own intentions then become both subject and object.

Our efforts fall into several categories. We must provide Sally with general knowledge, a sort of naive physics of rooms. We need to specify Assoc so that 'suitable' things come to attention, yet not in an ad hoc manner. We must also provide some means of interaction with her world, both affecting actions and noting situations. Finally, some beliefs must pertain to general methods for
attacking and reasoning about problems. Note that 'meta-knowledge' of the sort 'I should already have recalled such-and-such if I knew it,' readily fits into this scheme via a goal (recall such-and-such) and a belief about LTM organization (possibly false, but highly usable) within that structure itself.

The category of associations will necessarily be rather large, if we are to avoid trivial situations that illustrate little of the trap embodied in the frame problem. Moreover, we will need a 'clock' of sorts, so that changing information can be traced and compared to goals; even when a belief is changed the old belief is remembered (and sometimes confused with the present). We will not treat this in any sophisticated way in the present system, however. Essentially, we will let ITM-position furnish this, namely, the beliefs more recently occurring in STM will be stored nearer the 'front' of ITM.

All this is a fairly ambitious outline; we shall necessarily be sketchy about it, as a full treatment would be a major research undertaking in itself: the design of a functioning system in a non-trivial domain with cognitive overtones. Nonetheless, in a sense our intention is to show that this is trivial, i.e., an essentially rather simple mechanism suffices for getting around in the world, even 'gracefully' and adaptively, if the appropriate memory structures are available. The cognition itself, if we may so style it, is subject to rather simple restraints, namely ITM/STM.

There is an obvious objection here, that we have put everything into LTM and so of course it is easy to describe a trivial mechanism for the trivial leftovers. However, our point of view is that behavior can have innumerable special methods for dealing with special situations and even unspecial situations. Rather than look for one general problem-solver for all problems, we allow any methods whatsoever that Sally may be taught or come across, and we concentrate then on how it all fits together in one attention-focussed entity. In effect, when we learn new techniques and even new goals, we change and yet there is a continuity in our past and present as we have focussed our attention over time. It is this that allows us to learn and feel our way through situations in which goals aren't always clear or are subject to unpredictable change as our attention is distracted by our own memories as much as by the environment. Goals can come and go. If a goal G is present, and if a method M comes to attention for reaching G, we apply it as a routine (itself likely involving further attention-inference steps), often forgetting G in the process.
We claim further that this 'freeness' is a tremendous plus for real-world problem-solving since often being flexible allows a wholly new view of a situation and old goals become modified or restated in more tractable fashion. This would seem to be very difficult for a robot with a prespecified goal of a particular form that can't be fit into a broader but heretofore unknown context. Of course, this also means proceeding much more slowly than with a ready-made algorithm for a specific domain, just as arithmetic is accomplished faster and more accurately by standard programs than by people. But this is something the cognitive side of artificial intelligence will gladly live with.

2. Axioms

We have then the following categories of axioms to fill out, at least partially:

1. methods
2. meta-knowledge

Sally will have two actions she can perform. They are Transport(x,y) and Take(x); formally these will be predicates which are true when the indicated action is completed. In the former case x is an object and y another object to which x may be transported, and in the latter Sally may grasp x. Often we will need to use sentences formed from these predicates in quoted form inside other predicates, such as

\[ \text{Achieved(Transport(book,shelf))} \]

in which we have left quotes off for ease of reading.

When an action is a Goal and when it is Enabled, she Causes it to occur. More specifically, some sentences will be designated 'Conditions' to indicate that their validity can be verified at will by Sally by means of performing a 'Check' operation (intuitively: 'looking'). Other sentences are 'Elementary' if they are actions Sally can perform at will once the specified preconditions are met; 'Enabled' applies to actions for which this has been done. 'Take(x)' is elementary; 'Transport(x,y)' is not, requiring
first the conditions 'Taken(x)' and 'Freepath(x,y).' Although time is implicit here, and is of obvious importance, we have elected not to represent it explicitly in the present application, for reasons of simplicity. 'Result(x,y)' means y results immediately upon completion of action x. Finally, 'Cause(x)' triggers via LTM an inference of the belief 'Expect(x)' which then leads Sally to search her STM looking for the expected resulting action.

Methods

\[ \text{Goal}(A \land B) \rightarrow \text{Goal}(A) \land \text{Goal}(B) \]

\[ \text{Goal}(G) \land \text{Imply}(H,G) \rightarrow \text{Goal}(H) \]

\[ \text{Elementary}(x) \land \text{Goal}(x) \rightarrow \text{Goal}(\text{Enabled}(x)) \]

\[ \text{Result}(y,x) \land \text{Goal}(x) \rightarrow \text{Goal}(y) \]

Meta-knowledge

The following rules involve Sally's noting (as a belief in her STM) that certain beliefs are in her STM. The first two are her version of quotation and un-quotation: if she

\[ \text{Infer}(x, \text{STM}(x)) \]

\[ \text{Infer}(\text{STM}(x), x) \]

\[ \text{STM}(x) \land \text{STM}(-x) \rightarrow \text{Jeopardy}(x, -x) \]

Clock knowledge:

The first 'clock' rule in the present system simply accounts for the need to get some feedback from actions after they
are performed. We have allowed the inferential process to supply the time parameter implicitly, i.e., Sally must think a little to derive 'Expect(x)' from 'Cause(x)' and therefore the expecting will occur automatically after the causing. This works for the current examples, but should be made explicit in more ambitious settings. The last two rules indicate when something is wrong with incoming information, either not what was expected or nothing at all. Again, this is not tied to any error-recovery methods at present.

\[
\text{Cause}(x) \rightarrow \text{Expect}(x)
\]

\[
\text{Expect}(x) \land \neg \text{STM}(x) \rightarrow \text{Jeopardy}(x)
\]

\[
\text{Check}(x) \land \neg \text{STM}(x) \land \neg \text{STM}(-x) \rightarrow \text{Jeopardy}(x \lor -x)
\]

**Primitive I/O:**

\[
\text{Elementary}(\text{Transport}(x,y))
\]

\[
\text{Condition}(\text{Freepath}(x,y)) \land \text{Condition}(\text{Taken}(x))
\]

\[
\text{Enabled}(\text{Take}(x))
\]

\[
\text{Result}(\text{Transport}(x,y), \text{On}(x,y))
\]

\[
\text{Goal}(\text{On}(\text{book}, \text{shelf}))
\]

\[
\text{Taken}(x) \land \text{Freepath}(x,y) \rightarrow \text{Enabled}(\text{Transport}(x,y))
\]

Below is a list of Sally's long-term memory associations, in the form (assoc x y ... z) where the last (right-most) argument to assoc is the one to be brought into STM when the other (left-hand) arguments are already in STM. The predicate 'Imply' states that the last argument is implied by the others. The notation we are using is Lisp-motivated, and is close to that of the program version of Sally. For example, the first rule below is interpreted as "if '(stm x)" is in STM, then the belief 'Imply (stm x) x' is brought into STM in the next step."
1. (assoc (stm x) (imply (stm x) x))
2. (assoc (expect u) (imply (expect x) (not (stm x)) (jeopardy x)))
3. (assoc (freepath u v) (imply (taken x) (freepath x y) (enabled (transport x y))))
4. (assoc (goal (take u)) (enabled (take x)))
5. (assoc (goal (take u)) (elementary (take x)))
6. (assoc (goal (taken u)) (result (take x) (taken x)))
7. (assoc (infer u v) (imply x (infer x y) y))
8. (assoc (check x) (infer (not x) (stm (not x))))
9. (assoc (check x) (infer x (stm x)))
10. (assoc (cause u) (imply (cause x) (expect x)))
11. (assoc (check u) (imply (check x) (not (stm x)) (not (stm (not x))) (jeopardy (or x y))))
12. (assoc (goal (freepath u v)) (condition (freepath x y)))
13. (assoc (goal (taken u)) (condition (taken x)))
14. (assoc (and u v) (imply (and x y) y))
15. (assoc (and u v) (imply (and x y) x))
16. (assoc (goal (enabled (transport u v))) (result (taken x) (freepath x y) (enabled (transport x y))))
17. (assoc (goal (transport u v)) (elementary (transport x y)))
18. (assoc (goal (on u v)) (result (transport x y) (on x y)))
19. (assoc (on book x) (goal (on book shelf)))
20. (assoc (goal (transport u v)) (result (taken x) (freepath x y) (enabled (transport x y))))
21. (assoc (goal u) (condition u) (imply (condition x) (goal x) (check x)))
22. (assoc (goal u) (result v w u) (imply (goal x) (result y z x) (and (goal y) (goal z))))
23. (assoc (goal u) (elementary u) (imply (elementary x) (enabled x) (goal x) (cause x)))
24. (assoc (goal u) (elementary u) (imply (elementary x) (goal x) (goal (enabled x))))
3. Inference and focus

The rules for use of Assoc and ITM are as follows. First, we take the size of STM it to be 8. The rules then are simple: each set of 8 Focus elements trigger all possible inferences and Assoc's (usually there aren't many) thereby pushing these into Focus and 'popping' older elements further into ITM.

MPO (modus ponens) will be used in a rather liberal sense: if A & B ... & K -> W and A,B,...,K are in focus (STM), then W comes into focus. Also, MPO binds variables automatically as follows: A(x) -> W(x) and A(c) yield W(c).

Once MPO has been used to exhaustion on the current STM we then form all indicated associations.

Now, if the above produces no additions to STM, as is often the case, then ITM is queried for the most recent Goal(x) for which x is not in ITM or STM.

In more algorithmic detail:

1. Let Q be a temporary queue.

2. Place in Q all elements of ITM of the form (...) where Goal(...) is in current STM. This prevents Sally from trying to achieve goals already achieved.

3. Place in Q all consequences of using MPO with current STM for hypotheses.

4. Next, all associations in LTM are queried for ones that can be triggered by current STM (not Q) where if Goal(...) is a left-hand argument, then (...) must not be in STM. Right-most arguments are entered in Q.

5. Any element of Q of the form Goal(...) where (...) is already in STM, is deleted from Q.

6. If Q above is empty, i.e., if the above steps yielded nothing, then ITM is searched from the present back until an element Goal(...) is found such that (...) hasn't been encountered in ITM already. Then Goal(...) is placed in Q.
7. Q is now pushed onto the end of STM, thereby popping any elements of STM that overflow the canonical STM size (for now, 8).

8. Q is deleted.

9. Repeat forever.

If external information is forthcoming, it is allowed into STM at any time. Checks are answered in the next chunk of STM, i.e., at the time of STM-search as illustrated in the next section.

4. Trial runs

We will now sketch a sequence of STM states for Sally as she proceeds within the room. We provide her with occasional perceptions (e.g., visual) in the form of processed beliefs. We endow her with a permanent goal of keeping any books on the shelf. However, this may fail to be triggered, or maybe forgotten, as events proceed.

The following consists of the beginning, middle, and final portions of a sequence of STM states for Sally (running as a program) as she solves the problem of getting a book from its original position on a table, to a bookshelf. Each portion should be read from top down, i.e., the first elements to appear in STM are given at the top. Sets of new beliefs that come into STM in the same inference stage are surrounded by xxxxx's. Thus Sally begins with only one belief in STM: (on book table); then she gets one more belief: (goal (on book shelf)), etc. The last addition to STM is '(cause (transport book shelf))', which can be performed after she has '(taken book)' in intermediate steps not shown here, and checked '(freepath book shelf)'.

Checking involves information about Sally's state of information, to wit, whether she is aware of certain expected results from external sources (such as seeing no obstacles are between the book and the shelf). She expects to get an answer in STM one way or the other: (freepath book shelf) or (not (freepath book shelf)). If she gets no answer at all, then something is wrong and she should know this. The 'jeopardy' predicate records this. By invoking 'STM-search' she gets the appropriate answer placed in STM automatically. This means she can tell, on certain occasions, whether a given belief is or is not present in her STM. Thus she has an internal version of quotation and
1. (on book table)

At this point Sally has only one belief in STM. Whatever she is supposed to have been thinking just before can be considered either to have passed into ITM already, or left in STM but unrelated to books. Thus (on book shelf) is the only element of STM that will trigger inference or recall from LTM.

2. (goal (on book shelf))

This rule has been brought into STM by assoc 19 of section 2 on axioms (pp. 139-141). Thus at this stage STM consists of (goal (on book shelf)) and (on book table), in that order, with therefore (on book table) closer to being 'popped' out of STM into ITM.

3. (result (transport x y) (on x y))

This is brought in by assoc 18.

4. (imply (goal x) (result y x) (goal y))

5. (goal (transport book shelf))

This is the first instance of MPO. Here it is operating on the previous three elements in STM, to bring in one more belief. This is not a recalled belief, but an inferred one. At this juncture, STM has five beliefs starting with the original one; none has been lost to ITM yet.

6. (result (taken x) (freepath x y) (enabled (transport x y)))

7. (elementary (transport x y))

For the first time more than one association has been triggered at once: 17 and 20. Since in the
present implementation LTM is searched linearly (from 1 to 25) this yields a fixed order to elements coming into STM 'at once.' Clearly some sort of parallelism and indeterminacy would be more realistic. Notice that now STM has seven elements.

8. (implies (elementary x) (goal x) (goal1 (enabled x)))

9. (implies (elementary x) (enabled x) (goal x) (cause x))

Two more beliefs have come into STM (from assoc 23 and 24), forcing one of the previous seven into LTM: (on book shelf).

10. (goal (enabled (transport book shelf)))

Another instance of MPO. Now the early belief (goal (on book shelf)) has been popped to LTM and thus 'lost' as far as playing a role in future inferences, until such time as it is recalled. Fortunately, another (sub-) goal is still present: (goal (transport book shelf)). However, it too will soon be 'lost' and not retrieved until much later.

11. (implies (goal x) (result y z x) (and (goal y) (goal z)))

From assoc 22.

12. (and (goal (taken book)) (goal (freepath book shelf)))

(MPO)

13. (implies (and x y) x)

14. (implies (and x y) y)

From assoc 14 and 15.

15. (goal (taken book))
Here go roughly 15 stages of STM inference in which Sally notes that 'freepath' and 'taken' are conditions, that 'taken' results from 'take,' and that these can be checked.

'S' trigger quite a few associations: 8, 9, 11, two of them twice due to the two checks above.

Stm-search is not really essential as an element of STM; 'check' already attests to the need to do some sort of quotational inference. But we have made it occur so that the self-referential character is more explicit. It is not an inference or a retrieval.

These are from 'external perception.'
28. \( \text{Impy} \times (\text{Infer} \times y) \ y \)

\( \text{From assoc 7} \)  
xxxxxxxxxxxxxxxxxxxxx

29. \( \text{Impy} (\text{taken} \times) (\text{freepath} \times y) \ (\text{enabled} \ (\text{transport} \times y)) \)

30. \( \text{stm} (\text{freepath book shelf}) \)

31. \( \text{not} \ (\text{stm} \ (\text{taken book})) \)

The first of these three is from assoc 3. The other two are the result of stm-search for the indicated beliefs.  
xxxxxxxxxxxxxxxxxxxxx

Here go about 15 stages of STM inference in which Sally recalls from ITM the goal of taking the book, proceeds to do so, recalls from ITM the goal of transporting it to the shelf, and then recalls from LTM that 'taken' and 'freepath' are preconditions for this.

32. \( \text{Impy} (\text{taken} \times) (\text{freepath} \times y) \ (\text{enabled} \ (\text{transport} \times y))) \)

33. \( \text{taken book} \)

The first of these two is from assoc 3; the second is external.  
xxxxxxxxxxxxxxxxxxxxx

34. \( \text{enabled} \ (\text{transport book shelf}) \)

(MPO)  
xxxxxxxxxxxxxxxxxxxxx

35. \( \text{goal} \ (\text{transport book shelf}) \)

Here we see a goal that has been recalled from ITM. This occurs now because all inferences and associations have been exhausted, STM is static.  
xxxxxxxxxxxxxxxxxxxxx

36. \( \text{result} \ (\text{taken} \times) (\text{freepath} \times y) \ (\text{enabled} \ (\text{transport} \times y))) \)
37. (elementary (transport x y))

These mimic exactly the sequence triggered by the above goal near the beginning of this entire trial.

38. (imply (elementary x) (goal x) (goal (enabled x)))

39. (imply (elementary x) (enabled x) (goal x) (cause x))

(ditto above comment)

40. (cause (transport book shelf))

Here we see an action 'in action,' as Sally completes her task. At this point she proceeds to 'expect' the result, 'checks' it, and (if she sees the book on the shelf) is free for another day's work.

The point we are illustrating is that apparently purposeful and coherent behavior can arise without explicit representation of centralized control/supervision, other than the little afforded by auto-inference. The rest is contained in general beliefs stored in LTM.

We have made the size of STM here 8, somewhat arbitrarily. Experiment has shown this to be roughly optimum, at least for the present application. That is, the same program except with STM of 4 elements failed to get very far at all before starting to loop. This was due to inability to get all the needed hypotheses into STM at once before losing the goal these hypotheses could enable. On the other hand, using 16 elements in STM gave no ascertainable advantage: the (cause (on book shelf)) result actually required slightly more stages of STM-updating than with 8 elements. It is probably a wild speculation, but one pleasant to make, that this bears on the well-known measures of human short-term memory as having 7 (plus-or-minus 2) 'chunks' of information (Anderson 1976). (With 5 elements Sally almost made it to (cause (transport book shelf)) but couldn't quite get the final inference. With 6 she got it, apparently as fast as with 8.)

Another comparison with humans is interesting: if we assume each inference stage takes 1/10 of a second, as seems roughly true for many high-level cortical processes in people, then Sally did the entire operation above in about
4.5 seconds (45 STM stages, at about two new beliefs per stage). That's not bad at all, though slower than we would manage it! Considering that practice makes a lot of this automatic in us, Sally has done quite respectably indeed. (It would be interesting to see how long it would take a person to perform a similar but unfamiliar task.)

Now to relate this to the frame problem. Suppose Sally drops the book while transporting it to the shelf. Then her expectation of seeing it there will be disappointed and the goal (on book shelf) won't be achieved. She may revert to (taken book) momentarily in trying to achieve it again, but since this is a Condition and hence Check-able, she'll see this too is false (if Checks are made to override other beliefs). The (taken book) element isn't lost from memory, but it won't be active since every time it may be recalled it will just as quickly be dismissed. The (on book table) memory may also eventually return and be dealt with similarly.

Thus beliefs are dealt with on a need-to-know basis, and false beliefs generally are simply allowed to lie around in memory. ITM holds the information that these conditions were changed, and in readily accessible form, but not that, say, (inCenter book room) also changed or that (inCenter table room) did not change. As we stated in Chapter III, this is viewed according to our approach as discernable by

PIJ rules that are invoked when recalled and not sooner, by the same mechanism on which all Sally's reasoning is based. (Whether this can succeed in complex domains remains to be seen.) If Sally really can't tell what to do from simple ITM recall or from checking facts directly, she will need more elaborate plans and auxiliary STM ('pencil and paper') much as a person does.
CHAPTER V
CONCLUSIONS AND FUTURE WORK

We conclude that it is possible to give a consistent and intuitively satisfactory treatment of truth, sufficient for a rich expression of syntax and semantics, including the self-referential cases, and which via reductivity has a computational significance.

Thus truth-labels have in a finite and length-bounded domain a computationally effective character. The usual paradoxes then simply fail to terminate when so computed, due to looping forever, but this feature can be exploited only when explicitly recognized in the underlying representational mode such as we have urged here. It is highly satisfying that this can be done in a classical first-order setting. Other approaches to 'decision logic' where care is taken to allow lack of decision as distinct from negative decision, are found in (Hayes 1975), (Levesque 1979), (Scott 1976), and various papers in (IJCAI-79). However, to our knowledge, none of these achieves the expressive power and naturalness of our method.

Moreover, such an approach allows the careful development of a belief and inference system that models the processes of inference limited to small portions of the entirety of beliefs present in the system. This provides an apparently fruitful method for approaching some of the standard problems of reasoning addressed in artificial intelligence.

In particular, we have shown that epistemic possibility and its associated issues of the frame problem and default reasoning can be expressed and at least to some extent dealt with naturally along these lines. Whether factual possibility can receive equally satisfying treatment this way is far less clear, and requires further work.

Since in principle self-reference can occur in computational systems used in artificial intelligence, and since self-reference will almost certainly have to occur in machines that communicate about their own behavior and intentions (e.g., 'I only use English.' or 'My statements are consistent'), it is well that designers of representation systems begin thinking about this issue. This may seem outlandish to some, since such systems generally deal with finite domains and relatively few levels of reference of one sentence to another. But in fact self-reference occurs easily and naturally in a finite domain and without constructing complex iterations of
CONCLUSIONS AND FUTURE WORK

sentential reference, viz., by simple consistency assertions. Indeed our derivation of paradox via RR or Consis in a too liberal usage of Tr, makes no use of the implicit infinitude of constants or the unbounded potential length of formulas.

It seems likely that our treatment of truth can be used in foundational questions in set theory, since a traditional difficulty there has been paradox arising from the comprehension principle. A first step would be to define classes a la Frege, and then to specialize to sets in the usual ranked hierarchy. This would augment work begun by Gilmore.

Recalling Sally's efforts to place a book on the shelf, it is easy to imagine a slightly more complicated situation in which she might not be able to get all the relevant beliefs in focus to draw the needed inferences. But this is not a severe criticism, since we often have such trouble too, and resort to pencil and paper to extend our capabilities. This presumably could also be built into Sally, together with a feature for recognizing when she's in a loop (nice little ones, for she won't solve the halting problem!). Then she could get out of the loop by using paper for awhile. Furthermore, she should have some power of selection over what stays in STM, such as keeping goals and beliefs relevant to them in focus. This would allow a sieve-like function, letting lots of beliefs pass through STM until enough useful ones are found.

A major undertaking that would extend this work is that of incorporating time more explicitly in beliefs themselves. An even more ambitious undertaking, but necessary for the full development and testing of our hypothesis concerning auto-inference, is to allow the Assoc mechanism to be provided by a learning process rather than 'deus ex machina.' Presumably when LTM becomes large, weights or probabilities will be necessary to provide a selective mechanism among competing associations. Also, large axiom sets as in the 'naive physics' of (Hayes 1979) could be very useful robotics-type applications of these methods; this would require very fine tuning and presumably would mesh with the use of learning. In addition, some means for switching over from conscious to reflex forms of behavior as in (Russell 1981) seem appropriate here.

Our results suggest that meaning be analyzed in terms of an interplay of representations within a reasoning entity. A symbol A for an object B can be a symbol only with respect to some entity E that so uses it. And the determination of its being so used must be given by E and its relationship to B. Our study of quotation suggests the following: A means B for system E, if S has both a token A used in some appropriate conjunction with the (usually
external object B, and also a name for the token A, such as 'A', so that the usage of A to denote B is recorded in E. In a picture (inspired in part by (Minsky 1968)): 

We have been pretty vague about how B comes into this, but that is an aspect seemingly less related to the cognitive side of the picture: how hands and arms and physical signals interrelate. There is one interesting point about this, though: The way in which the world is a model of the statements in E's head isn't at all arbitrary. For at least some of E's symbols have a meaning pinned down fairly uniquely, namely, the voluntary affectors of E's body. This may even make it possible to pin down a large portion of the world that E interacts with.

We have then come full circle in our investigation, back to our initial contention that beliefs in a system are best viewed in terms of strings expressing their own relationship to the world in which they exist.
APPENDIX A

PROOFS

Let L be a syntaxal language, which for illustration we shall take to be that of Chapter II with Hollerith quotation. Consider an extension of L containing the predicate symbol Tr, of one argument. It is natural to consider as an axiom for a theory with language L, the following for strings x naming closed formulas e1...en:

\[ \text{Tr}(x) \iff e_1 \cdots e_n, \]

and which we ungrammatically write as \( \text{Tr}(x) \iff x \). However, this leads to Russell's heterological paradox, as follows: we will construct a formula \( R(x) \) that intuitively says that (the formula named by) \( x \) does not apply to its own name as argument, and then \( R(R) \) will appear to assert its own denial. This will involve precisely a meaning shift via Tr and '-', and the result is an equivalence of the form

\[ R(R) \iff \neg \text{Tr}(R(R)) \]

First we give meaning to Sub by the (defining) formula

\[
\text{Sub}(y,t) : \\
\text{Concat} ( \text{xval}(t,\text{Entry}(1,y)), \ldots, \text{xval}(t,\text{Entry}(27,y)) )
\]

where \( \text{xval} \) itself is a new function symbol of two arguments \( (x,y) \) corresponding to the function that returns \( x \) if \( y \) is the letter 'x', and returns \( y \) otherwise. In other words, \( \text{xval}(x,y) \) 'evaluates' \( y \) by using \( x \) for \( y \) if \( y \) is 'x'. Formally,

\[
(x)(y)(z) \ ( \text{xval}(x,y) = z \iff \left( z = x \land y = 1 \land x \right) \lor \left( z = y \land \neg y = 1 \land x \right) ) .
\]

Then \( \text{Sub}(y,t) \) 'returns' the result of using the term \( t \) for each 'x' in the first 27 units of \( y \). We use 27 terms above since our single application below will call for precisely that number. Now let \( R(x) \) be the formula

\[
- \text{Tr}( \text{Sub}(x,\text{quote}(x)) ) .
\]

Thus \( R(x) \) says that the formula resulting from substituting into (the first 27 units of) the string represented by \( x \), that very string's name for all occurrences of the letter \( x \), is not true. Thus \( x \) stands for
a predicate with a variable (argument) 'x', and R(x) says that predicate is false when applied to itself as a string. Note R(x) is itself an abbreviation for a string with 27 units, where we recall quote(x) itself abbreviates a string, namely, Concat(Ln(x),x), of 15 units. We will write 'R(x)' to abbreviate this string-name beginning with 27: ...

Now if we form R('R(x)'), we get

- Tr( Sub('R(x)',quote('R(x)'))) )

This unpacks to

- Tr('R('R(x)')') .

Thus we have

RR <-> -Tr(RR)

where RR abbreviates R('R(x)'), and the proposed schema then gives

RR <-> -RR .

As a consequence, the proposed axiom schema for truth is seen to be inconsistent. (In (Cocchiarella 19??) it is argued that the intent of Russell's paradox is best expressed as R(R) rather than as R('R(x)'); but this seems to be a confusion of use and mention. See (Kapian 1973) and (Smullyan 1957) for other views on self-referential paradox.) We will then begin with a framework far weaker and slowly build up to a reasonable characterization of formal possibilities for truth. We have already established by the above, the following:

Theorem 1: A classical truth system (defined, p. 68) does not admit any formula A(x) such that for all closed formulas F the formula A(F) <-> F is a theorem. (This is a generalized version of Tarski's famous 'no truth-definition' theorem.)

Worse, we also have

Theorem 2: A classical truth system cannot have as theorems all formulas x --> Tr(x), where x ranges over closed formulas.

Proof: Otherwise we would have RR --> Tr(RR) and since RR <-> -Tr(RR) then RR --> -RR, whence - RR. From this and the hypothesis we get Tr(RR)
as well as $\text{Tr('RR')}$, contradicting the definition of a truth system.////

We note that this result is a very general (although 'weak') version of Godel's incompleteness result for extensions of $P$, Peano arithmetic. Namely, we have the following corollary:

**Corollary:** No classical truth system with rule $x / \text{Tr('x')}$. is complete.

**Proof:** The above theorem shows some sentence of the form $F \rightarrow \text{Tr('F')}$. not to be a theorem. But also its negation $F \wedge \neg \text{Tr('F')}$. is not a theorem since the rule would then give $\text{Tr('F')}$. from $F$.////

Note that this does indeed generalize Godel's result. In the theory $P$ and its recursive extensions, we indeed have $x / \text{Thm('x')}$. where $\text{Thm}$ is a Godelization of the attribute of being a theorem. (That is, the rule $x / \text{Thm(x)}$. can be used in forming proofs of theorems of $P$ from other theorems of $P$; perhaps 'proof-step rule' is a better name than 'rule of inference.' ) A proof of $x$ can be encoded as an integer, and a proof of $\text{Thm(x)}$. involves simply a (finite) calculation to determine properties of its various factors. Clearly $\text{Thm}$ is an example of a $\text{Tr}$ predicate in a truth system. However, to offset possible confusion in the future, we point out that this rule is subsumed by the first-order structure of $P$, and is not an independent rule.

Our next results resemble Godel's second incompleteness theorem.

**Theorem 3:** A classical truth system with rule $x / \text{Tr(x)}$. cannot admit as theorems all formulas $\text{Tr(x)} \rightarrow x$.

**Proof:** Assuming $\text{Tr(x)} \rightarrow x$. then since $\neg \text{RR} \leftrightarrow \text{Tr( RR)}$. we get $\neg \text{RR} \rightarrow \text{RR}$. hence RR and so $\neg \text{Tr( RR)}$. But from RR we also get $\text{Tr( RR)}$. a contradiction.////

Recall that a constructive truth system is one with rule $x / \text{Tr(x)}$. and axioms

\[
\text{Tr(x)} \rightarrow \text{Tr(Tr(x))}
\]
\[
\text{Tr}(x \rightarrow y) \rightarrow (\text{Tr}(x) \rightarrow \text{Tr}(y)).
\]

**Theorem 3':** In a constructive truth system, Consis is not a theorem.

**Proof:** Recall Consis is \((x) \neg \neg (\text{Tr}(x) \& \text{Tr}(\neg x)).\) We already have \(\text{Tr}(\neg \neg) \rightarrow \neg \neg).\) But then our rule would give \(\text{Tr}(\neg \neg) \rightarrow \neg \neg),\) and so the second axiom of constructivity above produces \(\text{Tr}(\text{Tr}(\neg \neg)) \rightarrow \text{Tr}(\neg \neg).\) Then using the first axiom of constructivity we get

\[
\text{Tr}(\neg \neg) \rightarrow \text{Tr}(\neg \neg).
\]

Now using \(\neg \neg\) for \(x\) in Consis, we get \(-\text{Tr}(\neg \neg) \vee -\text{Tr}(\neg \neg).\) But if \(-\text{Tr}(\neg \neg)\) then also \(-\text{Tr}(\neg \neg)\) from the above conditional. So \(-\text{Tr}(\neg \neg)\) follows from the disjunction. But this is also equivalent to \(\neg \neg\), and so another application of our rule would give \(\text{Tr}(\neg \neg),\) contradicting \(-\text{Tr}(\neg \neg).\)\]

The next theorem dashes hopes that we can avoid negative results as above by weakening the logic intuitionistically.

**Theorem 4:** Theorems 1, 2, 3, and 3' remain valid for intuitionistic truth systems.

**Proof:** The proofs of theorems 1, 2 and 3' are already valid intuitionistically. Theorem 3 uses \(-\neg \neg \rightarrow \neg \neg\), which is not intuitionistically valid. But we can use instead of the predicate \(R(x)\) in the formation of \(\neg \neg\), the predicate \(S(x): \text{Tr}(\text{Concat}(\neg, \text{Sub}(x, \text{quote}(x)))).\) Letting \(SS\) be \(S(S(x)')\), one checks then that \(SS \leftrightarrow \text{Tr}(S \neg S)\). So by denying the conclusion of Theorem 3, \(SS \rightarrow \neg S).\) But then we get \(-SS\), and so (by the given rule \(x / \text{Tr}(x)\) of Theorem 3) \(\text{Tr}(\neg S).\) This however is equivalent to \(SS\), contradicting \(-SS).\)

Now we turn to positive results. First we show that a constructive kind of truth is possible, namely by using inference rules instead of axioms.
Theorem 5. There are classical truth systems with rules of inference $x \rightarrow \text{Tr}(x)$ and $\text{Tr}(x) \rightarrow x$.

Proof: Peano arithmetic $P$ is such a system, using Thm for Tr, as we have noted earlier. We assume the natural numbers form a model for $P$ to get the second rule: if Thm(x) is a theorem of $P$ then Thm(x) holds in the natural numbers, so $x$ really is a theorem of $P$.

Theorem 6: Any consistent first-order theory can be extended to a classical truth system, with rules as above.

Proof: Simply adjoin a new predicate symbol Tr and iteratively include axioms Tr(x) for each theorem $x$. This will close at the first limit ordinal and satisfy the two rules given. We will never get a clash of Tr(x) and Tr(-x) this way, nor of Tr(x) and -Tr(x) since the only axioms bearing on Tr are already in Peano arithmetic where no such clash occurs.

Theorem 7: If $T$ is a consistent first-order syntactical theory then $T$ has an extension $T'$ which is a classical truth system with axiom schema Tr(x) $\leftrightarrow x$ for all positive formulas $x$.

Proof (modifying Gilmore as motivated by Kripke): Let MO be a model of $T$, with domain $D$. We can regard MO as determined by its true atomic formulas, i.e., these serve to interpret the predicate and function symbols. We will extend a model $M$ of $T'$, where $T'$ is a classical truth system with axioms those of $T$ plus the schema Tr(x) $\leftrightarrow x$ for positive $x$. We will do this by interpreting Tr in stages, initially (i.e., in MO) as the null relation, so that in MO we have -Tr(x) for all $x$. As we extend the applicability of Tr in further models $M_u$, we will be automatically determining new atomic formulas which are to hold. The idea here is that -Tr(x) isn't necessarily permanent unless we first have decided Tr(-x); the latter is regarded as definite once established, while the former may change as the sense of Tr grows.

Recall that for any formula $A$ of $T'$, $A^*$ is the result of passing '!' through to predicate letters, and through Tr's as well so that -Tr(x) becomes Tr(-x). It is then easy to verify that for positive $A$, $A \leftrightarrow A^*$.

Now, for any ordinal $\alpha$ for which $M_\alpha$ has been defined, let $M_{\alpha+1} = M_\alpha + \text{the set of } '\text{Tr}(x)' \text{ for which } x \text{ is true in } M_\alpha$. That is, we change the
Interpretation of Tr in Mu+l by making Tr hold for some additional strings.

This requires explanation. We suppose Tr to be part of the underlying language, so that Tr(x) does not hold in MO as noted above. Then in M1, for each atomic truth in MO, such as x=x, we get Tr('x=x') as an atomic truth in M1 by definition, whereas in MO we have -Tr('x=x').

For limit ordinals i, with Mu defined for u<i, let Mi = UMu (u<i), where again Mu is regarded as represented by the set of its true atomic formulas.

Now the underlying language will have some cardinality k, so also the number of formulas is k, and thus the sequence

MO c M1 c M2 c ... c Mu c ...

must at some ordinal e become constant: Mu = Me for all u > e.

Let M = Me. This is our candidate for a model of T'. (Note that T' has no non-first-order rules of inference, so that this shall be a model in the usual sense.)

Since our goal is to show T' is a classical truth system, we must show (Tr(x) & Tr(-x)) is not a theorem of T'. This will follow if indeed M is a model of T' and (Tr(x) & Tr(-x)) is false in M for all x.

First, to show M is a model for T', we need only verify the axiom schema Tr(x) \iff x for positive x, since the other axioms already hold by virtue of MO (and hence M) being a model of T. So let Tr(x) hold in M for some positive x. Then Tr(x) already holds in some Mu, since it is atomic and M = UMu (u < e). Assume u is the least such ordinal, hence not a limit. Then x* holds in Mu-1. But positive formulas, once true, remain so in our construction (this is a simple lemma) so that also x* holds in M. But x* \iff x (recall x is positive) so that x holds in M. Thus we have shown that Tr(x) \implies x in M for positive x. Actually we will see later that this holds even for non-positive x.

Now we turn to the converse: x \implies Tr(x) in M. Let x (positive) hold in M. Then x \iff x* and so x* holds in M. But M = Me = Me+1, and Tr(x) holds in Me+1 by construction. This gives the desired result.
We see then that \( \text{Tr}(x) \iff x \) holds in \( M \) for positive \( x \), and thus \( M \) is a model of the theory \( T' \). Moreover, \( \text{Tr}(x) \rightarrow x \) holds in \( M \) for all \( x \), and we will need this fact to proceed. So in the above argument for positive \( x \) we observe that if we had \( x^* \rightarrow x \) for all \( x \) in \( M \), then the identical reasoning would apply. But in fact \( x^* \rightarrow x \) for all \( x \) in \( M \). We see this as follows: Let \( x \) be \( \neg\text{Tr}(y) \), so \( x^* \) is \( \text{Tr}(-y) \). Then if \( \text{Tr}(-y) \) is true in \( M \), then \( (-y)^* \) holds in \( M \); but inductively assuming the above implication for fewer instances of connectives, quantifiers, and \( \text{Tr}'s \) than in \( x \), we get \( (-y)^* \rightarrow \neg y \), hence also \( \neg y \) holds in \( M \). Now if \( \text{Tr}(y) \) were true in \( M \), then again we would have \( y^* \) in \( M \) and thus \( y \) as well, contradicting \( \neg y \). So we have \( \neg \text{Tr}(y) \), i.e., \( x \), from the hypothesis \( x^* \). The more general case for arbitrary \( x \) follows similarly.

The above observation leads immediately to the conclusion that in \( M \), \( \text{Tr}(x) \rightarrow x \) for all \( x \), since we have \( x^* \rightarrow x \) as well as \( \text{Tr}(x) \rightarrow x^* \). Now we see that \( (\text{Tr}(x) \& \text{Tr}(-x)) \) is false in \( M \) since otherwise we would have both \( x \) and \( \neg x \) true in \( M \). Therefore \( \text{Tr}(x) \& \text{Tr}(-x) \) cannot be a theorem of \( T' \), and so \( T' \) is indeed a classical truth system. Note that by construction, \( x^* \iff \text{Tr}(x) \) in \( M \), and that the formulas (12) hold in \( M \) as well.

Corollary: Any theory \( T \) as in the theorem above can be extended to a classical truth system \( T'' \) with axioms \( \text{Tr}(x) \rightarrow x \) for all \( x \), and \( x \rightarrow \text{Tr}(x) \) for positive \( x \).

Proof: Use the model \( M \) in the construction above, to extend \( T \) to the theory of \( M \), i.e., \( M \) is already a model of the desired theory \( T'' \).//

Corollary: \( x / \text{Tr}'(x') \) is not a correct rule in \( T'' \).

Proof: Immediate from Theorem 3.//

Corollary: Any theory \( T \) as above, and its extension \( T' \), can be extended to a classical truth system in which not only do we have the schema \( \text{Tr}(x) \iff x \) for positive \( x \) inherited from \( T' \), but also the rule \( \text{Tr}(x) / x \).

Proof: In \( T'' \) we already have the rule \( \text{Tr}(x) / x \) by fiat due to the schema \( \text{Tr}(x) \rightarrow x \) for all \( x \).//

We have the open question whether this can be strengthened to include as well the rule \( x / \text{Tr}(x) \).
We can examine a similar rule that may shed more light on this, namely, $x / x^*$. Now $F \lor \neg F$ is an axiom, and hence a theorem, so by this rule $(F \lor \neg F)^*$ would be a theorem. But this is just $F^* \lor (\neg F)^*$. Applying this to $RR$ for $F$, we get $RR^* \lor (\neg RR)^*$, i.e., $\text{Tr}(\neg RR) \lor \neg RR$ since in fact $\neg RR$ is positive; but then the disjunction becomes simply $\neg RR \lor \neg RR$ which then reduces to $\neg RR$, whether in a definite or positive or reductive system. Hence we have $\neg RR$ as a theorem given the rule in question, in a truth system with any of the above three conditions. But in each of these cases this is contradictory, for immediately we get both $\text{Tr}(RR)$ (because this is just $\neg RR$), and either $\text{Tr}(\neg RR)$ from positivity or reduction, or else $RR$ from definiteness. We have then established

**Theorem 8:** A truth system with positivity, reduction, or definiteness cannot have the rule $x / x^*$.

Moreover, we readily see

**Theorem 2:** A reductive or definite system cannot have the rule $x / \text{Tr}(x)$.

This result follows from Theorem 8 and Theorem 3.
This chapter represents a rather severe departure from the rather concrete issues already addressed. We intend to push the notion of discrete computational processes beyond the domain of artificial intelligence into foundational questions in physics.

Having dealt at some length (though by no means exhaustively) with the notion of epistemic possibility in the context of planning and limited resources, we then turn to factual possibility as a last excursion. This has important connections with planning, to be sure, since in many situations we are really interested in whether something is in fact possible, for instance whether a bridge can collapse under certain kinds of misuse or certain weather conditions. Then we make every effort to be exhaustive in our investigation of the phenomenon in question. Indeed, factual possibility and ways to investigate it presumably are simply additional routines that we learn in our intellectual training. However, this kind of behavior does seem to have a distinct and highly separable character from the more casual everyday reasoning we have been looking into. We will not further pursue parallels between these except to note that a full-blown actor in the world must have both capabilities represented as part of the self-same set of beliefs. Thus we are turning away from the control issues predominant in Chapter IV, to a more epistemological set of concerns.

To start off our discussion we note that computability has been linked to notions of prediction as well as to the possible discreteness of space and time (Chaitin 1974), (Kreisel 1976), (Simon 1978). Also formal semantics for the notion of possibility and its relationship to the growth of knowledge have been linked to foundational questions in physics, and recently this has been given some technical backing by (Bressan 1972), (d'Espagnat 1979), and (Dishkant 1977). Thus we are not completely in isolation in our highly speculative endeavor.

The final version of possibility is the one relevant to scientific discourse, in which we explicitly look at the models underlying our everyday notions, ask about causal relationships implicit in them, and seek full logical implications of these. The material in this section will be speculative. Roughly, the idea is that if only a finite amount of information is available to a given process in a
finite time, then predictions of the future will in general be faulty since further information will begin to impinge on the situation later and play an unpredicted role in eventual outcomes. Moreover, the same holds for postdicting the past, for not all the history of a process can be stored in finite information in the present; so the past also is not fully determined. We think it may be useful to explore the senses in which past and present here are 'open to possibility'. We contend that neither formal nor epistemic possibility is in line with the intuitive or the scientific senses for this context. We find of interest here the work of (Bressan 1972), (d'Espagnat 1979), (Everett 1957), and (Wheeler 1978).

These ideas have an interesting connection with prediction. It has been suggested by (Popper 1957) that a kind of diagonal argument shows the impossibility of a general computational predictor of the future. We can amplify on this briefly. If we take $W(t)$ to be the state of the universe at time $t$ (seen by some fixed observer) then if $P(t) = (W(t), t)$ is a predictor yielding at time $t$ the predicted world $W(t)$ for time $t$, we can show that $t > t$ in general. For we simply construct a device $D$ which accepts as inputs the messages "ON" or "OFF" and responds by having its one light bulb be off or on, respectively. Then if $P(t)$ yields $W(t)$ in which D's bulb is, say, on, then we (or the predictor-mechanism itself) send "ON" to D which then turns off its light. This will eventually falsify the prediction unless time $t$ is never reached, i.e., unless $t$ is already past by the time the prediction is made ($t > t$) so that indeed D may well have been lighted at time $t$ as the prediction said.

This peculiar argument seems to say that there are strict limits to how fast a function such as $P(t)$ can be calculated. Perhaps we should not be surprised, since $P$, in order to reflect accurately what the world does, must somehow take account symbolically of all the steps followed by the world. Put differently, it appears we can speed up the calculation of $P$ only to the point of being as fast as the world itself ($P$ being part of the world) and thus that the actual course of events represents the fastest possible calculation of that very process. This hints that physical time be thought of as computational steps in the world's self-calculation, which in turn suggests that a discrete model of the universe may be very interesting indeed. Furthermore, there is implicit here the notion that we cannot model all the world in part of the world, i.e., that there is no superfluity or redundancy of information: An event occurs precisely when all its preconditions are met, and there is not time of waiting for the consequences to
work themselves out. Any apparent waiting for things to spin along in some Newtonian clockwork can on this view be better regarded as extending the context to further data that has begun to impinge on the situation.

But why then isn't factual possibility the same as formal? If we knew enough detail about the physical world, couldn't we regard possibilities therein as precisely those statements consistent with the facts (both laws and initial data)? If indeed the conditions surrounding, say, Skylab, make it physically impossible for it to fall (in the current scheme of events, assuming nothing unknown can intervene), then to be sure there seems no reason to distinguish the two kinds of possibility. But there is a further point: properties are held by objects, and it is such that lead us to speak of their possibilities: Skylab could fall if things were otherwise, for it has inertia, the ability to be pulled. So there is still something to be dealt with here.

Following a line examined by Bressan, we recall that properties are, at first glance, often understood in terms of interactions, e.g., an object's having mass \( m \) can be thought of as its having a certain potential for interaction with other objects with mass. But this potential itself seems to have little meaning unless we take seriously the possibility of such interactions occurring. And here arises the problem: what does it mean to say an interaction would occur in the right conditions, or is possible, if in fact it doesn't happen to occur? It appears that the only reason to insist that such interactions will occur when expected, is to attribute a feature to the objects in question apart from the mere happenstance events we observe. For otherwise we'd have no reason to expect a given pattern to continue, i.e., the next time conditions were 'right' we'd again see the same pattern manifested. But if the pattern of behavior is built into the objects as a continuing property even when not invoked, then this property isn't just a relationship between the object in question and other objects at the moments of 'right conditions,' but rather obtains in some further sense.

The 19th century logician/philosopher C. S. Peirce argued in his notion of pragmatism that possibilities must be in some sense real aspects of reality and not mere linguistic phenomena. He was concerned in particular with counterfactual hypotheses such as 'If \( X \) had been the case then \( Y \) would have occurred.' Here we suppose \( X \implies Y \) to be an ordinary scientific assertion about the world. Peirce observed that such statements, if interpreted as conditionals, express nothing at all of interest in cases where \( X \) does not obtain. But then our earlier statement should either be interpreted as vacuous or have another sense. Since we usually regard such statements as having a genuine and physically significant meaning, the problem is
of interest. Peirce suggested that we should regard the envisioned possibility of X as real in some physical-but-unspeclfied manner, and in an equally unclear manner this is to be 'followed' by Y.

Certainly this idea has some appeal, for we often feel emotionally bound to insist on the 'truth' of certain counterfactual hypotheses. For instance, suppose John claims 'If I had known you wanted a drink I'd have shared mine with you.' Since apparently John really didn't know, then any conclusion is justified from his supposed knowing, even that he still keeps the drink to himself. We may say that he is expressing the true existence of intentions or attitudes in himself: 'I know myself and my feelings and past behavior toward you, I was in a state of readiness to share with you.' This is at least a partial accounting for what we mean to express, and it involves no counterfactuals.

But will it also do for laws of science? We can say a system was in a state ready for such-and-such, even if the latter does not occur; then it will be true that in those cases where such-and-such occurs, the system performs as it was ready to do. But in other cases we now cannot claim (in this view) that the system would have done the same if the key events that did not occur had occurred. Does this matter? We usually imagine that systems have certain properties that endow them with the capabilities of performing according to given laws, and that these properties or capabilities are there even in those moments they aren't being utilized. Thus a concrete dam retains its holding power even when the water level is nearly zero, or so we normally think. To give this up would be a big sacrifice in our view of the world, yet at the same time that view is clearly incomplete in its account of the meaning of such statements.

As a further sample of our direction of thought, the statements 'Mars would have fallen into the Sun last year if its velocity had been zero,' and 'This mousetrap would have sprung shut if a mouse had nudged the spring-trigger,' can readily be regarded as true, but not because of default due to vacuity. We intend that these be factual statements about the actual constituent nature of the physical entities involved. But how is this to be explained? To say, as does for instance (Burks 1977), that this means that in some possible-but-imaginary world Mars does fall, or the trap is sprung, says nothing whatever beyond the fact that we can consistently form paper models that behave as we command: fall, or snap shut. They do so by our definition and not by real properties of mousetraps and planets, i.e., we aren't addressing the proper matters. The formal possibility of such depends on what we decide to take as axioms, e.g., planets fall..., mousetraps snap..., and this decision is based on our already believing these are laws with factual
meaning—clearly a circular definition! We wish to say that causation resides outside a mere sequence of observed events, in a prior property of the physical entities. We would like in particular to be able to say what it is that makes some possibilities real and others merely formal, or else to admit them all as equally real.

It appears we must either follow Peirce and try to give clearer sense to 'real-possibles', or grant that properties don't reside in objects but rather in interactions. In fact both of these alternatives have appeared in recent thought. First, modal logic has been provided with a semantic interpretation (due to S. Kripke) involving precisely the notion of possible worlds, and this has a well-understood structure. Essentially it involves considering all possible models of given axioms, i.e., all 'worlds' consistent with assumptions, indexed by a 'world-variable'. Also, this technique has been widely used in formal studies of natural language, in particular to analyze precisely such counterfactual hypotheses as we have seen. Moreover, Bressan has applied these ideas to the formalization of classical mechanics, in order to give definite meaning to 'possible measurements' apparently necessary in the definition of mass and other physical quantities. It then is of interest both to see whether a similar treatment would be of use in quantum mechanics, and to try to give an account of the 'possible worlds' that is not merely a formal convenience (i.e., to clarify Peirce's suggestion).

Second, d'Espagnat and others have recently presented evidence that in fact some apparent properties of particles obtain only at the time of measurement, and not before. This had long been envisioned as a possibility in the philosophy of quantum mechanics, with strong feelings on both sides of the argument, but now it appears to be an empirical matter to be given technical consideration.

Third, we note that in a sense the modal formalism and the idea of many alternative 'realities' can be combined with the thesis that any process can avail itself of only a finite amount of information in a finite time (suggesting that each process only 'knows' a limited amount about its past and future, since it can keep only limited records). Together these notions seem to say that the past and future are indeterminate, i.e., many pasts can have led to the (local) present, just as many futures can result (from these many pasts). In fact, Wheeler has suggested that there is no more meaning to the past than what little of it is encoded in the present. Since this philosophic observation applies equally to the world of classical physics, it would appear that here can be found an interpretation of Bressan's formal structures, i.e., as more than mere consistency statements. If so, then the possibility of an interesting link between classical and quantum indeterminism opens up.
This is made much more dramatic by the fact that such an interpretation has already been given for quantum mechanics by Wheeler's student Everett, but without use of any formal apparatus such as modal logic. Everett supposes that the wave function does not 'collapse' to a single eigenstate but instead retains all eigenstates as genuine physical descriptions, which however belong to different histories 'splitting' away from one another at the time of measurement. This rather incredible 'theory' in fact was shown by Everett to be implicit in the very formalism of quantum mechanics, i.e., it is 'there' in the mathematical information, although the degree of 'reality' one chooses to endow the histories with becomes a matter of personal prejudice. Certainly it isn't something one can accept very readily, however squarely it falls within the earlier demands of Peirce and the modalist view of physics.

In any case, we needn't worry over issues of palatability, as long as there are interesting technical questions involved, and it appears that this is the case regarding a modal interpretation of indeterminism in mechanics (classical and quantum). In fact, d'Espagnat's surprising evidence for non-local effects seems to disappear in the many-worlds view. Since Everett's scheme appears consistent with d'Espagnat's conclusions (due to the latter's finding that quantum mechanics itself denies the prior existence of certain properties before measurement), this should simultaneously shed light on the extent to which classical objects can be said to 'have' properties defined by counterfactual (pre-interaction) hypotheses. In any event, Everett's approach hasn't been developed with sufficient formality at present to make precise this relationship to d'Espagnat.

In (Prior 1971) a treatment of belief is given that views propositions and facts as distinct from any physical representation. We look at this briefly now, for it bears directly on properties. 'The Sun is hot' expresses a proposition that was true before any sentences (in English) existed. But of what does the proposition consist? Indeed the Sun was hot, and long before we were around to talk about it, but this is to say that a certain physical situation obtained. We might say the proposition that the Sun is hot isolates an aspect of a physical situation, so that there is not a strict identity of proposition and situation. But the Sun's being hot as an isolated phenomenon isn't very cogent except as something singled out for expression in some language by an entity that can focus attention on part of a situation. In any case, the truth ascribable to such a proposition derives entirely from the reality of the Sun's heat, i.e., in material events in the Sun (motion of hydrogen atoms, etc.). It would seem there is no further thing to 'locate,' no proposition beyond these material conditions out of which the former supposedly...
May we then continue to regard beliefs (and propositions) as part of physical reality? If we call someone's attention to the 'fact' that the Sun is hot today, are we doing more than calling attention to the heat of the Sun, i.e., to the Sun, the heat, and the apparent coming of the latter from the former? If not, then this is equally well expressed by saying simply 'The Sun is hot.' On the other hand, if we say that it is a fact that the Sun is hot today, we also direct attention more explicitly to language, connecting a perceived situation with a specific representation of it ('that...'). The 'fact' then is either the reality, or an expression of it recognized both as symbolism and symbolic of something specific.

It can be argued that false propositions do not correspond to reality and yet are propositions just the same, believable and ponderable. But they are linguistic entities, they have meaning only insofar as we interpret them via our expectations and actions. An internal representation can be wrong and still be nothing more than a representation.

Have we then dispensed with propositions and facts? Are we able to get along just with physical reality? It is surely tempting to think so, especially since the additional concepts we conveniently refer to so often arise precisely from that reality. But although conceptual entities such as propositions, which are to be true or false and so presumably are sensible only to a user of language, may be constructs as we have indicated above, still the matter cannot rest there. For we also speak of hotness, a property, and this doesn't seem quite so evident as a physical entity. That is, although heat is assuredly a physical phenomenon, it is a general one, rather than any specific occurrence of heat.

Now is heat too just a construct? It is hard to see how such could arise without there already being something to relate the various instances we wish to regard as leading to the concept of heat. Indeed were it not for the regularities attendant to these instances, such as our experience of warmth, then no such generalization would seem possible. So it seems we must accept that features of similarity between physical events are part of the 'situation' of things, even though the manner by which those similarities obtain isn't part of what we customarily view as the physical world. To be sure, heat can be regarded (correctly) as a conjunction of attributes of motion of molecules. But then we have just substituted motion for heat, and the above discussion seems to apply with the same force.
Here we close our brief foray into the philosophy of science. Although we have made no real headway on the issue of factual possibility or the associated issue of properties, we hope to have illustrated that the problems seem real indeed.

APPENDIX C
IMPLEMENTATION

The program that simulates Sally's STM stages was written in QLOG, which is a PROLOG-like system built on top of LISP. That is, QLOG consists of certain LISP functions which act on special lists called assertions and determine whether a given query is derivable from the assertions. The long-term memory of Sally given in the text of Chapter IV consists essentially of QLOG assertions, although in a very limited form. The more general form used below is as follows: ((P ?x ...) (Q ?x ...) (R ?x ...) ...) with the meaning that P(x...) follows from Q(x...) & R(x...) & ... in the more familiar notation of first-order logic.

The 22 assertions below contain the facts determining the temporary queue referred to as 'Q' in Chapter IV and as 'qtm' here. This is the set of new beliefs to go into STM at any given time. QLOG queries of the form (qtm ?x) return all such new beliefs, and some LISP code places these in STM by making new assertions of the form (stm ?x) for each such ?x returned. The predicate 'ttm' plays the role of...
temporary storage for beliefs that must be kept out of qtm, namely, those already in stm and those goals already achieved in stm. The various 'f'-predicates such as ftm, record when the corresponding predicate fails to hold, e.g., if a query (ttm ?x) returns nothing. Numbers 2, 3, and 4 appended to various predicates, serve to mark the number of arguments used, to facilitate searching by QLOG. Finally, the predicate 'oldgoal' refers to goals in ITM that haven't been achieved.

2. ((qtm ?x) (stm (result ?y ?x)) (stm ?y))
3. ((qtm ?x) (stm (goal ?x)) (itm ?x))
4. ((qtm (stm-search ?x)) (stm (expect ?x)))
5. ((qtm (stm-search ?x)) (stm (check ?x)))
6. ((qtm (not (stm ?x))) (stm (stm-search ?x)) (fstm ?x))
7. ((qtm (stm ?x)) (stm (stm-search ?x)) (stm ?x))
8. ((qtm ?x) (stm (imply2 ?y ?x)) (stm ?y))
14. ((fttm ?x) (ttm ?x) (cut) (fail))
15. ((fttm ?x))
16. ((fstm ?x) (stm ?x) (cut) (fail))
17. ((fstm ?x))
18. ((fitm ?x) (itm ?x) (cut) (fail))
19. ((fitm ?x))
20. ((ttm (goal ?x)) (stm ?x))
21. ((ttm ?x) (stm ?x))
22. ((oldgoal ?x) (itm (goal ?x)) (fstm ?x) (fitm ?x))

BIBLIOGRAPHY


31. Hayes, P. 'In defense of logic,' IJCAI-77.


63. Quine, W., 'Concatenation as a Basis for Arithmetic,' *J Symb Log.* 11, 1946.


65. Reiter, R., 'A logic for default reasoning,' *Artificial Intelligence,* 13, 1 and 2, 1980.


80. Winograd, T., 'Extended inference modes in reasoning by computer systems,' Artificial Intelligence, 13, 1 and 2, 1980.