READINGS IN KNOWLEDGE REPRESENTATION

SCHANK & RIEGER
1974

edited by

Ronald J. Brachman
Schlumberger Palo Alto Research
Currently at
AT&T Bell Laboratories

and

Hector J. Levesque
University of Toronto
and
The Canadian Institute for Advanced Research

MORGAN KAUFMANN PUBLISHERS, INC.
SAN MATEO, CALIFORNIA

7 / Roger C. Schank and Charles J. Rieger III
Inference and the Computer Understanding of Natural Language

This paper concerns itself with the crucial problem of specifying language-free inferences in a natural language understanding context. While Schank's earlier work on Conceptual Dependency—a representation based on structures built out of a small number of primitive actions—was clearly influential, it gains some of its most important force from the work Rieger did on determining how inferences can be generated from it. As detailed in this article, inferences can be spontaneously generated from a Conceptual Dependency structure by considering only easily specified patterns of primitive relations, thus avoiding any language-dependency. Schank and Rieger illustrate this point with a set of twelve kinds of inferences that help add to a parsed sentence information that is likely to be correct. They also spend considerable time discussing the implementation of their ideas in the MARGIE program. Among the more interesting aspects of this work is the length to which they go to position themselves opposite work on formal deduction (*á la* [Moore, Chapter 18], say). The work has perhaps more of a formal flavor to it than the authors would like to admit, but their protest against deduction is well taken and is similar to others' (*e.g.*, see [Minsky, Chapter 12] and [Garvey *et al.*, Chapter 27]). On the other hand, their problem probably has more to do with the need for default reasoning [Reiter, Chapter 23] and the inexactness of normal thought processes than with the narrow view of deduction taken here.

Appeared in *Artificial Intelligence* 5(4), 1974, 373–412.

Inference and the Computer Understanding of Natural Language[1]

Roger C. Schank[2] and Charles J. Rieger III[3]

Computer Science Department, Stanford University, Stanford, Calif. 94305, U.S.A.

Recommended by E. Sandewall

ABSTRACT

The notion of computer understanding of natural language is examined relative to inference mechanisms designed to function in a language-free deep conceptual base (Conceptual Dependency). The conceptual analysis of a natural language sentence into this conceptual base, and the nature of the memory which stores and operates upon these conceptual structures are described from both theoretical and practical standpoints. The various types of inferences which can be made during and after the conceptual analysis of a sentence are defined, and a functioning program which performs these inference tasks is described. Actual computer output is included.

CONTENTS

1. Introduction

The question of what belongs to the domain of parsing and what is part of the domain of inference inevitably comes up when attempting to put together a system in order to do natural language understanding. This paper is intended to explain the difference within the context of Conceptual Dependency Theory [6, 7, 8], categorize the kinds of inferences that are necessary within such an understanding system, and outline the basic elements and processes that make up the program at Stanford that currently handles these inference tasks.

We shall assume in this paper that it is the desire of those researchers who work on the problems of computational linguistics to have a system that is capable of responding intelligently, on the basis of its own model of the world, in reaction to a given input sentence. Thus, we assume here that a system responds as follows (for example) is both an interesting and useful system if it accomplishes these things:

(1) INPUT: I am going to buy some aspirin for my cold.
OUTPUT: Why don't you try some chicken soup instead?
(2) INPUT: John asked Mary for a book.
OUTPUT: A book about what?
(3) INPUT: Do you want a piece of chocolate.
OUTPUT: No, I don't want to spoil my appetite for dinner.
(4) INPUT: John went to the store.
OUTPUT: What did he want to buy?

Before getting into the descriptions of the various kinds of inferences to which a conceptual memory should be sensitive, the notion of inference and how it differs from logical deductions (for instance in a theorem-prover or question-answerer) should be made clear.

In its broadest sense, we consider an inference to be a new piece of information which is generated from other pieces of information, and which may or may not be true. The intent of inference-making is to "fill out" a situation which is alluded to by an utterance (or story line) in hopes of tying pieces of information together to determine such things as feasibility, causality and intent of the utterance at that point. There are several features of all inferences which should make clear how an inference differs in substance and intent from a formal deduction:

(1) Inference generation is a "reflex response" in a conceptual memory. That is, one of the definitions of "processing conceptual input" is the generation of inferences from it. This means that there is always an implicit motivation to generate new information from old. In a theorem-prover or question-answerer, deductions are performed only upon demand from some external process.

(2) An inference is not necessarily a logically valid deduction. This means that the new information represented by the inference might not bear any formal logical relationship to those pieces of information from which it is generated. A good example of this is called "affirmation of the consequent", a technique fruitfully utilized by Sherlock Holmes, and certainly utilized by people in everyday situations. Briefly, this refers to the "syllogism" $A \supset B, B;$

[1] This research was supported by the Advanced Research Projects Agency of the Department of Defense under Contract SD-183.

The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Advanced Research Projects Agency or the U.S. Government.

[2] Present address: Computer Science Department, Yale University, New Haven, Connecticut 06520, U.S.A.

[3] Present address: Computer Science Department, University of Maryland, College Park, Maryland 20740, U.S.A.

therefore *A*. In this sense (and there are other examples), conceptual memory is strikingly different from a formal deductive system.

(3) An obvious consequence of (2) is that an inference is not necessarily true. For this reason, it is useful for memory to retain and propagate measures of the degree to which a piece of information is likely to be true. Memory must also be designed with the idea that *no* information is inviolably true, but rather must always be willing and able to respond to contradictions.

(4) The motivations for inference generation and formal deduction are entirely different. Formal deductions are highly directed in the sense that a well-defined goal has been established, and a path from some starting conditions (axioms and theorems) to this goal is desired. Inferences on the other hand are not nearly so directed. Inferences are generally made to "see what they can see". The "goal" of inferencing is rather amorphous: make an inference, then test to see whether it looks similar to, is identical to, or contradicts some other piece of information in the system. When one of these situations occurs, memory takes special action in the form of discontinuing a line of inferencing, asking a question, revising old information, creating causal relationships or invoking a belief pattern.

(5) A memory which uses the types of inference we will describe needs some means of recourse for altering the credibility of a piece of information when the credibility of some piece of information which was used in its generation changes. In other words, memory needs to remember *why* a piece of information exists. In contrast, a formal deductive system in general doesn't "care" (or need to know) where a fact came from, only that it exists and is true.

Having made these distinctions between conceptual inference and other types of logical deductions, we will describe some distinct types of inference.

2. Inference and Parsing

We take as one of our operating assumptions, that the desired output for a conceptual analyzer is a meaning representation. Since it is possible to go directly from an input sentence into a meaning representation (see [4, 6, 11] for descriptions of computer programs that do this), we shall disregard any discussion of syntactic parsing output.

What then should be present in a meaning representation? We claim that it is necessary for a meaning representation to contain each and every concept and conceptual relation that is explicitly or implicitly referred to by the sentence being considered.

By explicit reference we mean the concepts that underlie a given word. Thus we have the concept of John for "John" and the concept of a book for "book" in sentence (5):

(5) John bought a book.

However, we claim in addition that an adequate meaning representation must make explicit what is implicit but nonetheless definitely referenced in a given sentence. Thus, in (5) we have the word "bought" which implicitly references two actions of transfer, one whose object is the book and the other whose object is some valuable entity. We assume that hearers of (5), unless specifically told otherwise, will assume that this object is "money".

It is here then that we shall make our first distinction between the province of parsing (or the extraction of explicit and implicit information) and that of inference (the adding-on of probably correct information). The word "buy" has a number of senses in English, but the surrounding information disambiguates "buy" so that in (5) it can only mean that two actions of transfer occurred and that each action caused the other's existence. Furthermore, it is always true that whenever one of these transfer actions is present (hence called ATRANS for abstract transfer) it is also true that an actor did the ATRANSing; there was an object acted upon, and there was a recipient and a donor of this object.

We now state our first inference type which we call LINGUISTIC-INFERENCE:

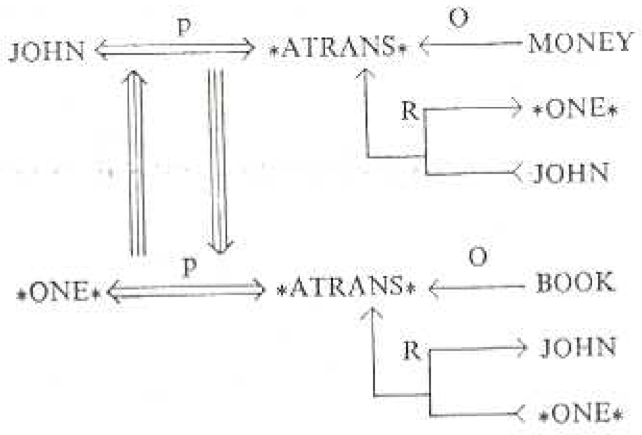
1. An instance of LINGUISTIC-INFERENCE exists when, in the absence of specific information to the contrary, a given word or syntactic construction can be taken to mean that a specific but unmentioned object is present in a predicted case for a given ACT with a likelihood of near certainty.

In the above example, the ACT is ATRANS, its predicted cases are OBJECT, RECIPIENT (includes receiver and donor) and INSTRUMENT. The word "buy" by definition refers to the ACT ATRANS and therefore implicitly references its cases. However, in addition "buy" has as a linguistic inference the object "money" as the object of the ATRANS whose actor is the subject of the sentence in which "buy" appears.

We assign to the conceptual analyzer the problem of handling explicit reference, implicit reference, and linguistic inference within a meaning representation because these are consequences of words. Using Conceptual Dependency notation (where $\Leftrightarrow$ denotes the relation between actor and action; $\leftarrow$O denotes the relation between action and object; $\Leftarrow$ denotes causality dependence; and

R

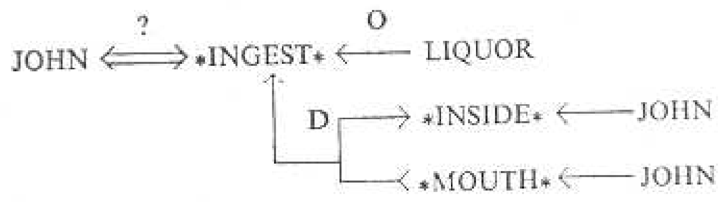
denotes the relation between action, object, recipient and donor), the conceptual analyzer (described in [4]) outputs the following for (5):



Two more common examples of linguistic inference can be seen with reference to sentences (6) and (7):

(6) Does John drink?
(7) John hit Mary.

In (6), it is reasonable to assume that the referenced object is "alcoholic beverages" although it is unstated. It is a property of the word "drink" that when it appears without a sentential object "alcoholic beverage" is understood. (In fact, this a property of quite a few languages, but from this it should not be thought that this is a property of the concept underlying "drink". Rather it is an artifact of the languages that most of them share common cultural associations.) Thus, given that this is a linguistic inference, and that our conceptual analyzer is responsible for making linguistic inferences, our analyzer puts out the following conceptual structure for it:



The ACT INGEST is used here. We shall explain the notion of a primitive ACT in the next section.

In (7), we again have the problem that what hearers usually assume to be the meaning of this sentence is in fact quite beyond what the sentence explicitly says. Sentence (7) does not explicitly state what John did. Rather we must call upon some other information to decide if John threw something at Mary or if he swung his hand at her (and whether his hand was holding some object). Notice that the same ambiguity exists if we had sentence (8), but that one meaning is preferred over the other in (9):

(8) John hit Mary with a stick.
(9) John hit Mary with a slingshot.

We shall claim that for (7) when no other information is explicit, the most likely reading is identical with the reading for (10):

(10) John hit Mary with his hand.

Thus, (7) is another example of linguistic inference and it is the responsibility of the conceptual analyzer to assume "hand" as the thing that hit Mary on the basis of having seen "hit" occurring with no syntactic instrument. (Note that syntactic instrument is quite different from the conceptual INSTRUMENTAL case mentioned earlier.) Before we get into inferences that are not linguistic it will be necessary to explain further the elements of the meaning representation that we use as the input to our inference making procedures.

We would like to point out at this point that we assign the problem of extracting conceptual structures and making linguistic inferences to the domain of the conceptual analyzer. This is because the information that is used for making the decisions involved in those processes is contained in the particular language under analysis. From this point on in this paper we shall be discussing inferences that come from world knowledge rather than from a particular language. It is those interlingual processes that we assign to the domain of a memory and inference program such as we shall describe in Section 6.

3. The Twelve Primitive Actions

Conceptual Dependency theory is intended to be an interlingual meaning representation. Because it is intended to be language free, it is necessary in our representations to break down sentences into the elements that make them up. In order to do this it is necessary to establish a syntax of possible conceptual relationships and a set of conceptual categories that these relate. Furthermore it is necessary that requirements be established for how a given word is mapped into a conceptual construction.

There are six conceptual categories in Conceptual Dependency:

PP Real world objects,
ACT Real world actions,
PA Attributes of objects,

AA Attributes of actions.
T Times.
LOC Locations.

These categories can relate in certain specified ways which are considered to be the syntactic rules of conceptualizations. There are sixteen of these conceptual syntax rules, but we shall list here only the ones that will be used in this paper:

PP ⇔ ACT	indicates that an actor acts.
PP ⇔ PA	indicates that an object has a certain attribute.
$ACT \xleftarrow{O} PP$	indicates the object of an action.
ACT ← [R: → PP, ← PP]	indicates the recipient and the donor of an object within an action.
ACT ← [D: → PP, ← PP]	indicates the direction of an object within an action.
$ACT \xleftarrow{I} \Updownarrow$	indicates the instrumental conceptualization for an action.
X ⇑ Y	indicates that conceptualization X caused conceptualization Y. When written with a "c" this form denotes that X COULD cause Y.
PP ⇐ [→ PA2, ← PA1]	indicates a state change of an object.
PP1 ← PP2	indicates that PP2 is either PART OF or the POSSESSOR OF PP1.

In Conceptual Dependency, tenses are considered to be modifications of the main link between actor and action (⇔), or the link between an object and its state (⇔). The main link modifiers we shall use here are:

p	past.
f	future.
(null)	present.
ts = x	begin a relation at time x.
tf = x	end a relation at time x.
c	conditional.
/	negation.
?	question.

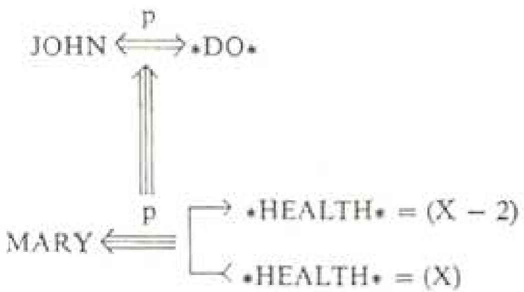
The most important category for our purposes here is the ACT. A word maps into an ACT when it specifically refers to a given possible action in the world. Often verbs only reference unstated actions and make specific reference to states or relationships between these unspecified actions. As an example we have sentence (11):

(11) John hurt Mary.

Here, the real world action that John did is unstated. Only the effect of this action is known: namely that it caused Mary to enter a "hurt" state. Similarly, in (12) the word "prevent" is not a specific real world action but rather refers to the fact that some unstated action caused that some other action (that may or may not be specified later on in the sentence) did not occur.

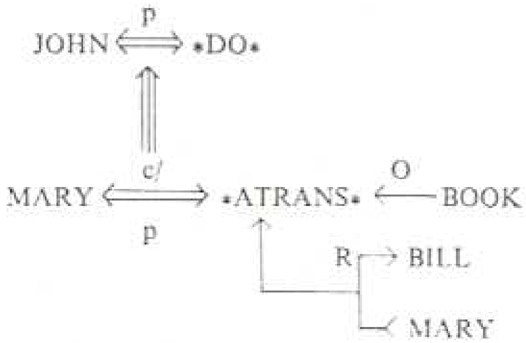
(12) John prevented Mary from giving a book to Bill.

The analyses of these sentences (11) and (12) are as follows:



(indicates HEALTH has gone down 2 points on a 10 point scale)

and



Since many verbs are decomposed into constructions that involve only unstated actions (denoted by DO) and/or attributes of objects (PA's) and since we require that any two sentences that have the same meaning be represented in one and only one way, the set of primitive ACTs that are used is important.

We have found that a set of only twelve primitive actions is necessary to account for the action part of a large class of natural language sentences. This does not mean that these primitives are merely category names for types of actions. Rather, any given verb is mapped into a conceptual construction that may use one or more of the primitive ACTs in certain specified relationships plus other object and state information. That is, it is very important that no information be lost with the use of these primitives. It is the task of the primitives to conjoin similar information so that inference rules need not be written for every individual surface verb, but rather inference rules can be written for the ACTs. This of course turns out to be extremely economical from the point of view of memory functioning.

The twelve ACTs are:

ATRANS	The transfer of an abstract relationship such as possession, ownership, or control.
PTRANS	The transfer of physical location of an object.
PROPEL	The application of a physical force to an object.
MOVE	The movement of a bodypart of an animal.
GRASP	The grasping of an object by an actor.
INGEST	The taking in of an object by an animal.
EXPEL	The expulsion from the body of an animal into the world.
MTRANS	The transfer of mental information between animals or within an animal. We partition memory into CP (conscious processor), LTM (long-term memory), and sense organs. MTRANSing takes place between these mental locations.
CONC	The conceptualizing or thinking about an idea by an animal.
MBUILD	The construction by an animal of new information from old information.
ATTEND	The action of directing a sense organ towards an object.
SPEAK	The action of producing sounds from the mouth.

The following important rules are used within Conceptual Dependency:

(1) There are four conceptual cases: OBJECTIVE, RECIPIENT, DIRECTIVE, INSTRUMENTAL.

(2) Each ACT takes from two to three of these cases obligatorally and none optionally.

(3) INSTRUMENTAL case is itself a complete conceptualization involving an ACT and its cases.

(4) Only animate objects may serve as actors except for PROPEL.

We are now ready to return to the problem of inference.

4. Language-Free Inferences

The next class of inference we shall discuss includes those that come from objects and relate to the normal function of those objects. As examples we have sentences (13) and (14):

(13) John told Mary that he wants a book.

(14) John likes chocolate.

These sentences have in common that they refer to an action without specifically stating it. In these examples, this missing act concerns the probable use of some object. In (13) that ACT is probably MTRANS (i.e., people usually want books because they want to MTRANS information from them) and in (14) that ACT is probably INGEST (i.e., people normally "like" chocolate because they like to INGEST it). While it is certainly possible that these were not the intended ACTs (John could like burning books and painting with chocolate) it is highly likely that without contrary information most speakers will assume that those ACTs were referenced. In fact, psychological tests have shown (see [5], [10], for example) that in many cases most hearers will not actually remember whether the ACTs were specifically mentioned or not. Notice in the first example that the missing MTRANS (of information from the book) is an inference which occurs AFTER the meaning representation of the sentence has been established (i.e., this sentence is analyzed as "if someone were to ATRANS a book to me it would cause me pleasure"). On the other hand, the missing INGEST in the second example is inferred during the analysis because the REPRESENTATION itself depends upon the analyzer knowing what it means to "like" a food. Therefore, the determination of an object's probable relation to an actor is never strictly a part of just the analyzer or just the memory, but rather a task of conceptual analysis in general.

It is important to mention that, regardless of the ultimate correctness of the chosen ACT, Conceptual Dependency predicts that an ACT is missing because verbs like "want" and "like" are represented as states. In the parsing of each of these sentences it is found that an actor and an object are present with no ACT to link them. This causes a search to be made for the correct ACT to fill that spot.

We thus have our second and third inference-types:

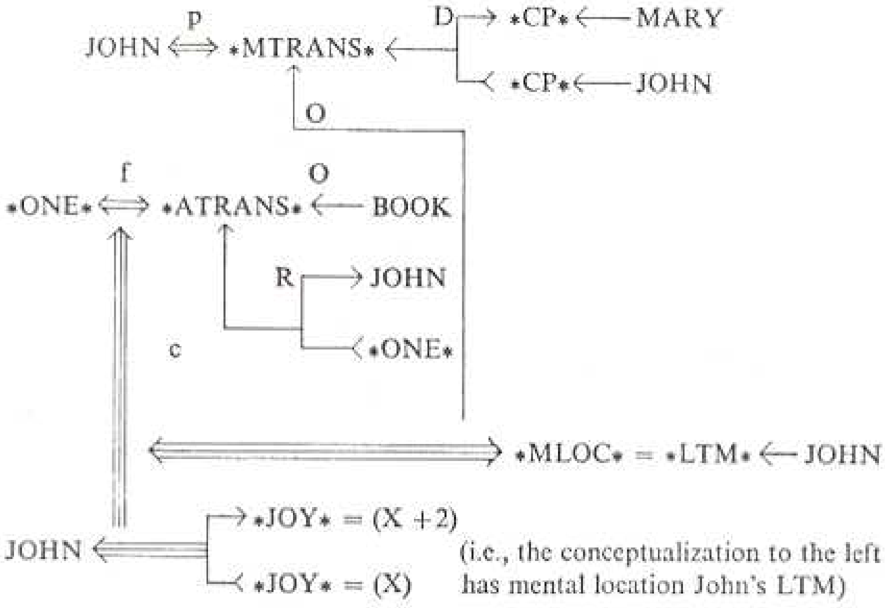
2. An instance of ACT-INFERENCE is present when an actor and an object occur in a conceptualization without an ACT to connect them, and the object in question has a normal function in the world. In this case the normal function is assumed to be the implicitly referenced ACT.

and

3. A TRANS-ENABLE-INFERENCE *occurs with conceptualizations*

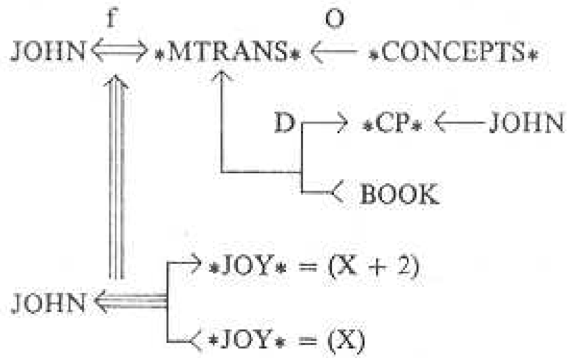
involving one of the TRANS ACTs. It is inferred that the TRANS conceptualization enables another conceptualization involving the same actor and object to take place. The specific act for this inferred conceptualization then comes about via ACT-INFERENCE. Inferences of this type are frequently useful for inferring the intended use of a physical or mental object.

The finished analyses for (13) and (14) after ACT-INFERENCE and TRANS-ENABLE-INFERENCE take place are then:

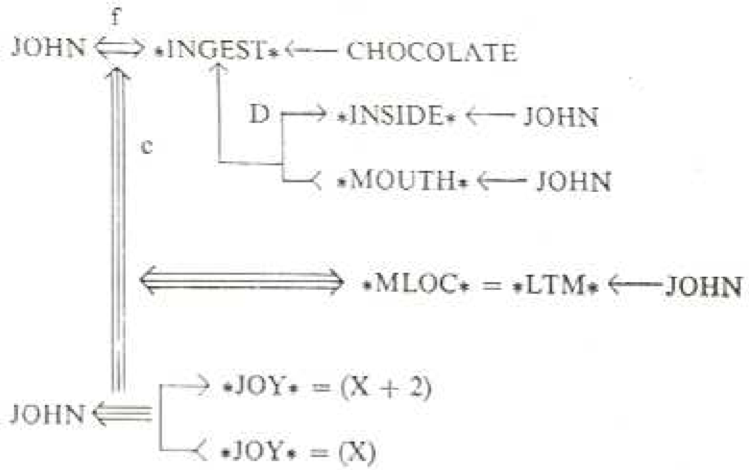


(i.e., the conceptualization to the left has mental location John's LTM)

(which eventually leads to:



i.e., John wants to read the book), and



The next kind of inference that we shall discuss has to do with the results of a given ACT. Consider sentences (15), (16) and (17):

(15) John went to South Dakota.
(16) John told Mary that Bill was a doctor.
(17) John gave Mary a book.

Each of these sentences refers to an ACT that has a predictable result. Here again, when no information is given that contradicts this prediction, it is reasonable to assume that the normal result of the action was achieved. (Here, as in most of the examples given in this paper, it is necessary in English to use the conjunction "but" to indicate that the inferred result did not take place. Thus, unless we add "but he didn't get there' to (15), the hearer will assume he did.)

We thus have our fourth example of inference:

4. RESULT-INFERENCE can be made whenever a TRANS ACT is present and no information exists that would contradict the inferred result.

Thus, whenever PTRANS is present, we can infer that the location of the object is now the directive case of PTRANS. Whenever ATRANS is present we can infer that there is a new possessor of the object, namely the recipient, and lastly, whenever an MTRANS occurs we can assume that the information that was transferred to the conscious processor (CP) of the brain became present there. Thus for (16), Mary can be assumed to "know" the information that was told to her since "know" is represented as "exist in the long term memory (LTM)" and "tell" involves MTRANSing to the conscious processor which leads to LTM. A program that deals with this problem will be discussed later on in this paper.

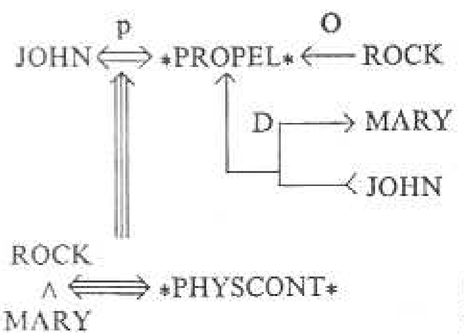
The fifth kind of inference that we shall discuss is called OBJECT-AFFECT-INFERENCE. This kind of inference also concerns the result of an ACT but here we mean result to refer to some new physical state of the object involved. Sentences (18) and (19) illustrate this problem:

(18) John hit Mary with a rock.
(19) John ate the egg.

Both (18) and (19) make an implicit statement about a new physical state of the item that is in the objective case. In (18) we can guess that Mary's state of physical health might have been diminished by this ACT (i.e., she was hurt). In (19) we know that the egg, no matter what state it was in before this ACT, is now in a state of not existing at all anymore. Thus we have inference-type 5:

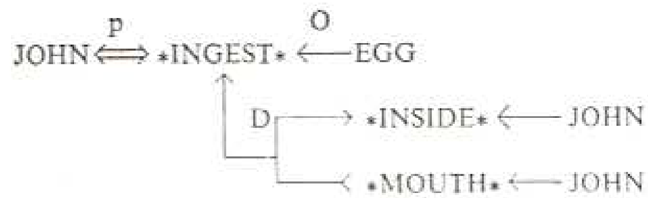
> 5. An instance of OBJECT-AFFECT-INFERENCE may be present with any of the physical ACTs (INGEST, EXPEL, PROPEL, GRASP, MOVE). The certainty of any of these inferences is dependent on the particular ACT, i.e., INGEST almost always affects the object, PROPEL usually does and the effects of the others are less frequent but possible. When OBJECT-AFFECT-INFERENCE is present, a new resultant physical state is understood as having been caused by the given ACT.

The analyses for (18) and (19) are given below. Note that if "rock" is replaced by "feather" in (18) the inference under discussion is invalid. Thus, in order to accomplish this inference correctly on a machine, the specifications for under what conditions it is valid for a given ACT must be given. Obviously these specifications involve mass and acceleration as well as fragility in the case of PROPEL.



(*PHYSCONT* represents the state of physical contact; many such states are used)

and



The next kind of inference we shall discuss concerns the reasons for a given action. Until now, we have only considered the effects of an action or the unstated pieces of a given conceptualization. However, in order to conduct an intelligent conversation it is often necessary to infer the reason behind a given event. Consider sentences (20), (21) and (22):

(20) John hit Mary.
(21) John took an aspirin.
(22) John flattered Mary.

We would like a computer to have the ability to respond to these sentences as follows:

(20a) What did Mary do to make John angry?
(21a) What was wrong with John?
(22a) What does John want Mary to do for him?

In order to accomplish this, we need to use some of the inference-types discussed above first. Thus, in (20), we must first establish that Mary might be hurt before we can invoke an appropriate belief pattern. By belief pattern we mean a sequence of causally-related ACTs and states that are shared by many speakers within a culture. Such a sequence usually deals with what is appropriate or expected behavior and is often a prescription for action on the part of the hearer.

The belief pattern called by (20) is commonly described as VENGEANCE. It states that people do things to hurt people because they feel that they have been hurt by that person. This belief pattern supplies a reason for the action by the actor. Thus we come to the sixth kind of inference:

> (6) An instance of BELIEF-PATTERN-INFERENCE exists if the given event plus its inferred results fit a belief pattern that has in it the reason for that kind of action under ordinary circumstances.

In example (21) we have an instance of the WANT belief pattern which refers to the fact that people seek to obtain objects for what they can use them for (this is intimately related to inference-type 2 discussed above). Sentence (22) refers to the RECIPROCITY belief pattern (which deals with "good" things (i.e., those that cause positive changes on the JOY scale), VENGEANCE taking care of the "bad" ones). RECIPROCITY comes in two types. The one being used here is anticipatory. That is, the action is being done with the hope that the nice results achieved for one person will encourage that person to do something which will yield nice results for the original actor.

We will further discuss (20) later on in this paper when we outline the procedure by which our computer program produces (20a) in response to it.

The next kind of inference we shall discuss is called INSTRUMENTAL-INFERENCE. It is the nature of the primitive ACTs discussed earlier that they can take only a small set of ACTs as instrument. Thus, for example, whenever INGEST occurs, PTRANS must be its instrumental ACT because by definition PTRANS is the only possible instrument for INGEST. The reason for this is that in order for someone to eat something it is necessary to move it to him or him to it. Thus, whenever INGEST is present we can make the legitimate inference that the object of INGEST was PTRANSed to the mouth (nose, etc.) of the actor. If this inference is incorrect, it is only because the direction of motion was mouth to object instead. Also, whenever PTRANS appears, the instrument must have been either MOVE or PROPEL. That is, in order to change the location of something it is necessary to move a bodypart or else apply a force to that object (which in turn requires moving a bodypart). Thus we have the seventh inference type:

7. INSTRUMENTAL-INFERENCE can always be made, although the degree of accuracy differs depending on the particular ACT. Whenever an ACT has been referenced, its probable instrument can be inferred.

The list of instrumental ACTs for the primitive ACTs follows:

INGEST	instrument is PTRANS.
PROPEL	instrument is MOVE or GRASP (ending) or PROPEL.
PTRANS	instrument is MOVE or PROPEL.
ATRANS	instrument is PTRANS or MTRANS or MOVE.
CONC	instrument is MTRANS.
MTRANS	instrument is MBUILD or SPEAK or ATTEND or MOVE.
MBUILD	instrument is MTRANS.
EXPEL	instrument is MOVE or PROPEL.
GRASP	instrument is MOVE.
SPEAK	instrument is MOVE.
ATTEND	no instrument is needed, although MOVE often applies.

Using this table it is possible, for example, to make the following inferences from these sentences:

(23) John is aware that Fred hit Mary.
(24) John received the ball.

Since (23) refers to CONC and CONC requires MTRANS as instrument, we can infer (from the possible instruments of MTRANS) where John got his information. He could have MBUILDed it (not likely here because Fred hit Mary is an external event); he could have perceived it from his senses by ATTEND eye to it himself; or by ATTEND ear to someone else which MTRANSed it to him. Since (24) refers to PTRANS, we have two possible instruments MOVE or PROPEL. From this we can infer that the ball was handed to him (move someone else's bodypart) or else it was rolled or thrown (or underwent some other manner of applying a force to a ball).

The next type of inference is PROPERTY-INFERENCE:

8. Whenever an object is introduced in a sentence, certain subpropositions are being made. The most common instance of this is the predication that the object being referenced exists. The inference of these subpropositions we call PROPERTY-INFERENCE.

In some instances, PROPERTY-INFERENCE is dependent on other inference types. Thus, in the sentence "John hit Mary", not only is it necessary to make the PROPERTY-INFERENCE that both John and Mary exist, but it is also necessary to realize that John must have arms in order to do this. This inference is thus dependent on the LINGUISTIC-INFERENCE that, unless otherwise specified, "hit" refers to "hands" as the object of the PROPELing.

PROPERTY-INFERENCE is necessary in a computer understanding system in order to enable us to respond either with surprise or a question as to manner if we know that John does not have arms. Furthermore, in answering questions, it often happens that the checking of subpropositions associated with PROPERTY-INFERENCE will allow us to find an answer with less work. Thus for sentence (25):

(25) Did Nixon run for President in 1863?

Two separate subpropositions that can be proved false allow the question to be answered most efficiently. Establishing that "Nixon was alive in 1863" is false or that "there was a presidential election in 1863" is false is probably the best way of answering the question.

We have not discussed to this point the standard notions of logical inference for two reasons: (a) the problems involving logical inference are already fairly well understood, and (b) we do not view logical inference as playing a CENTRAL role in the problem of computer understanding of natural language. However, there exists a related problem that bears discussion.

Consider the problem of two sentences that occur in sequence. Often such sentences have additional inferences together which they would not have separately. For example, consider:

(26a) All redheads are obnoxious.
(26b) Queen Elizabeth I had red hair.
(27a) John wants to join the army.
(27b) John is a pacifist.

In (26), (26b) has its obvious surface meaning, but also can mean either one of two additional things. Either we have the inference that Queen Elizabeth I was obnoxious according to the speaker, or if (26b) were spoken by a different speaker from (26a), there exists the possibility that (26b) is intended as a refutation of (26a).

For (27), a sophisticated language analyzer must discover that (b) is essentially a contradiction of (a) and hence the inference that the speaker of (b) believes that the speaker of (a) is in error is probably correct. We thus introduce inference-type 9:

9. An instance of SEQUENTIAL-INFERENCE is potentially present when one sentence follows another and they share a subject or a proposition. When subpropositions or inferences of subpropositions can be detected as common to both conceptualizations, and satisfy certain set inclusion or contradiction rules, SEQUENTIAL-INFERENCE may apply.

The next kind of inference is quite straightforward:

10. An instance of CAUSALITY-INFERENCE is present if two sentences are connected by an "and" or by their appearing in sequence. Then if one could have caused the other, it can be inferred that this is what happened.

Consider sentences (28) and (29a) and (29b):

(28) John hit Mary and she died.
(29a) John hit Mary.
(29b) John died.

In these sentences it is usually correct to assume causality. For (28) we infer that the hitting caused Mary's death. For (29) we infer that (a) caused (b). It is our knowledge of the world, however, that would cause us to wonder about the connection in (29) but not in (28). A good program would discover this to be a different kind of causality from the straight result present in (28). Kinds of causality are discussed in [7].

Another important inference type BACKWARD-INFERENCE. This type of inference can be made whenever an action has occurred that required another action to precede. The possible actions that can be inferred for a given ACT as BACKWARD-INFERENCE are often quite similar to those which can be inferred as instruments for a given ACT. We use this kind of inference whenever an object is acted upon. Thus if we have:

(30) John ate a banana.

we can infer that the banana must have been PTRANSed to him at some time. Likewise, whenever a mental item is operated upon its previous MTRANSing can also be inferred. If we have:

(31) John knows where Mary is.

then we can infer that this information must have been MTRANSed to John at some point (either from his eyes or from someone else MTRANSing this information to him). Thus we have inference type 11:

11. All conceptualizations are potentially subject to BACKWARD-INFERENCE. Depending on the nature of the object, one of the TRANS ACTs can be inferred as having enabled the current conceptualization's occurrence.

The last kind of inference we shall discuss concerns the intention of the actor. Consider the following sentences.

(32) John hit Mary.
(33) John told Bill that he wants to go to New York.

We assume that a person does something because he wants to do it and that he wants to do it because of the results that he expects to achieve. Thus a valid inference here is that it is the intention of the actor that the things inferred with OBJECT-AFFECT-INFERENCE or RESULT-INFERENCE will occur, and that these things are desired by the actor.

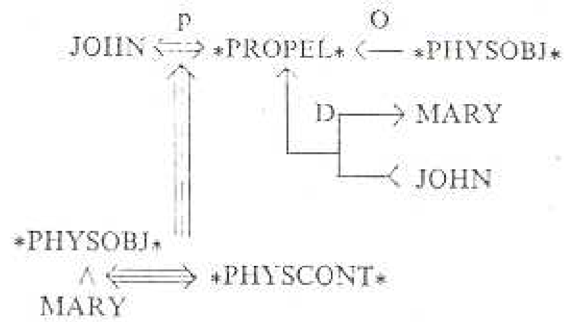
Thus from (32) using inference-type 6 we get that "Mary is hurt pleases John". From (33), using inference-type 5, we get that "being located in New York will please John" and "Bill knowing this pleases John". Thus we have inference-type 12:

12. INTENTION-INFERENCE is assumed whenever an actor acts unless information to the contrary exists.

5. Observations

Using the inference types discussed above we can see that an effective analysis of a sentence is often quite a bit more than one might superficially imagine. If we start with the sentence "John hit Mary" for example, our conceptual analyzer would perform the following conceptual analysis:

(34)

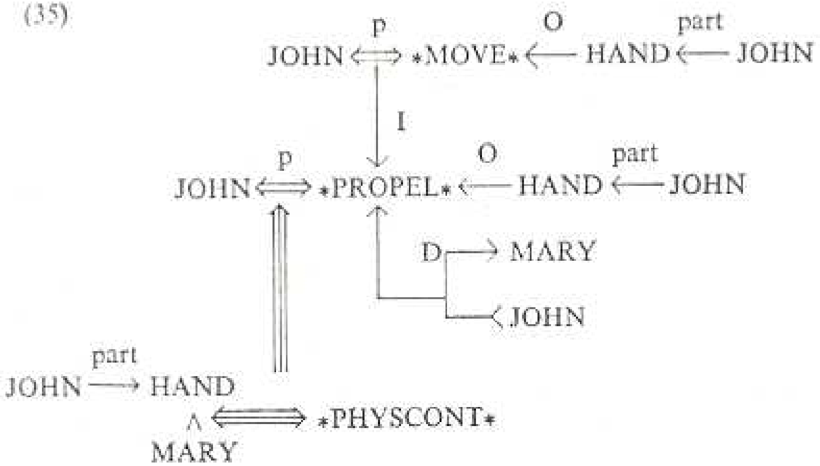


During and after the language analysis the consultation of the above inference processes would yield the following results:

LINGUISTIC	add "hand" as object of PROPEL.
OBJECT AFFECT	add causal "recipient (Mary) be hurt".
BELIEF PATTERN	add potential cause of the entire event as Mary DO cause John be hurt cause John be angry.
INSTRUMENTAL	add instrument of MOVE "hand".
PROPERTY	add predication that John and Mary exist and that John has hands and that they were in the same place at the same time.
INTENTION	add that John knew that it would cause him pleasure if Mary was hurt and that is why he did it.

The graph after analyzer-initiated inferences have filled out the meaning representation, but before MEMORY gains direct processing control is:

(35)



6. The Program

There currently exists at the Stanford Artificial Intelligence Laboratory a functioning program which works in conjunction with the analysis program written by Riesbeck [4] and the generation program written by Goldman [2]. This program is capable of making some but not all of the inferences described here and of generating responses which demonstrate the kind of understanding to which we have been referring in this paper.

We will now describe the theory of the operation of this program and trace in detail one of the examples we have discussed. Please bear in mind that it is the intent of both the program and this paper to be as theoretically correct as possible. Therefore on occasion we have sacrificed efficiency for theory. It was not the intent of this program to do a dazzling job on a few isolated examples. Rather we have tried to produce a program that is easily extendable that will further the cause of computer understanding.

After conceptual analysis of "John hit Mary" is complete, MEMORY gains processing control (MEMORY has already played a passive role during analysis, having been called upon for knowledge of objects and people, and asked to supply the missing linguistic and object-affect information).

Before examining the flow of an example, a brief explanation of MEMORY's data structures and goals is in order. All propositional information is stored in list positional notation, with the predicate first and the conceptual case slots following. The internally-stored form of a proposition is called a *bond*, and is stored as a single entity under a LISP generated atom (*superatom*). In this way propositions are easily embedded, and, except for their bond, look like *simple concepts*. Simple concepts have only an *occurrence set* to define them (superatoms have occurrence sets too). The occurrence set is a set of pointers to superatoms which contain instances of the simple concept. MEMORY is therefore fully two-way linked. The totality of knowledge about a simple concept are those propositions pointed to by the occurrence set.

In addition to bonds and occurrence sets, superatoms have other characteristics. Most important among these are STRENGTH, MODE, TRUTH, REASONS and OFFSPRING. STRENGTH is a measure of how much credibility a proposition has, and usually represents a composite credibility from those propositions from which it arose. MODE modifies the proposition truth-wise (negations are stored as MODE = FALSE). TRUTH is a flag which is TRUE if this proposition is true in the world at the present time. (This one is for convenience, since this information could be determined from the time modifications or nesting of the proposition.) REASONS is the set of superatoms which participated in the generation of this proposition in the system (i.e., what facts were used to infer this proposition), and OFFSPRING is its inverse (i.e., what other propositions has this one played a part in inferring). These last two are very important because they give MEMORY recourse to retrace its paths and modify STRENGTHs, or to discuss its reasoning. There is one last feature of both superatoms and simple concepts: RECENCY. This is the value of the system clock which is stored each time the superatom or concept is accessed. It is chiefly used for reference establishment.

Inferencing is done breadth-first to a heuristically controllable depth. Inferences have the same data structure as described above, namely, each new inference becomes a superatom, complete with its occurrence set and the other properties mentioned. Inferences are organized as lambda functions under predicates, and are invoked directly by conceptualizations. Pattern matching is done within these lambda functions in the form of program tests and branches. Times are processed along with each proposition, and the system emphasizes an awareness of time relationships, since out-of-date propositions are never discarded, but rather modified by new time relations. A forgetting function is viewed as peripheral to the types of tasks we are currently performing. Briefly, these tasks are the following:

(1) To establish referents of all concepts appearing in a conceptual graph. This requires full access to the inference mechanism, and is not compartmentalized as a well-defined preprocessor.

(2) To serve as a passive data bank and access mechanism for the analysis and generation phases. This includes answering simple queries during the analysis such as "is there a concept which is a human and has name John" as well as performing arbitrarily involved proofs. Typical of proof requests are time relation proofs required by the conceptual generator.

(3) To store the analyzed contents of each sentence. This involves (1) as a subtask, and in general involves the storage of a number of subpropositions. Old information is detected as such, so that unless MEMORY has insufficient information to identify an event or state, its existence in MEMORY is discovered. This, of course, applies to the maintenance of simple concepts as well: MEMORY tries to identify all concepts and tokens of concepts and tokens of concepts with existing ones, and notes which it was unable to identify.

(4) To perform appropriateness checking on all peripheral implications of an input. This primarily involves such tasks as making sure that actors are alive and well and in the right places for their actions, and that the actions are reasonable.

(5) To generate unsolicited inferences of the types described earlier and elevate some of them to the status *predictions* of three basic classes in response to every new input. (A prediction is simply an inference the system has chosen to focus on as being noteworthy at some point.) These three classes of predictions are (a) completatory predictions, (b) causal predictions, and (c) result predictions. Completatory predictions augment conceptualizations by supplying a most likely candidate for some missing information. Causal predictions try to relate the input to belief patterns in the computer memory which could explain the reasons behind the input. Result predictions establish possible outcomes caused by the input, and also access belief patterns.

(6) To maintain a record of inferencing and prediction activity, and be able to answer questions about and discuss reasons for inferred information. This capability includes the ability to modify STRENGTHs and MODEs when assumptions which lead to them change at some future time.

(7) To answer "who", "whether", "when" and "why" type questions concerning the conceptualizations it has been given, together with their inferences.

We now return to the example "John hit Mary". The conceptualization has form (36). This is the positional form of the analyzed version (34) shown at the end of Section 5. Notice that, although the words "JOHN", "HAND", etc. were used in that diagram, what the analyzer actually passes to memory are *descriptive sets*: sets of conceptual propositions which MEMORY can use to identify the actual referents of the concepts described. The notation

Cn: {(P1) . . . (Pk)}

is used to denote some concept having descriptive propositions P1, . . . , Pk, which has not yet been identified as a concept with which MEMORY is familiar (the referent has not been determined). For the examples, #⟨word⟩ will stand for the unique concept which "⟨word⟩" references (and will be unambiguous in these examples).

```
(36)
  ((CAUSE ((PROPEL C1:{(ISA—#PERSON)(NAME)—"JOHN")}
                   C2:{(ISA—#HAND)(PART—C1)}
                   C1
                   C3:{(ISA—#PERSON)(NAME—"MARY")}
           ))
           ((PHYSCONT C2 C3))
  )  (TIME—C4:{(ISA—#TIME)(BEFORE—#NOW)})
  )
```

MEMORY's first task is to establish the referents of as many of the simple concepts (C1, . . . , Ck) as possible. [3] discusses this procedure and its problems in some detail, and a short example is included as Appendix B. We will assume here that all referents have been correctly identified. After this phase, the conceptualization has form (37).

(37)

```
((CAUSE ((PROPEL #JOHN #C0001 #JOHN #MARY))
        ((PHYSCONT #C0001 #MARY))
 (TIME— #C0002))
```

where C0001 is the concept in MEMORY for John's hand, C0002 is the concept in MEMORY for the time of the causal event.

Next, MEMORY fragments the conceptualization into subpropositions, each of which will be submitted to the inferencer. The average English sentence contains many conceptual subpropositions. A subproposition is any unit of information which is conveyed directly (without non-analyzer-initiated inference) by a conceptualization. Subpropositions can be classified into three categories: (1) explicit-focussed, (2) explicit-peripheral, and (3) implicit. Explicit subpropositions are always complete conceptualizations, whereas implicit subpropositions are generally communicated through single, isolated dependencies.

To illustrate these categories, consider the sentence:

"The engine of Beverly's new car broke down while she was driving on the freeway late last night."

The explicit-focussed proposition is: "a car engine broke down". This is the "main reason" for the conceptualization's existence. It is not necessarily always the most interesting subproposition for MEMORY to pursue, however.

Some of the explicit-peripheral propositions are:

1. the car is new,
2. the car is owned by Beverly,
3. the time of the incident was late last night,
4. the location of the incident was on the freeway,
5. Beverly was driving a car.

These are additional facts the speaker thought essential to the hearer's understanding of the conceptualization. They are "peripheral" (dependent) in the conceptual dependency sense, and for the purposes of parsing. However, they frequently convey the most interesting information in the conceptualization.

Some of the implicit propositions are:

1. cars have engines as parts,
2. people own things,
3. Beverly performed an action,
4. cars can be *PTRANS*ed (i.e., they are moveable),
5. the car, engine and Beverly were on the freeway (i.e., the actors and objects involved in an event have the event's location).

Briefly, these are very low-level propositions which affirm conceptual case restrictions, and which must strictly adhere to MEMORY's knowledge of normality in the world. These typically lie on the borderline between what was said and what the hearer nearly always infers without further thought.

In the example "John hit Mary", the fragmentation process yields the following subpropositions from the input conceptualization:

1. JOHN PROPELLED SOMETHING
2. A HAND WAS PROPELLED
3. JOHN MOVED SOMETHING
4. A HAND WAS MOVED
5. A HAND IS PART OF JOHN
6. SOMETHING WAS PROPELLED FROM JOHN TO MARY
7. A HAND AND MARY WERE IN PHYSICAL CONTACT
8. JOHN PROPELLED HIS HAND
9. 8 CAUSED 7
10. IT WAS BEFORE "NOW" THAT 1–9 OCCURRED

We do not pursue all of these in the following description, but bear in mind that MEMORY subjects each of the above 10 subpropositions (some of which are redundant in the information they convey) to inferencing.

Having been "perceived" externally, the causal relation (9 above) is stored as a superatom, assigned strength 1.0, given TRUTH T, MODE T and REASONS T (there are no reasons, it is just true). In addition, its superatom is entered on the inference queue, which now has this single entry. Inferences organized under CAUSE are then called. Two nominal inferences with strength propagation factor 1.0 are that the two parts of the causal relation are themselves true: the PROPEL and PHYSCONT propositions are thus inferred with propagated strength still 1.0, TRUTH T, MODE T and REASONS a list of one item: the superatom for the causal proposition. In addition, TIME propositions are created for these two new superatoms using #C0002. These receive STRENGTH 1.0, TRUTH T, MODE T, having as REASONS a list of one item which is the superatom for the causal time proposition. These two new time propositions are not, however, added to the inference list. The PROPEL proposition, when subjected to inferencing will, among other things, look to see if an instrumental is present, and, seeing that one isn't, will add the most likely one: (MOVE #JOHN #C0001 #JOHN #MARY). This will in turn be added to the inference queue. When its inferences are generated, among them will be the inference that #JOHN has at least one movable hand. Were MEMORY to find a contradiction at this point, it would have access to the MOVE completatory inference which produced the contradiction, and would alter its strength of belief and note that a contradiction had occurred. Later, a response concerning this problem might be generated.

Among the other inferences organized under CAUSE, one has an invocation pattern which is matched by this (CAUSE(PROPEL . . .)(PHYSCONT . . .)) pattern. This is the inference that recognizes that someone's PROPELLing an object has caused the contact of that object with an animal. The inference is that the animal is likely to have been hurt:

(38) (NEGCHANGE #MARY #PSTATE)

Notice the reason for organizing this inference under CAUSE rather than PROPEL or PHYSCONT: PROPEL alone says nothing about actual contact, only that an actor has propelled an object in a direction. PHYSCONT alone is not enough, because it also appears in sentences like "John is touching the wall," where there are no such violent dynamics. This pattern also knows that the outcome of a propelling which causes physical contact can lead to different kinds of inferences based on the features of the propelled object and the target object. For example, it knows that to hit a bodypart of an animal is the same as hitting the animal, and that a measure of the amount of injury done is a function of the hardness, heavyness, sharpness, etc. of the propelled object, and of the particular bodypart hit.

The NEGCHANGE inference is thus stored as a superatom and added to the inference queue. Its REASONS are the original CAUSE and the facts that (ISA #MARY #PERSON) and (ISA #PERSON #ANIMAL). Notice that the actual inference rule is not recorded as a reason, since a semblance of it can always be reconstructed from its parts.

This same CAUSE pattern also asserts the actor's volition since it detects no information to the contrary: John wanted this causal relation to exist. This is a general operating assumption of MEMORY: that it is essential at every point in inferencing to keep track of the intentionality of actions. Actions which stand by themselves are always assumed to be volitional. Likewise, causal relationships such as this one (where an action causes a state), are assumed to be the result of the actor's volition. (Deciding an actor's intent in most cases is a difficult problem. [3] discusses problems of this nature in some detail.)

At this point, (39) is stored and entered on the inference queue. Its REASON is simply the original superatom. Notice that MEMORY has now made an important distinction between the physical and intentional components of the event. They will proceed in parallel.

```
(39)
  ((MLOC ((CANCAUSE ((NEGCHANGE #MARY #PSTATE))
                    ((POSCHANGE #JOHN #JOY))))
      #C0003)
   (TIME—#C0002))
                              (#C0003 is John's LTM)
```

We return now to (38). (NEGCHANGE #MARY #HEALTH) accesses inferences organized under NEGCHANGE. MEMORY first checks to determine what caused this situation and finds the REASONS which were generated along with the NEGCHANGE. Had MEMORY not found any REASONS, it would have attempted to apply world knowledge to make a prediction. This knowledge is stored using the predicates CAUSE and CANCAUSE, and is accessed by the MEMORY query: find all probable causes of (NEGCHANGE #PERSON #HEALTH), i.e., find all X such that (CANCAUSE X (NEGCHANGE #PERSON #HEALTH)), and similarly for CAUSE. This situation would occur in the following type of story: "Mary was hurt." "John had hit her with a rock." where one member of the predicted set is borne out by the next line of the story. Such a process is called "knitting" (see [3]), and is the chief measure of "understanding" in several-line stories.

In addition to this determination of causality which was trivially satisfied in this case, MEMORY detects applicability of the following belief pattern: when a person undergoes a NEGCHANGE (on any scale, since all scales are positive), he will want to undergo a POSCHANGE on that scale. MEMORY thus infers (40):

```
(40)
  ((MLOC((CANCAUSE((POSCHANGE #MARY #HEALTH))
                  ((POSCHANGE #MARY #JOY))))
       #C0004)
   (TIME—#C0005))
                              #C0004 is Mary's LTM,
                              #C0005 is (AFTER C0002)
```

This subsequently will be detected by the belief pattern (organized under MLOC) that when a person wants a future event, he will perform some action to try to achieve that event or state. Once again, CAUSE and CANCAUSE information is called into play to predict Mary's likely actions. An example of this type of information is:

```
  ((CANCAUSE((INGEST #PERSON #MEDICINE))
            ((POSCHANGE #PERSON #HEALTH))))
```

Using information collected in this manner, a prediction of Mary's future actions is made. This prediction has the form of a bond, and indicates that any or all of the actions listed are possible. Notice that only actions are being predicted. If some causes of the state the actor desires are not actions but rather states or statechanges themselves, further CANCAUSE and CAUSE chains are considered until an action is found. For instance, suppose Mary wants a NEGCHANGE on her own health scale. *One cause of a NEGCHANGE on the health might be to have one's heart in PHYSCONT with* a knife. Since this is not an action, memory must be searched for things which could cause the required PHYSCONT. Among them would be the action of PROPELling the knife to that location. This PROPEL might then be a valid action prediction for Mary at that point.

At this point, (41) is generated, and inferencing on this line is stopped.

(41)
((PREDICTIONSET #MARY
((INGEST #MARY #MEDICINE #UNSPEC #C0006))
((PTRANS #MARY #MARY #UNSPEC #C0007))))
where C0006 is Mary's INSIDES,
C0007 is a token of a #HOSPITAL

We now return to (39). This inference accesses the belief pattern organized under MLOC which we have labelled VENGEANCE: if a NEGCHANGE (on any scale) of a person, P1, would cause a POSCHANGE on the joy scale for someone else, P2, then P2 must be angry at P1. MEMORY therefore infers (42):

(42) ((MFEEL #JOHN #ANGER #MARY)(TIME—C0002))

Stored under MFEEL is the belief pattern that the reason people are in a state of directed anger toward another person is probably that the second person did something which caused a NEGCHANGE on some scale of the first person. MEMORY first looks to see if Mary is known to have done something which caused a NEGCHANGE in John. In this example it finds none. Had one been found from a previous sentence, MEMORY would have again "knitted" one piece of knowledge with an existing one. In this example, having found no actions on the part of Mary, MEMORY generates a prediction about Mary's PAST actions, once again utilizing CAUSE and CANCAUSE knowledge of the world. After making prediction (43), MEMORY also poses a question of the form "What did Mary do?", stores the question, and notes its potential answer as being of interest to the prediction just made.

(43)
((PREDICTIONSET #MARY
((CAUSE ((PROPEL #MARY #PHYSOBJ #UNSPEC #JOHN))
((PHYSCONT #PHYSOBJ #JOHN))))
((ATRANS#MARY #PHYSOBJ #JOHN #MARY))))

i.e., Mary either hit John first, or took something from him. (It should be clear that we are not intending to specify an exhaustive prediction list. Rather we seek to demonstrate the PROCESSES which occur in MEMORY.) At this point MEMORY stops inferencing and poses the question "What did Mary do to John?"

(44)
((CAUSE (DO #MARY *?*)((NEGCHANGE #JOHN #UNSPECIFIED)))
(TIME—C0010))
where C0010 is BEFORE C0002.

To summarize, MEMORY has taken the conceptual analysis underlying an English sentence and generated new probabilistic information from it in an attempt to relate it to knowledge MEMORY may already have stored. The new information took three basic forms: (a) predictions about the causes of the input, (b) predictions about the possible results of the input, and (c) predictions about future and past actions of people. The effects of inferencing are seen at the end either in the form of a question or a comment which indicates that the sentence indeed interacted with some of MEMORY's knowledge and belief patterns.

Appendix A. Computer Examples

What follows is output from the MARGIE system currently operating at Stanford. MARGIE is a combination of three programs each of whose output is shown here. The analysis program produces conceptual structures from a given input sentence. The memory program stores this output in a special format and makes inferences about it based on its knowledge of the world. It then recodes these inferences into Conceptual Dependency structures. These structures are then read by a generating program that codes them into semantic structures that are English based (after Fillmore [1]). A modified version of a program written by Simmons [9], then encodes these structures into English.

The examples presented here are intended only to show the flavor of the inference-making program. The entire system is quite a bit more powerful than these examples demonstrate. That is MARGIE can answer questions about what it has been told, ask questions about what it would like to know, as well as parse sentences more complex than those shown here. Here we merely want to indicate the inference capability.

In the interest of space, we have manually edited out some of the less interesting (generally repetitive from example to example) inferences. This explains the apparent discrepancy between the number of "INFERENCES" and "THINGS TO SAY" in the following examples.

TYPE INPUT
*(JOHN TOLD MARY THAT BILL WANTS A BOOK)

OUTPUT FROM PARSER:
TIM00: ((VAL *T*))
TIM01: ((BEFORE TIM00 X))

TIM02: ((AFTER TIM00 X))
TIM03: ((AFTER TIM00 X))
TIM04: ((AFTER TIM00 X))
((ACTOR (JOHN1) ⇔ (*MTRANS*) TO (*CP* PART (MARY1) REF-(*THE*)) FROM (*CP* PART (JOHN1) REF (*THE*)) MOBJECT - ((CON ((CON ((ACTOR (*ONE1*) ⇔ (*ATRANS*) OBJECT (BOOK1 - REF (*A*)) TO (BILL1) FROM (*ONE1*)) TIME (TIM04)) ⇐C ((ACTOR (BILL1) ⇔ T (*JOY*) ⇔ F (*JOY*)) INC (2) TIME (TIM03)))) ⇔ (*MLO-C* VAL (*LTM* PART (BILL1) REF (*THE*)))) MODE (NIL) FOCUS ((⇔ VAL PART)) TIME (TIM00))) TIME (TIM01))

PARTIALLY INTEGRATED RESULT:
((*MTRANS* (#JOHN1) ((*MLOC* ((CANCAUSE ((*ATRANS* (#SO-MEONE) (G0012) (#SOMEONE) (#BILL1)) (TIME—G0014))) ((STATE-CHANGE (#BILL1) (#JOY) #UNSPECIFIED #UNSPECIFIED) (TI-ME—(G0017)) (INC—(#TWO)))) (G0019)) (TIME—(G0015))) (G0022) (G-0025)) (TIME—(G0028)))
INTEGRATION RESULT: G0038
INFERENCES:
(G0040 G0036 G0047 G0057 G0056 G0055 G0063 G0062 G0065 G0070)
THINGS TO SAY:
((ACTOR (BOOK REF (*A*)) ⇔ (*XABT* VAL (*?*))))

(A BOOK ABOUT WHAT)

((CON ((CON ((ACTOR (MARY) ⇔ (*ATRANS*) OBJECT (BOOK REF (*A*)) FROM (MARY) TO (BILL)) TIME (G0014) FOCUS (ACTOR))) ⇐C ((ACTOR (JOHN) ⇔F (*JOY*) ⇔T (*JOY)) TIME (G0017) INC (2)))) - ⇔ (*MLOC* VAL (*LTM* REF (*A*) PART (JOHN)))) CERTAINTY - (0.8) TIME (G0015))

(JOHN POSSIBLY WANTS MARY TO GIVE BILL A BOOK)

((CON ((CON ((CON ((ACTOR (*ONE1*) ⇔ (*ATRANS*) OBJECT (BO-OK REF (*A*)) FROM (*ONE1*) TO (BILL)) TIME (G0014) FOCUS ((A-CTOR))) ⇐C ((ACTOR (BILL) ⇔F (*JOY*) ⇔T (*JOY*)) TIME (G0017) - INC (2)))) ⇔ (*MLOC* VAL (*LTM* REF (*A*) PART (BILL)))) TIME - (G0015)) ⇔ (*MLOC* VAL (*CP* REF (*A*) PART (MARY)))) TS (G002-8))

(MARY EXPECTED BILL TO WANT A BOOK)

((CON ((CON ((ACTOR (*ONE1*) ⇔ (*ATRANS*) OBJECT (BOOK RE-F (*A*)) FROM (*ONE1*) TO (BILL)) TIME (G0014) FOCUS ((ACTOR))-) ⇐C ((ACTOR (BILL) ⇔F (*JOY*) ⇔T(*JOY*)) TIME (G0017) INC (2)))-⇔ (*MLOC*) VAL (*LTM* REF (*A*) PART (BILL)))) TIME (G0015))
(BILL WANTS A BOOK)

((CON ((CON ((CON ((ACTOR (*ONE1*) ⇔ (*ATRANS*) OBJECT (BO-OK REF (*A*)) FROM (*ONE1*) TO (BILL)) TIME (G0014) FOCUS ((A-CTOR))) ⇐C ((ACTOR (BILL) ⇔F (*JOY*) ⇔T (*JOY*)) TIME (G0017) - INC (2)))) ⇔ (*MLOC* VAL (*LTM* REF (*A*) PART (BILL)))) TIME (-G0015)) ⇔ (*MLOC* VAL (*LTM* REF (*A*) PART (MARY))))

(MARY KNOWS BILL WANTS A BOOK)

((ACTOR (MARY) ⇔ (*ATRANS*) OBJECT (BOOK REF (*A*)) FROM-(MARY) TO (BILL)) TIME (G0017) CERTAINTY (0.8))

(MARY MIGHT GIVE BILL A BOOK)

((CON ((CON ((ACTOR (BOOK REF (*A*)) ⇔ (*POSS* VAL (BILL))) TS-(G0014) TIME (G0066)) ⇐C ((ACTOR (BILL) ⇔F (*JOY*) ⇔T (*JOY*)) - INC (2) TIME (G0067)))) ⇔ (*MLOC* VAL (*LTM* PART (BILL) REF (-*THE*))))

(BILL THINKS HE WOULD LIKE TO COME TO HAVE A BOOK)

((CON ((CON ((ACTOR (BOOK REF (*A*)) ⇔ (*POSS* VAL (*ONE1*)))-TF (G0014) TIME (G0070)) ⇐C ((ACTOR (BILL) ⇔F (*JOY*) ⇔T (*JOY-*)) INC (2) TIME (G0071)))) ⇔ (*MLOC* VAL (*LTM* PART (BILL) RE-F (*THE*)))))

(BILL THINKS HE WOULD LIKE SOMEONE TO CEASE TO HAVE A - BOOK)

((CON ((CON ((ACTOR (*ONE1*) ⇔ (*PTRANS*) OBJECT (BOOK REF-(*A*)) FROM (*ONE1*) TO (BILL)) FOCUS ((ACTOR)) TIME (G0074)) - ⇐C ((ACTOR (BILL) ⇔F(*JOY*) ⇔T (*JOY*)) INC (2) TIME (G0075)))) - ⇔ (*MLOC* VAL (*LTM* PART (BILL) REF (*THE*)))))

(BILL WANTS TO GET A BOOK FROM SOMEONE)

((CON ((CON ((ACTOR (BOOK REF (*A*)) ⇔ (*LOC* VAL (BILL))) TS - (G0014) TIME (G0078)) ⇐C ((ACTOR (BILL) ⇔F (*JOY*) ⇔T (*JOY*)) I-NC (2) TIME (G0079)))) ⇔ (*MLOC* VAL (*LTM* PART (BILL) REF (*-THE*)))))

(BILL THINKS HE WOULD LIKE A BOOK TO COME TO BE NEAR HIM))

((CON ((CON ((ACTOR (BOOK REF (*A*)) ⇔ (*LOC* VAL (*ONE*1))) - TF (G0014) TIME (G0082)) ⇐C ((ACTOR (BILL) ⇔F (*JOY*) ⇔T (*JOY-*)) INC (2) TIME (G0083)))) ⇔ (*MLOC* VAL (*LTM* PART (BILL) RE-F (*THE*)))))

(BILL THINKS HE WOULD LIKE A BOOK TO CEASE TO BE NEAR SOMEONE)

((CON ((CON ((ACTOR (BILL) ⇔ (*MTRANS*) MOBJECT (*CONCEP-TS*) FROM (BOOK REF (*A*)) TO (*CP* PART (BILL)) INST ((ACTOR-(BILL) ⇔ (*LOOK—AT*) OBJECT (BOOK REF (*A*))))) FOCUS ((ACT-OR)) TIME (G0086)) ⇔C ((ACTOR (BILL) ⇔F (*JOY*) ⇔T (*JOY*)) IN-C (2) TIME (G0087)))) ⇔ (*MLOC* VAL (*LTM* PART (BILL) REF (*T-HE*)))))

(BILL WANTS TO READ A BOOK)

TYPE INPUT
*(JOHN HIT MARY)

OUTPUT FROM PARSER:
TIM00: ((VAL *T*))
TIM01: ((BEFORE TIM00 X))
((CON ((ACTOR (JOHN1) ⇔ (*PROPEL*) OBJECT (*HAND* PART (J-OHN1)) TO (MARY1) FROM (JOHN1) INST ((ACTOR (JOHN1) ⇔ (*M-OVE*) OBJECT (*HAND* PART (JOHN1))))) TIME (TIM01) MODE (NI-L)) ⇐ ((ACTOR (*HAND* PART (JOHN1)) ⇔ (*PHYSCONT* VAL (M-ARY1))) TIME (TIM01) MODE (NIL) FOCUS (CON ACTOR))))

PARTIALLY INTEGRATED RESULT:
((CAUSE ((*PROPEL* (#JOHN1) (G0009) (#JOHN1) (#MARY1)) (TI-ME—(G0012)) (INST—((*MOVE* (#JOHN1) (G0009) (#UNSPECIFIE-D) (#UNSPECIFIED))))) ((*PHYSCONT* (G0009) (#MARY1)) (TIME—(G0012)))))

INTEGRATION RESULT: G0021

INFERENCES:
(G0023 G0022 G0016 G0019 G0024 G0026 G0027)

THINGS TO SAY:
((CON ((CON ((ACTOR (MARY) ⇔F (*PSTATE*) ⇔T (*PSTATE*)) IN-C (−2) CERTAINTY (1.0) TIME (G0031)) ⇐C ((ACTOR (JOHN) ⇔F (*J-OY*) ⇔T (*JOY*)) INC (2) TIME (G0032)))) ⇔ (*MLOC* VAL (*LTM* -PART (JOHN) REF (*THE*)))) CERTAINTY (1.0) TIME (G0012))
(JOHN WANTED MARY TO BECOME HURT)
((ACTOR (MARY) ⇔F (*PSTATE*) ⇔T (*PSTATE*)) INC (—2) CERT-AINTY (1.0) TIME (G0012))

(MARY BECAME HURT)

((ACTOR (JOHN) ⇔ (*PROPEL*) OBJECT (*HAND* REF (*A*) PART -(JOHN)) FROM (JOHN) TO (MARY) INST ((ACTOR (JOHN) ⇔ (*MOV-E*) OBJECT (*HAND* REF (*A*) PART (JOHN)) FROM (*ONE*) TO (-*ONE*)) FOCUS ((ACTOR)))) TIME (G0012) FOCUS ((ACTOR)) CERT-AINTY (1.0))

(JOHN SWUNG HIS HAND TOWARD MARY)

((ACTOR (*HAND* REF (*A*) PART (JOHN)) ⇔ (*PHYSCONT* VAL (-MARY))) TIME (G0012) CERTAINTY (1.0))

(JOHNS HAND TOUCHED MARY)

((ACTOR (JOHN) ⇔ (*MFEEL*) MOBJECT (*ANGER*) TO (MARY)) -FOCUS ((ACTOR)) CERTAINTY (1.0) TIME (G0012))

(JOHN WAS ANGRY AT MARY)

((CON ((CON ((ACTOR (MARY) ⇔F (*PSTATE*) ⇔T (*PSTATE*)) IN-C (2) TIME (G0035)) ⇐C ((ACTOR (MARY) ⇔F (*JOY*) ⇔T (*JOY*)) I-NC (2) TIME (G0036)))) ⇔ (*MLOC* VAL (*LTM* PART (MARY) REF -(*THE*)))) CERTAINTY (1.0))

(MARY WANTS TO FEEL BETTER)

((CON ((ACTOR (MARY) ⇔ (*?*)) TIME (G0028) FOCUS ((ACTOR)) C-ERTAINTY (1.0)) ⇐ ((ACTOR (JOHN) ⇔T (*ONE*) ⇔F (*ONE*)) INC -(—2) TIME (G0028) FOCUS (ACTOR))

(WHAT DID MARY DO TO JOHN)

TYPE INPUT
*(JOHN ADVISED MARY TO SELL BILL A BANANA)

OUTPUT FROM PARSER:
TIM00: ((VAL *T*))
TIM01: ((BEFORE TIM00 X))
TIM02: ((AFTER TIM01 X))
TIM 03: ((AFTER TIM01 X))
((ACTOR (JOHN1) ⇔ (*MTRANS*) TO (*CP* PART (MARY1) REF (*-THE*)) FROM (*CP* PART (JOHN1) REF (*THE*)) MOBJECT ((CON (-(CON ((ACTOR (MARY1) ⇔ (*ATRANS*) OBJECT (BANANA1 REF (*-A*)) TO (BILL1) FROM (MARY1)) TIME (TIM03)) ⇐⇒ ((ACTOR (BILL1)-⇔ (*ATRANS*) OBJECT (*MONEY* REF (*A*)) TO (MARY1) FROM (-BILL1)) FOCUS ((CON ACTOR)) TIME (TIM03)))) ⇐C ((ACTOR (MAR-Y1) ⇔T (*JOY*) ⇔F (*JOY*)) INC (2) TIME (TIM02) MODE (NIL))))) F-OCUS ((ACTOR)) MODE (NIL) TIME (TIM01))

PARTIALLY INTEGRATED RESULT:
((*MTRANS* (#JOHN1) ((CANCAUSE ((DUALCAUSE ((*ATRANS* (-#MARY1) (G0004) (#MARY1) (#BILL1)) (TIME—(G0006))) ((*ATRA-NS* (#BILL1) (G0013) (#BILL1) (#MARY1)) (TIME—(G0006))))) ((ST-ATECHANGE (#MARY1) (#JOY) #UNSPECIFIED #UNSPECIFIE-D) (TIME—(G0015)) (INC—(#TWO))))) (G0017) (G0020)) (TIME—(G00-07)))

```
INTEGRATION RESULT: G0032
INFERENCES:
(G0049 G0045 G0040 G0031 G0066 G0023 G0086 G0083 G0084 G0099 G01-
00)
THINGS TO SAY:
((CON ((CON ((CON ((ACTOR (MARY) ⇔ (*ATRANS*) OBJECT (BAN-
ANA REF (*A*)) FROM (MARY) TO (BILL)) TIME (G0006) FOCUS ((A-
CTOR))CERTAINTY (1.0))⇐⇒((ACTOR(BILL)⇔(*ATRANS*)OBJEC-
T (*MONEY* REF (*A*)) FROM (BILL) TO (MARY)) TIME (G0006) F-
OCUS ((ACTOR))))) ⇐C ((ACTOR (MARY) ⇔F (*ONE*) ⇔T (*ONE*)) -
TIME (G0015) INC (2))) CERTAINTY (1.0)) ⇔ (*MLOC* VAL (*LTM* R-
EF (*A*) PART (JOHN)))) TIME (G0007) CERTAINTY (1.0))

(JOHN BELIEVES THAT MARY WOULD BENEFIT BY MARY SELLS
BILL A BANANA)

((CON ((CON ((CON ((ACTOR (MARY) ⇔ (*ATRANS*) OBJECT (BAN-
ANA REF (*A*)) FROM (MARY) TO (BILL)) TIME (G0006) FOCUS ((A-
CTOR))CERTAINTY(1.0))⇐⇒((ACTOR(BILL) ⇔(*ATRANS*)OBJEC-
T (*MONEY* REF (*A*)) FROM (BILL) TO (MARY)) TIME (G0006) F-
OCUS ((ACTOR))))) ⇔C ((ACTOR (MARY) ⇔F (*ONE*) ⇔T (*ONE*)) -
TIME (G0015) INC (2))) CERTAINTY (1.0)) ⇔ (*MLOC* VAL (*CP* REF-
(*A*) PART (MARY)))) TS (G0007) CERTAINTY (1.0))

(MARY BEGAN THINKING ABOUT MARY SELLS BILL A BANANA
BENEFITS MARY)

((CON ((CON ((ACTOR (MARY) ⇔ (*ATRANS*) OBJECT (BANANA -
REF (*A*)) FROM (MARY) TO (BILL)) TIME (G0006) FOCUS ((ACTO-
R))CERTAINTY (1.0))⇐⇒((ACTOR (BILL)⇔(*ATRANS*) OBJECT (*-
MONEY* REF (*A*)) FROM (BILL) TO (MARY)) TIME (G0006) FOCU-
S ((ACTOR))))) ⇐C ((ACTOR (MARY) ⇔F (*ONE*) ⇔T (*ONE*)) TIM-
E (G0015) INC (2))) CERTAINTY (1.0))

(MARY CAN BENEFIT FROM MARY SELL BILL A BANANA)

((ACTOR (MARY) ⇔ (*MBUILD*) FROM ((ACTOR (MARY) ⇔ (*AT-
RANS*) OBJECT (BANANA REF (*A*)) FROM (MARY) TO (BILL)) TI-
ME (G0006) FOCUS ((ACTOR)) CERTAINTY (1.0)) TO (*ONE*)) CERT-
AINTY (1.0))

(MARY CONSIDERED GIVING BILL A BANANA)

((ACTOR (MARY) ⇔ (*ATRANS*) OBJECT (BANANA REF (*A*)) FR-
OM (MARY) TO (BILL)) TIME (G0033) FOCUS ((ACTOR)) CERTAINT-
Y (0.60))

(MARY POSSIBLY WILL GIVE BILL A BANANA)

((CON ((CON ((ACTOR (BILL) ⇔ (*ATRANS*) OBJECT (MONEY REF -
(*A*)) FROM (BILL) TO (MARY)) TIME G0033)) ⇐ ((ACTOR (MARY) -
⇔F (*ONE*) ⇔ T (*ONE*)) TIME (G0033) INC (2)))) ⇔ (*MLOC* VAL
(*LTM* REF (*A*) PART (JOHN)))) TIME (G0006) CERTAINTY (0.50))

(JOHN POSSIBLY BELIEVES THAT MARY WOULD BENEFIT FROM
BILL GIVE MARY MONEY)

((CON ((CON ((ACTOR (MARY) ⇔ (*ATRANS*) OBJECT (BANANA -
REF (*A*)) FROM (MARY) TO (BILL)) TIME (G0033)) ⇐C ((ACTOR (-
BILL) ⇔F (*ONE*) ⇔T (*ONE*)) TIME (G0033) INC (2)))) ⇔ (*MLOC* -
VAL (*LTM* REF (*A*) PART (JOHN)))) TIME (G0006) CERTAINTY (-
0.50))

(JOHN POSSIBLY BELIEVES THAT BILL WOULD BENEFIT FROM
MARY GIVE BILL A BANANA)

((CON ((CON ((ACTOR (BILL) ⇔ (*INGEST*) OBJECT (BANANA REF-
(*A*))) TIME (G0034)) ⇐C ((ACTOR (BILL) ⇔F (*JOY*) ⇔T (*JOY*)) T-
IME (G0034) INC (2)))) ⇔ (*MLOC* VAL (*LTM* REF (*A*) PART (BIL-
L)))) TIME (G0006) MODE ((*?*)))

(DOES BILL WANT TO EAT A BANANA)
```

```
TYPE INPUT
*(JOHN PREVENTED MARY FROM HITTING BILL BY CHOKING -
MARY)

OUTPUT FROM PARSER:
TIM00: ((VAL *T*))
TIM01: ((BEFORE TIM00 X))
TIM02: ((BEFORE TIM01 X))
((CON ((CON ((ACTOR (JOHN1) ⇔ (*GRASP*) OBJECT (*NECK* PAR-
T (MARY1))) TIME (TIM02)) ⇐ ((ACTOR (MARY1) ⇔ (*INGEST*) OB-
JECT (*AIR* REF (*A*)) FROM (*MOUTH* PART (MARY1)) TO (*IN-
SIDE* PART (MARY1))) TIME (TIM02) MODE ((*CANNOT*)))) FOCU-
S (CON ACTOR)) ∧ ((CON ((ACTOR (MARY1) ⇔ (*INGEST*) OBJEC-
T (*AIR* REF (*A*)) FROM (*MOUTH* PART (MARY1)) TO (*INSID-
E* PART (MARY1))) TIME (TIM02) MODE ((*CANNOT*))) ⇐ ((CON ((-
ACTOR (MARY1) ⇔ (*PROPEL*) OBJECT (*HAND* PART (MARY1))-
TO (BILL1) FROM (MARY1) INST ((ACTOR (MARY1) ⇔ (*MOVE*) O-
BJECT (*HAND* PART (MARY1))))) TIME (TIM01) MODE ((*CANNO-
T*))) ⇐ ((ACTOR (*HAND* PART (MARY1)) ⇔ (*PHYSCONT* VAL (-
BILL1))) TIME (TIM01) MODE ((*CANNOT*)) FOCUS (CON ACTOR))-
) MODE ((*NEG*)))) FOCUS ((CON ACTOR)))))
PARTIALLY INTEGRATED RESULT:
```

((ANDX ((CAUSE ((*GRASP* (#JOHN1) (G0002)) (TIME—(G0005))) ((-CANNOT ((*INGEST* (#MARY1) (G0012) (G0016) (G0021)) (TIME—(-G0005))))))) ((CAUSE ((CANNOT ((*INGEST* (#MARY1) (G0012) (G00-16) (G0021)) (TIME—(G0005))))) ((NOT ((CAUSE ((CANNOT ((*PROPEL* (#MARY1) (G0024) (#MARY1) (#BILL1)) (TIME—(G0006)) (INST-—((*MOVE* (#MARY1) (G0024) (#UNSPECIFIED) (#UNSPECIFIE-D))))))) ((CANNOT ((*PHYSCONT* (G0024) (#BILL1)) (TIME—G0006))-))))))))))
INTEGRATION RESULT: G0047
INFERENCES:
(G0032 G0046 G0027 G0031 G0035 G0045 G0053 G0058 G0066 G0079 G00-70 G0074 G0079)
THINGS TO SAY:
((CON ((ACTOR (JOHN) ⇔ (*GRASP*) OBJECT (*NECK* REF (*A*) P-ART (MARY))) TIME (G0005) FOCUS ((ACTOR)) CERTAINTY (1.0)) -⇐ ((ACTOR (MARY) ⇔ (*INGEST*) OBJECT (*AIR* REF (*A*)) FRO-M (*MOUTH* REF (*A*)) TO (*INSIDE* REF (*A*))) TIME (G0005) F-OCUS ((ACTOR)) MODE ((*CANNOT*)) CERTAINTY (1.0))) CERTAI-NTY (1.0))

(JOHN CHOKED MARY)

((CON ((ACTOR (MARY) ⇔ (*INGEST*) OBJECT (*AIR* REF (*A*)) F-ROM (*MOUTH* REF (*A*)) TO (*INSIDE* REF (*A*))) TIME (G0005)-FOCUS ((ACTOR)) MODE ((*CANNOT*)) CERTAINTY (1.0)) ⇐ ((CON-((ACTOR (MARY) ⇔ (*PROPEL*) OBJECT (*HAND* REF (*A*) PART-(MARY)) FROM (MARY) TO (BILL) INST ((ACTOR (MARY) ⇔ (*MO-VE*) OBJECT (*HAND* REF (*A*) PART (MARY)) FROM (*ONE*) T-O (*ONE*)) FOCUS ((ACTOR)))) TIME (G0006) FOCUS ((ACTOR)) MO-DE ((*CANNOT*))) ⇐ ((ACTOR (*HAND* REF (*A*) PART (MARY)) -⇔ (*PHYSCONT* VAL (BILL))) TIME (G0006) MODE ((*CANNOT*)))) -MODE ((*NEG*)) CERTAINTY (1.0))) CERTAINTY (1.0))

(MARY NOT HIT BILL BECAUSE MARY WAS UNABLE TO BREATH E)

((ACTOR (JOHN) ⇔ (*GRASP*) OBJECT (*NECK* REF (*A*) PART (-MARY))) TIME (G0005) FOCUS ((ACTOR)) CERTAINTY (1.0))

(JOHN GRABBED MARYS NECK)

((ACTOR (MARY) ⇔ (*INGEST*) OBJECT (*AIR* REF (*A*)) FROM (-*MOUTH* REF (*A*)) TO (*INSIDE* REF (*A*))) TIME (G0005) FOCU-S ((ACTOR)) MODE ((CANNOT*)) CERTAINTY (1.0))
(MARY WAS UNABLE TO BREATHE)
((ACTOR (MARY) ⇔F (*PSTATE*) ⇔T (*PSTATE*)) INC (−2) TIME (-G0005) CERTAINTY (1.0))

(MARY BECAME HURT)

((CON ((ACTOR (MARY) ⇔ (*PROPEL*) OBJECT (*HAND* REF (*A*-) PART (MARY)) FROM (MARY) TO (BILL) INST ((ACTOR (MARY) -⇔ (*MOVE*) OBJECT (*HAND* REF (*A*) PART (MARY)) FROM (*-*ONE*) TO (*ONE*)) FOCUS ((ACTOR)))) TIME (G0006) FOCUS ((AC-TOR)) MODE ((*CANNOT*))) ⇐ ((ACTOR (*HAND* REF (*A*) PART -(MARY)) ⇔ (*PHYSCONT* VAL (BILL))) TIME (G0006) MODE ((*CA-NNOT*)))) MODE ((*NEG*)) CERTAINTY (1.0))

(MARY NOT HIT BILL)

((CON ((CON ((CON ((ACTOR (MARY) ⇔ (*PROPEL*) OBJECT (*HA-ND* REF (*A*) PART (MARY)) FROM (MARY) TO (BILL)) TIME (G0-005)) ⇐ ((ACTOR (*HAND* REF (*A*) PART (MARY)) ⇔ (*PHYSCON-T* VAL (BILL))) TIME (G0005)))) ⇐C ((ACTOR (MARY) ⇐F (*JOY*) ⇔ T (*JOY*)) INC (2)))) ⇔ (*MLOC* VAL (*LTM* PART (MARY) REF (*-THE*)))) CERTAINTY (1.0) TIME (G0005))

(MARY WANTED TO HIT BILL)

((CON ((CON ((ACTOR (BILL) ⇔F (*PSTATE*) ⇔T (*PSTATE*)) INC (-−2) CERTAINTY (1.0) TIME (G0006)) ⇐C ((ACTOR (MARY) ⇔F (*JO-Y*) ⇔T (*JOY*)) INC (2) TIME G(0006)))) ⇔ (*MLOC* VAL (*LTM* PA-RT (MARY) REF (*THE*)))) CERTAINTY (1.0) TIME (G0005))

(MARY WANTED BILL TO BECOME HURT)

((ACTOR (MARY) ⇔ (*MFEEL*) MOBJECT (*ANGER*) TO (BILL)) F-OCUS ((ACTOR)) CERTAINTY (1.0) TIME (G0005))

(MARY WAS ANGRY AT BILL)

((CON ((ACTOR (BILL) ⇔ (*?*)) TIME (G0028) FOCUS ((ACTOR)) CE-RTAINTY (1.0)) ⇐ ((ACTOR (MARY) ⇔ T(*ONE*) ⇔F (*ONE*)) INC (-−2) TIME (G0028) FOCUS (ACTOR))

(WHAT DID BILL DO TO MARY)

((ACTOR (JOHN) ⇔ (*MFEEL*) MOBJECT (*ANGER*) TO (MARY)) F-OCUS ((ACTOR)) CERTAINTY (0.8) TS (G0005))

(JOHN POSSIBLY BECAME ANGRY AT MARY)

Appendix B. Inference and Reference Establishment

We include this appendix to illustrate briefly how inferences are useful in establishing references to tokens of real world concepts. A scheme has been devised which permits MEMORY to proceed with other aspects of "understanding" even though all referents may not have been established before understanding begins. This scheme also provides for the eventual establishment of these referents as another goal of the inference process. It is not hard to see that, in general, the solution of the reference problem for some concept can involve arbitrarily intimate and detailed interaction with the deductive processes of MEMORY, and that these processes must be designed to function with concepts whose features are not completely known.

Consider the sentence

"Andy's diaper is wet."

Assume a very simple situation for the sake of example: that MEMORY knows of exactly two concepts, MC1, MC2 such that

X ∈ {MC1, MC2}:

(D1) (ISA X #PERSON)
(D2) (NAME X "ANDY")

(i.e., MEMORY knows two people by the name Andy). However, possibly in addition to much other information, MEMORY also knows

(AGE MC1 #12 MONTHS) and (AGE MC2 #25 YEARS).

This is a typical reference dilemma: no human hearer would hesitate in the correct identification of "Andy" in this sentence using these pieces of knowledge (in no particular context). Yet the natural order of "establish references first, then infer" simply does not work in this case. In order to begin inferencing, the referent of "Andy" is required (i.e., access to the features of C1 in memory), but in order to establish the referent of "Andy" some level of deduction must take place. This is something of a paradox on the surface.

Actually, the fault lies in the assumption that reference establishment and inferencing are distinct and sequential processes. The incorrectness of this assumption is but another example of the recurring theme that NO aspect of natural language processing (from phonology to story comprehension) can be completely compartmentalized. In reality, reference establishment and inferencing are in general so intimately interrelated so as to be functionally almost indistinguishable. Nevertheless, there is an interesting sequence of processing which will solve this class of reference problem.

(We point out that there are many other interesting inferences to be made from this sentence. A glaring one is, of course, "what kind of fluid?" The inference which supplies this information is an example of LINGUISTIC-INFERENCE, and is quite similar to the case in which "hand" is inferred as the missing object implied by "hit". One difference is that, while "hand" is predicted from an ACT, "urine" is predicted from a PP, namely "diaper". Another difference is that "hand" is supplied in response to MISSING information, while "urine" is supplied to make a general concept more specific. We will ignore this and all other inferences not needed in the following description.)

At the point the reference problem is undertaken, the state of this conceptualization is the following:

(*LOC* C1: {(ISA C1 #FLUID)}
C2: {(ISA C2 #DIAPER)
(*POSS* C2 C3: {(ISA C3 #PERSON)
(NAME C3 "ANDY")})})

i.e., there is some fluid located at the diaper which is possessed by a person whose name is Andy. Once the correct "#ANDY" has been identified, the referent of "diaper" can be established, using the principle that explicit subpropositions of a certain class (*POSS* among these) should appease the reference-finding mechanism. That is, "The diaper", occurring out of context with no conceptual modification, is referentially ambiguous, while "The diaper possessed by X" is a signal to MEMORY that the speaker has included what he feels is sufficient information either to identify or create the token of a diaper being referenced. However, this diaper processing must wait for the *POSS* proposition to be stored in MEMORY and this in turn involves the determination of reference to the possessor (the problem at hand). The reference to #FLUID is simply solved: the concept #FLUID is invoked as part of the definition of what it is to be wet, and MEMORY simply creates a token of this mass-noun concept. MEMORY realizes that references to mass nouns frequently occur with no explicit conceptual modification, and does not bother to identify them further unless contradictory inferences result from them later on. This token of #FLUID stands for the fluid which is currently in "Andy"'s diaper. Now only the person referent remains to be solved.

Using its standard intersection search, MEMORY uses the two descriptive propositions to locate MC1 and MC2 as possible candidates for the referent of P. Since no more can be done at this point, MEMORY creates a concept, MC3 (which will turn out in this case to be temporary) whose occurrence set (see beginning of Section 6) consists of the two propositions D1 and D2. In addition, MEMORY notes that this concept has been created as the result of ambiguous reference (specifically, it adds MC3 to the list !REFUN-ESTABLISHED). This done, a token of a diaper which is possessed by MC3 can now be created. This token too, by virtue of its referencing another possibly incorrectly identified concept in MEMORY, will be subject to reference reevaluation, pending identification of MC3. At this point, MEMORY has an internal form of the conceptualization, albeit incomplete, so inferencing begins.

Of interest to this example is the subproposition "MC3 possesses a diaper". Subpropositions are briefly discussed in Section 6. [3] describes in more detail the methods by which all subpropositions are extracted for examination by the inference mechanism. *In this example we have a clear-cut example of* where an explicit-peripheral subproposition plays a major part in the understanding of the entire conceptualization: one inference memory can make from

(*POSS* X: {(ISA X #PERSON)} Y: {(ISA Y #DIAPER)})

with a high degree of certainty is that the possessor is an infant; namely:

(AGE X #ORDERMONTHS)

(#ORDERMONTHS is a "fuzzy" concept which will match any duration concept within its "fuzzy" limits). The proposition (AGE MC3 #ORDERMONTHS) is therefore added to MC3's occurrence set, and other inferencing proceeds. Eventually, all inferencing will die out or be stopped by depth controls. At that point, MC3 is detected as still having been unestablished, so reference establishment is again undertaken. This time, however, new information is available which resolves the conflict: the AGE predicate is recognized as matching the AGE proposition stored on the occurrence set of MC1. MC3 has thus been identified. Its occurrence set, which has probably been augmented by other inferences, is then merged with that of MC1 to preserve any additional information communicated by the input or its inferences and MC3 is purged. Finally, all subpropositions of the original input are resubmitted to the inferencer in hopes of generating new information by making use of MC1's now-accessible occurrence set. Duplicated information is immediately rejected on this and subsequent passes. This procedure is repeated until no new information turns up. At that point, any unidentified references are communicated externally in the form "X who?" or "what X?"

REFERENCES

1. Fillmore, C. The case for case. *Universals in Linguistic Theory*, E. Bach and R. Harms (eds.), Holt, Rinehart and Winston, New York (1968).
2. Goldman, N. The generation of English sentences from a deep conceptual base. Ph.D. Thesis, Computer Science Dept., Stanford University, Stanford, Calif. (1974).
3. Rieger, C. Conceptual memory. Ph.D. Thesis, Computer Science Dept., Stanford University, Stanford, Calif. (1974).
4. Riesbeck, C. Computer analysis of natural language in context. Ph.D. Thesis, Computer Science Dept., Stanford University, Stanford, Calif. (1974).
5. Sachs, J. Recognition memory for syntactic and semantic aspects of connected discourse. *Perception and Psychophysics* **2** (1967).
6. Schank, R. Conceptual dependency: A Theory of natural language understanding. *Cognitive Psychology* **3** (October 1972).
7. Schank, R. Semantics in conceptual analysis. *Lingua* **30** (1972), 101–140.
8. Schank, R. The fourteen primitive actions and their inferences. Stanford A.I. Memo 183, Computer Science Dept., Stanford University, Stanford, Calif. (1973).
9. Simmons, R. and Slocum, J. Generating English discourse from semantic networks. *Commun. ACM* **15** (October 1972).
10. Wettler, M. Zur Speicherung syntaktischer Merkmale in Langzeitgedächtnis. *Proc. of 28th Congress*, Eckensberger, L. (ed.), DGFP-Hogrefe, Göttingen (in press).
11. Wilks, Y. An artificial intelligence approach to machine translation. *Computer Models of Thought and Language*, R. Schank and K. Colby (eds), Freeman, San Francisco (1973).

Received May 1973; revised September 1973; accepted April 1974