27

Representation and Understanding of Text

Roger Schank

Department of Computer Science
Yale University

INTRODUCTION

How can we get a computer to understand natural language? Our view of the problem has progressed over the years to a point where an answer to that question today would look quite different from one given ten or even five years ago.

Originally, researchers felt that the most relevant issue was syntax. Later, most people agreed that semantics was the most relevant field of study (although few would have agreed on what semantics was). Five years ago, or so, our research was concentrated on finding an adequate meaning representation for sentences. We then began to run into the problems of inference. Even if we could analyze sentences for their meaning, we did not know exactly what we had to do with those meaning structures that we got as output. The answer turned out to be: find out more about what is true about what was said, and tie it together with other things you know or have just heard. In trying to do this we ran into the problem of organizing knowledge of the world. This is about where we are now.

In this paper, we shall run through the issues mentioned above (except for syntax which we never considered very relevant). First we shall discuss the basics of Conceptual Dependency representation. Then we shall discuss the primitive actions which are the basis of any hookup between analysis and inference. We shall then talk about the problems of connecting up causal relations as a way of getting at inference problems. We shall give a glimpse of the MARGIE program and then go on to discuss the theory of scripts and plans and how world knowledge considerations affect representation issues.

CONCEPTUAL DEPENDENCY

The first step in computer understanding of natural language must be the analysis of sentence into a representation of their meaning. We have required of the meaning structures that we use that they be unambiguous representations of the meaning of an input utterance. We have required of those representations that they be unique. That is, the meaning representations of any two utterances which can be said to convey the same meaning should be identical.

Thus, we have concerned ourselves with the creation of conceptual structures, and the predictions and inferences that are possible given a formally defined conceptual structure.

Conceptual Dependency structures are intended to be a language-free unambiguous representation of the meaning of an utterance. Initially, however, the structures we used bore a great deal more similarity to the surface properties of English than we now believe should exist in such structures. Subsequently, we began looking for common concepts that could be used for representing the meaning of English sentences, that would facilitate paraphrase by the conceptual structures without losing information. The concept 'trans' was introduced (Schank, Tesler and Weber, 1970) as a generic concept into which words such as 'give' and 'take' could be mapped, such that by specifying attributed of the cases of 'trans' no information would be lost. (For example, 'trans' where the actor and recipient are the same is realized as the verb 'take', whereas, where the actor and donor part of the recipient case are the same, the verb is 'give'.) Such generic concepts simplified the conceptual networks, making them more useful. Furthermore, it became apparent that the linguist's problem of the representation of such concepts as 'buy' and 'sell' became solvable. Semanticists such as Katz (Katz, 1967) have argued that while these concepts seem close enough, it would be arbitrary to choose one as the basic form of the other, so the correct thing to do must be to write formal rules translating structures using 'buy' into structures using 'sell' when this is deemed necessary. Instead of doing this, we made the suggestion (Schank, 1972) that using 'trans' one could map 'buy' into 'trans money causes trans object' and 'sell' into 'trans object causes trans money.' Such a representation eliminates the 'which is more primitive than the other' problem and instead relates the two events that actually occurred.

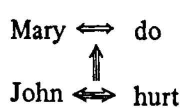
The naturalness of the concept 'trans' led us to consider whether there might be more of these generic concepts around. Thus we began a search for primitive concepts that can be used as the basis of conceptual structures.

We use what is basically an actor-action-object framework that includes cases of the actions. That is, any action that we posit must be an actual action that can be performed on some object by an actor. Nothing else qualifies as an action and thus as a basic ACT primitive. The only actors that are allowed in this schema are animate. Thus, an action is something that is done by an actor to an object. (The exception to this rule regards natural forces and machines which do not really alter our concept of action.)

Actors, actions and objects in our conceptual schema must correspond to real world actors, actions, and objects. To illustrate what is meant by this consider the verb 'hurt' as used in 'Mary hurt John'. To treat this sentence conceptually as (actor: Mary; action: hurt; object: John) violates the rule that conceptual actions must correspond to real world actions. 'Hurt' here is a resultant state of

John. It does not refer to any action that actually occurred, but rather to the result of the action that actually occurred. Furthermore, the action that can be said to have caused this 'hurt' is unknown. In order to represent, in our conceptual structure, an accurate picture of what is going on here the following conceptual relationships must be accounted for: Mary did something; John was hurt; the action caused the resultant state. In conceptual dependency representation, actor-action complexes are indicated by ⟺, denoting a mutual dependency between actor and action; object-state complexes are indicated by ⇔ denoting a predication of an attribute of an object or by ⇐[> denoting a change of state in the object. Causal relationships are indicated by ⇑ between the causer action and the caused action, denoting a temporal dependency. Causal arrows may only exist between two-way dependencies (⟺, ⇔ or ⇐[>). That is to say, only events or states can cause events or states.

Thus our representation for this sentence is:



The dummy 'do' represents an unknown action. ('Hurt' is ambiguous between mental hurt ($hurt_{MENT}$) and physical hurt ($hurt_{PHYS}$).)

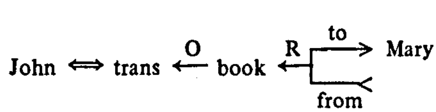
Conceptual dependency representation, then, seeks to depict the actual conceptual relationships that are implicit within a natural language utterance.

Actions, in conceptual dependency, are things that are done to objects. Actions sometimes have directions (either through space or between humans), and always have means (instruments). These things are called the conceptual cases of an action. Unlike syntactic cases, (as posited by Fillmore [Fillmore, 1968] for example) conceptual cases are part of a given action and therefore are always present whenever that action is present. Thus, if an action takes an object, whether or not that object was mentioned it is considered to be present conceptually. If the particular instance of that object was not stated and is not inferrable, then an empty object slot is retained.

The conceptual cases are: OBJECTIVE; RECIPIENT; DIRECTIVE; and INSTRUMENTAL. Using the notion of 'trans' mentioned above we can deal with the sentence:

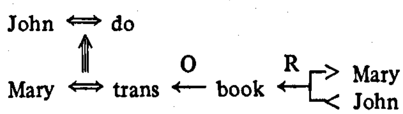
John gave Mary a book.

as follows:



The symbol ←O denotes 'object of the ACT' and the symbol ←R[to / from] denotes 'recipient of the object,' with the recipient of the object in the 'to' part, and 'donor of the object' in the 'from' part.

Actually, this analysis is not quite correct for this sentence since the sentence is conceptually ambiguous. The conceptual diagram above is correct for one sense of the sentence but it is possible that the transition was not done physically by John. Rather, John could have said 'you can have the book' and Mary would have taken it herself. Since we don't know what specifically John may have done we represent this sense as:

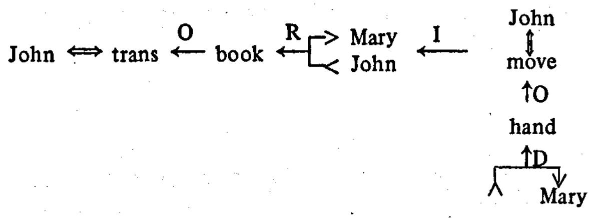


Either of these two structures may have been the intended one, but we assume unless given information to the contrary that the first is correct.

Suppose the sentence had been:

John gave Mary a book by handing it to her.

Here, the sentence is disambiguated by the 'by clause'. All actions require an instrument that is itself another actor-action-object complex (called a conceptualization). When the action in the main conceptualization is known; it is possible to delimit the set of possible instrumental actions. For 'trans' the ACT that is most often the instrument is 'move'. 'Move' represents the physical motion of a bodypart (which may be holding an object) by an actor, together with the direction that that action takes. The conceptual analysis then is:



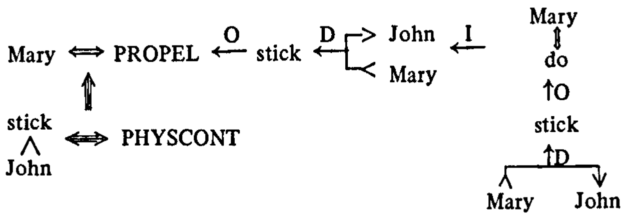
The instrumental case is indicated by $\overset{I}{\leftarrow}$, and the conceptualization that is the instrument is dependent upon (written perpendicular to) the main conceptualization. The directive case (indicated by $\overset{D}{\leftarrow}$) shows the physical direction of the action. Thus 'the book was moved towards Mary'. (It is necessary to indicate here that the hand is holding the book also, but we shall not enter into that here.)

Since every ACT has an instrumental conceptualization that can be said to be part of that ACT, we can see that it should therefore be impossible to ever actually finish conceptually diagramming a given sentence. That is, every ACT has an instrument and so on. In this sentence we might have conceptually something like: "John transed the book to Mary by moving the book towards Mary by moving his hand which contained the book towards Mary by grasping the book by moving his hand moving muscles by thinking about moving his muscles" and so on. Since an analysis of this kind is not particularly useful and is quite bothersome to write, we do not do so. Rather, whenever we represent a conceptualization we only diagram the main conceptualization and such instrumental conceptualizations as might be necessary to illustrate whatever point we are making. It is, however, quite possible that we might need many of these instrumental conceptualizations in a program that was intended to simulate certain body motions such as Winograd's (Winograd, 1971) block moving program. Thus, the ACT in a conceptualization is really the name of a set of actions that it subsumes (and are considered to be a part of it). These instrumental conceptualizations are not causally related since they are not actually separable from each other. In actuality, they express one event and thus are considered to be part of one conceptualization. The rule is then that one conceptualization (which may have many conceptualizations as a part of it) is considered to represent one event.

In ordinary English usage, the syntactic instrument of a given sentence usually corresponds conceptually to the object of a conceptualization that causes the conceptualization most directly related to the verb of which it is an instrument syntactically. Thus as an illustration of this we have:

Mary hit John with a stick.

We represent the conceptual action underlying 'hit' by PROPEL which means to apply a force to an object plus the resultant state PHYSCONT. Thus we have conceptually:



The 'do' in the instrumental conceptualization indicates that the action by which the PROPEL-ing was done is unknown. This corresponds to the fact that this sentence is actually ambiguous, the two most common interpretations being that 'she swung the stick' or that 'she threw the stick'. Representing such a sentence in this manner allows for the discovery of this ambiguity. (In an actual computer analysis schema the blank 'do's' can be realized as predictions about missing information which must be discovered either by inquiry or memory search.)

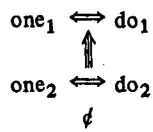
Predictions about what ACT's fit into this instrumental slot are made from the ACT in the main conceptualization. PROPEL requires either 'move' or 'move' + 'ungrasp' as actions for its first instrument. 'Swing' and 'throw' are mapped conceptually into 'move' and 'move' + 'ungrasp' respectively (with additional information as to manner).

It is interesting to consider how such a deep conceptual analysis of natural language utterances can help us in parsing and understanding those utterances. Consider:

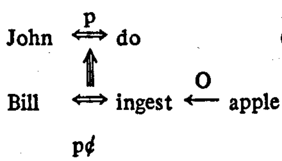
John prevented Bill from eating the apple.

The verb 'prevent' is conceptually a statement about the relationship of two events, namely that one event causes the inability of the occurrence of a second event. Unless we treat 'prevent' in this manner, important paraphrase recognition ability will be lost, and in addition even the ability to intelligently parse sentence derivative from this will be hindered.

Conceptually, then, 'prevent' is not something that anyone can do; rather, it expresses the following relationship between two events.



This is, $person_1$ doing something caused $person_2$ to not be able to (¢) do something else. Thus we have:



(p indicates past tense)

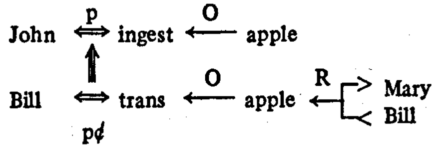
If we had an intelligent understanding system, we might want to know what John 'did' and this representation allows us to realize that we could ask that.

Now consider:

John prevented Bill's giving Mary the apple by eating it.

Along with the information that 'prevent' represents the conceptual structure shown above is a clue as to how to go about finding what might fill in the first 'do'. This clue is that if the ACT that replaces the 'do' is present it is most probably in the syntactic instrument of 'prevent', that is, in a by-clause.

Thus, that clue is used to give us:



It is important to notice that it is quite possible to realize the above structure in the following sentences as well.

> Bill couldn't give Mary the apple because John ate it.
> When John ate the apple it caused Bill to be unable to give it to Mary.
> When John ate the apple, it meant that Mary didn't get the apple.

The above sentences do not use 'prevent' in words but they do use the concept underlying 'prevent'. It is extremely important that any theory of understanding analyze these sentences or any of the myriad other paraphrases into only one conceptual structure in a natural way. This requires establishing the relationships between actual events rather than between the words that may have been used to describe those events. In order to do this, it is necessary to break words down into the primitive actions and events that they describe.

In summary, then, conceptual dependency is a representation for expressing the conceptual relationships that underlie liguistic expressions. The basic structure of this conceptual level is the conceptualization. A conceptualization consists of either an actor-action-object construction or an object-state construction. If an action is present then the cases of that action are always present. One case of an action is instrumental which is itself a conceptualization.

Conceptualizations may be related to other conceptualizations causally. Just as it is impossible to have an action without an actor so it is impossible to have the cause of a conceptualization be anything other than another conceptualization. (This means that 'John moved the table' must be conceptually 'John *did something* which caused the table to be in a different position'. This *doing* is not 'move' but rather something that was unstated. The doing can be inferred and is most probably 'apply a force to'.)

THE PRIMITIVE ACTIONS

Using the framework for language analysis that was just explained the total number of ACTs that are needed to account for any natural language sentence is twelve. In stating this, we are not claiming that this number is totally accurate. Rather, the claim is that the order of magnitude is correct and that these twelve ACTs or some set of ACTs not significantly different than those presented here are all that is necessary to represent the actions underlying natural language.

This result is caused partially by our rewriting a great many verbs into caused states conceptually. Nevertheless it is significant that so few ACTs are actually necessary to account for the basis of human activity.

There are four categories of ACTs that the twelve ACTs are broken down into: Instrumental (2), Physical (5), Mental (3), and Global (2).

Physical ACTs

The Physical ACTs are:

PROPEL
MOVE
INGEST
EXPEL
GRASP

It is our claim that these are the only ACTs that one can perform on a physical object. Furthermore, there are restrictions on what kinds of objects any given ACT will accept.

The meaning of the ACT and the objects are as follows:

PROPEL: means 'apply a force to'; its object must be under a certain size and weight, but for our purposes we will say that any object is acceptable.

MOVE: means 'move a bodypart'; the only objects that are MOVEd (in our sense of MOVE) are bodyparts.

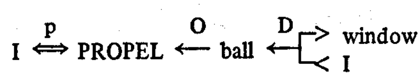
INGEST: means 'take something inside you'; INGEST's object must be smaller than the mouth of the actor or must be divided into pieces smaller than the mouth opening; object should be food.

EXPEL: means 'take something from inside you and force it out;' its object must have previously been INGESTed.

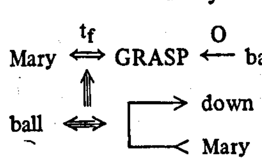
GRASP: means 'to grasp;' object must be within a size limit.

Some example sentences and their analyses are:

I threw the ball at the window.

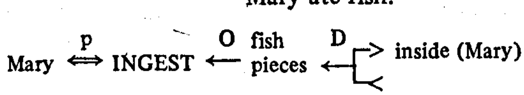


Mary dropped the ball.



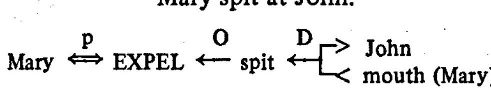
(where t_F means 'the end of' an action)

Mary ate fish.

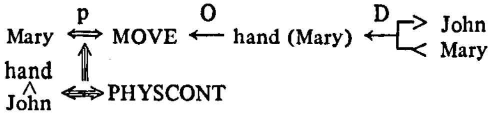


(parentheses denote the POSS relation)

Mary spit at John.



Mary touched John with her hand.

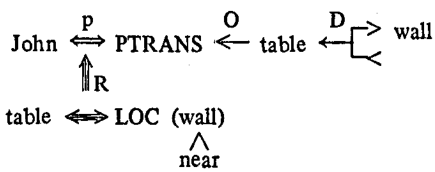


Global ACTs

As can be seen by the nature of the physical ACTs, very often an ACT is somehow more than the sum of its parts. That is, often the result of an ACT is focused on more directly than the ACT itself. Since the representations presented here are intended to represent human thought it is necessary to do the same focussing that humans do. We thus use the notion of Global ACTs which express the change of state consequences and intentions of a variable physical ACT.

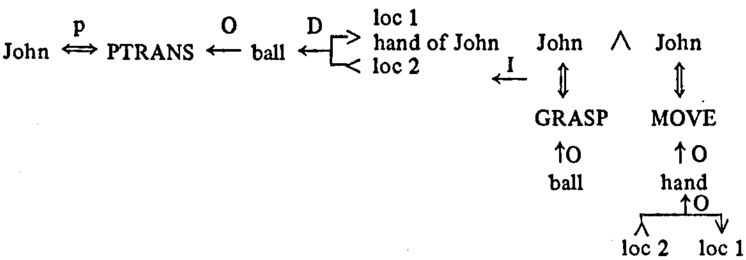
The most important Global ACT is PTRANS. PTRANS expresses the change in physical location of an object. In order to change the physical location of an object it is necessary to perform one of the physical ACTs upon that object first. That is we can have:

John moved the table to the wall.



and

John picked up the ball.



(loc 1 higher than loc 2)

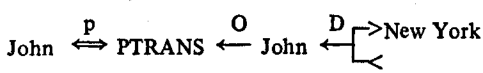
Since PTRANS is of such importance in Conceptual Dependency analysis it is worthwhile to spend some time discussing it. While the use of PTRANS for change of location verbs such as move and pick up is farily straightforward, we also use PTRANS to represent the ACT underlying the verb 'go'. This is a difficult point for speakers of English to accept and thus requires some explanation.

Most semantic analyses deal with 'John went', 'the car went', and 'the plane flew' as if the sentential subject is also the actor or agent semantically. In fact 'John' is the actor in 'John went'. What is important to realize is that 'John' serves a dual role conceptually here. 'John' is also the object of the sentence 'John went'. In saying this we pay careful attention to the problem of inference from a conceptual analysis.

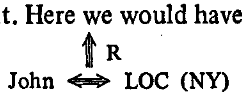
Since the conceptual representations that we are proposing here are used by a computer that is attempting to understand, it is important that the representations be consistent so the programs that operate on them can be general. One generality that we use is that whenever PTRANS is present, it can be inferred that the object of PTRANS is now located at the location present as the directive case for PTRANS.

Thus since it is true that John is the actor when he 'goes', 'John' must be in the actor slot. But, it is additionally the case that the location of John has been changed and that, just as for 'move' and 'pickup', John is now probably located at the directive case location.

Thus the sentence: John went to New York. is conceptually analyzed as:



Actually, this indicates that the direction is towards N.Y. The completed act requires a generated state result. Here we would have:

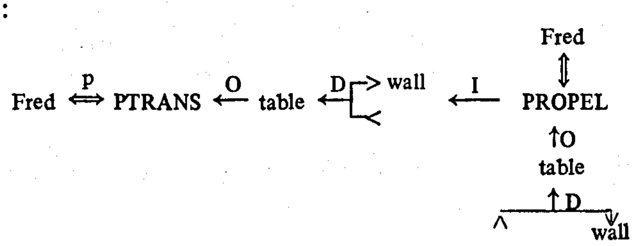


(That is, John is in New York.)

It can be seen that whenever PROPEL is present PTRANS can be inferred. Thus for:

Fred pushed the table to the wall.

we have:



The most abstract of the global ACTs is ATRANS. The objects that ATRANS operates upon are abstract relationships and the physical instruments of ATRANS are rarely specified. The 'trans' that was referred to in the beginning of this paper is what we call ATRANS. ATRANS takes as object the abstract relationship that holds between two real world objects. We have:

Mary gave the book to John.

$$\text{Mary} \overset{p}{\Longleftrightarrow} \text{ATRANS} \overset{O}{\longleftarrow} \text{OWNERSHIP: book} \overset{R}{\longleftarrow} \begin{cases} \to \text{John} \\ \leftarrow \text{Mary} \end{cases}$$

Mary loaned the book to John.

$$\text{Mary} \overset{p}{\Longleftrightarrow} \text{ATRANS} \overset{O}{\longleftarrow} \text{POSSESSION: book} \overset{R}{\longleftarrow} \begin{cases} \to \text{John} \\ \leftarrow \text{Mary} \end{cases}$$

In other words, ATRANS changes one of the parts of a two party abstract relationship. ATRANS can be actually effected in the real world by many means, not all of them physical. The most common instrument for ATRANS is 'MOVE $\overset{O}{\leftarrow}$ hand' where the hand is grasping the object being transferred. Often, however, OWNERSHIP is transferred by signing a paper or by simply saying so. That is, ATRANS can take place and the world can appear exactly as it was to an untrained observer. For this reason, ATRANS is the one ACT presented here that is not necessarily universal. That is, it is possible to conceive of a culture and therefore a language that has no notion of possession and therefore has no ATRANS.

ATRANS operates with a small set of abstract objects. We treat 'sell' as a change in the ownership relations:

Mary sold her car to Beth.

$$\begin{array}{l} \text{Mary} \overset{p}{\Longleftrightarrow} \text{ATRANS} \overset{O}{\longleftarrow} \text{OWNERSHIP: car} \overset{R}{\longleftarrow} \begin{cases} \to \text{Beth} \\ \leftarrow \text{Mary} \end{cases} \\ \Uparrow\Downarrow \\ \text{Beth} \Longleftrightarrow \text{ATRANS} \overset{O}{\longleftarrow} \text{OWNERSHIP: money} \overset{R}{\longleftarrow} \begin{cases} \to \text{Mary} \\ \leftarrow \text{Beth} \end{cases} \end{array}$$

Thus, we are saying that two abstract relationships changed because of some mutual causality. Any physical ACTs that took place (i.e., signing a check and handing it to Mary) are the instruments of the abstract action ATRANS.

We use the verb 'give' in English to denote the change of these abstract relationships. 'John gave the ball to Bill' is a change of possession so ATRANS is used:

$$\text{John} \overset{p}{\Longleftrightarrow} \text{ATRANS} \overset{O}{\longleftarrow} \text{POSSESSION: ball} \overset{R}{\longleftarrow} \begin{cases} \to \text{Bill} \\ \leftarrow \text{John} \end{cases}$$

Another abstract relationship that can be ATRANSed is 'control'. Thus when we say 'John gave his car to Bill', the most likely interpretation is that this is an ATRANS of control rather than ownership. 'ATRANS $\overset{O}{\longleftarrow}$ CONTROL' then, is to 'give the use of.'

John gave his car to Bill.

$$\text{John} \overset{p}{\Longleftrightarrow} \text{ATRANS} \overset{O}{\longleftarrow} \text{CONTROL: car} \overset{R}{\longleftarrow} \begin{cases} \rightarrow \text{Bill} \\ < \text{John} \end{cases}$$

The problem here is that the use of the above primitives makes clear an ambiguity that exists in English that is not otherwise always accounted for in semantic representations. Namely, 'give' can mean a change in possession that required no physical change as in 'Mary gave John the Empire State Building'. 'Give' can also refer to a change in control without a change in possession. Additionally, 'give' can refer to a change in physical location without a change in the abstract notion of possession, as in 'I gave him my handkerchief'. Basically, then, whether 'give' means ATRANS or PTRANS or both is dependent on the nature of the object and is often simply ambiguous. A great deal of the information needed to process languages is based on the thing involved rather than the action.

Instrumental ACTs

There are two instrumental ACTs:

ATTEND
SPEAK

These ACTs are not very interesting in that they are used almost totally as the instruments of some other ACT.

SPEAK is the ACT which actually produces sounds and its objects therefore are always 'sounds'.

ATTEND takes sense organs as objects and physical locations as directions. So 'ATTEND eye' is 'see', 'ATTEND ear' is 'hear', and so on.

Mental ACTs

The three mental ACTs are:

MTRANS
MBUILD
PLAN

Since these ACTs are by no means straighforward, we shall spend some time discussing them.

MTRANS

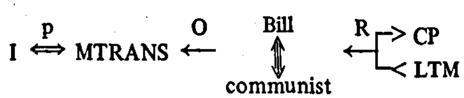
The ACT MTRANS is meant to handle the flow of information to and from the conscious mind. It, plus various mental building acts, should serve to represent all the ways in which we bring thoughts into our heads.

MTRANS represents a change in the mental control of a conceptualization (or conceptualizations) and underlies verbs like recall, commit to memory, perceive, sense, and communicate. It has several features different from the physical PTRANS. For one, the object that is MTRANSed does not leave control of the donor, but is copied into the control of the recipient. Further, the donor and recipient are not always two different people but often two different mental processors (or locations: the distinction in the mind is as fuzzy as the distinction between program and data in the computer), which are frequently within the same person. Five such processors are used:

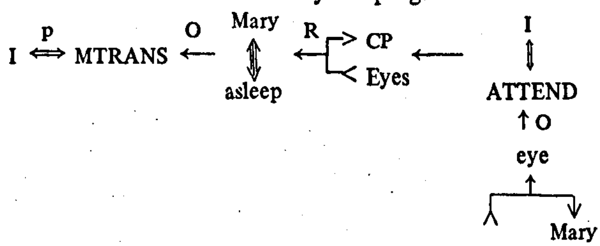
1. Conscious Processor (CP) – is the storage place for whatever is currently being processed.
2. Long Term Memory (LTM – this is primarily the store of beliefs one has about the world. It is a processor too, where such actions as forgetting and subconscious association occur, but the level of activity is both low and hard to characterize, so it shall be treated as a passive element here.
3. Immediate Memory (IM) – this is a storage area for the current context being processed as well as the processing area for conscious thought.
4. Sense-Organs (Eye, Ear, Nose, Tongue, and Skin) – these are all preprocessors, converting raw sense data into conceptualizations describing that data.
5. Body – this covers whatever processors handle internal sensations such as pain, unease, excitement, etc.

With these items, we can handle many mental verbs, such as

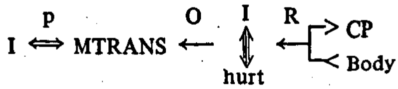
I remembered Bill was a communist.



I saw Mary sleeping.

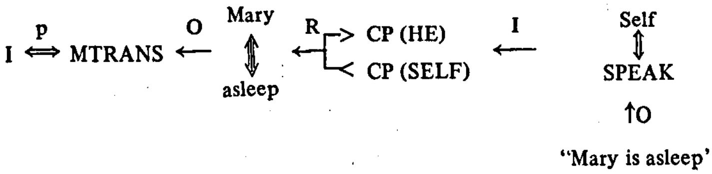


I feel pain.

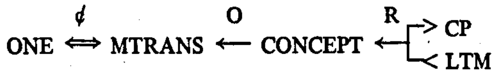


This use of MTRANS covers mental actions where the concept brought into awareness has been internally arrived at, rather than externally generated. Some external MTRANSes are:

I told him Mary was asleep.

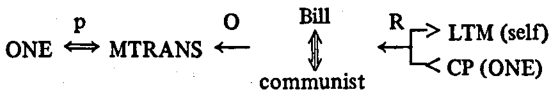


Forgetting is simply the inability to bring something from LTM:



Verbs such as 'learn' and 'teach' also involve MTRANS to LTM from CP. Thus:

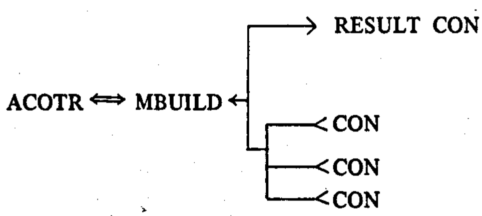
I was taught that Bill was a communist.



That is, 'teach' is really like communicate. The actual difference lies in the fact that the communicated information is said to be new in the case of 'teach'. Thus we also have the information that this information was not in the LTM of I before.

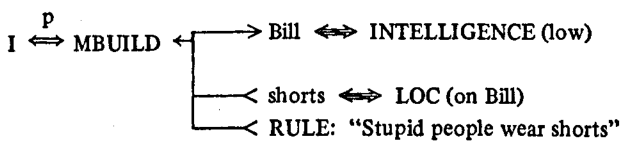
MBUILD

The ACT MBUILD accounts for thought combination. MBUILD is written as:



MBUILD takes as object a many-to-one 'functional' arrow that denotes the combination and transformation of several units into one resultant unit. MBUILD plays the role of the action which is antecedent to some more "final" act of accepting the result as knowledge or as a belief. Examples of this type are 'conclude', 'resolve', 'prove to oneself', 'solve', and so on. In others of these, MBUILD is the only ACT underlying the verb, and there is no result conceptualization yet produced (such as 'think over', 'consider', 'reason out', 'relate', etc.). This distinction between the process and the result of the process (and what becomes of the result afterward) is crucial to the unravelling of mental verbs. MBUILD refers only to the process of combination, or *attempted* combination, and includes no information about the success or failure of the operation. Success can be denoted by the presence of a result in the object slot, and failure by its absence.

I decided that Bill is stupid because he wears shorts.

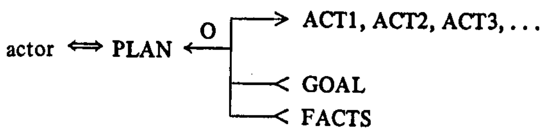


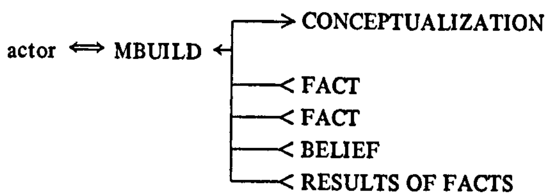
PLAN

PLAN is intended to account for an individual's ability to decide on a step-by-step course of action that leads to a goal. PLAN takes as input a goal and a set of possible plans that are known to be ways to achieve that goal. PLAN decides among them and produces as output a sequential chain of conceptualizations that the actor intends to act out in order to achieve the desired goal. PLAN takes MBUILD as its instrument where the items MBUILDed are the particular decisions that make up the selection of a given plan.

The general form of PLAN is with an object arrow that takes as input a goal and the plans associated with that goal and produces as output the chain of acts that the actor intends to perform to achieve the goal.

The general form of MBUILD is with an object arrow that takes multiple inputs dealing with the current facts and the logical consequences and beliefs associated with those facts and produces a new thought which may or may not be an intended future action.





The difference in these two ACTs with respect to their realization in English is as follows:

John decided to go to the Bahamas.	MBUILD new thought
John developed a plan to go to the Bahamas.	PLAN unknown set of ACTs. Goal that is input to ACT PLAN is known.
John knows how to get the money.	PLAN with known goal.
John figured out who the murderer is.	MBUILD new thought
John did that to get Mary to like him.	PLAN
John intended to hurt Mary.	PLAN (ambiguous if no ACT was done)
John intends to hurt Mary.	Ambiguous (Not known if he has figured out how)

INFERENCES

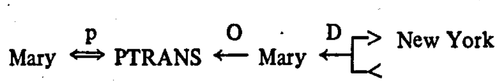
It should be clear that any attempt of this kind to put sentences into underlying representations that use only a few primitive ACTs must have as its intent the use of these ACTs in some prescribed fashion. Each ACT is basically a memory affector, in that whenever that ACT is present certain facts can be inferred from it.

In considering the problem of how to know when something would qualify as a new ACT, the pertinent question to ask is whether the inferences that would be drawn from that ACT are the same as the set of inferences that are drawn from some already existing ACT.

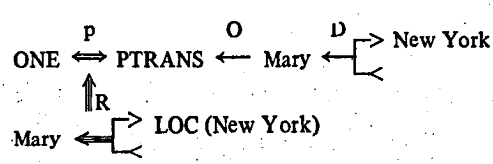
Here it is important to make clear what exactly we mean by an inference. For our purposes, an inference is a conceptualization that is true to some degree of probability whenever some other conceptualization or set of conceptualizations are true. For example, in the sentence

Mary went to New York.

it is not explicitly stated that Mary in fact arrived in New York. The sentence is graphed as:



while 'Mary arrived in New York' is graphed as:



that is, we don't know if she actually got to New York. We know only that she went in that direction. We infer that if we are told something and not explicitly told that the expected inference is invalid, then it is reasonable to draw that inference. In this case PTRANS causes the location inference to be generated in absence of information to the contrary.

Essentially, all the Conceptual Dependency is oriented to how the representations will best be used. They will be used in memory for inferencing. We conceive of the language understanding process as a set of procedures that seek to extract what is explicit in an input and attach it to the body of knowledge associated with that explicit information in a way that will allow the system to formulate questions that will fill in the "holes" that exist in the final meaning representation of that input. People understand more from a sentence than they have been told explicitly. They also understand what they have not been told and what they must infer in order to make sense of what they have been told. Knowing what you don't know allows for the processes to be generated that will fill in the empty holes.

What we are saying is that human communication is based largely on what is left out of a discourse. People rarely specify everything they intend to communicate. Rather, they specify enough to lead the hearer to an understanding of what was meant. They leave out anything that they assume the hearer can figure out for himself. It is easy to see that the problem for an understander is the recreation of what has been left out.

We call the propositions that have been left out *inferences*. Simply defined, an inference is anything that is likely to be true about a given input but is not necessarily true. A rule of thumb that can be used in determining what is and is not an inference, we call the *but test*. A proposition is a reasonable inference from another proposition (in English) if the negation of that proposition preceded by the word 'but' in conjunction with the first sentence yields a sensible sentence. If the result of this 'butting' yields a sentence that denies its own premise, then the second proposition is said to be *implicit* within the first. If the result of the conjunction is nonsense because there is no obvious relation, then the second proposition is not an inference from the first. Some examples of this are:

John punched Mary but his hand didn't touch her.	*implicit*
John punched Mary but she didn't cry.	*valid inference*
John punched Mary but it didn't rain.	*invalid inference*

The question of understanding sentences like the last one here is not to be dismissed by marking it invalid. If somebody said the last sentence we would want to know why he expected that such a consequence would result. (For the purposes of the test we simply show that raining is not usually to be inferred from punching.)

Inferences, then, are never certain. They come from rules that a hearer learns to apply to various words, concepts, and situations. Inference is the core of the understanding process and thus inferences lie at the center of human communication. They serve to tie together inputs into a related whole. Often the inferences themselves are the main point of the message.

If inferences are such a crucial part of understanding, we might wonder why they are left out of spoken language. The answer is simply one of space. People try to be as concise as possible and avoid putting in remarks that are obvious to the hearer. This process involves assessing what the hearer knows while talking to him. This assessment is not always made well, and if what has been left out is important and must be inferred, a misunderstanding can result. Inferences are also left out because the speaker doesn't want to have been said to have said them. Politicians make use of this inferential capability of hearers all the time.

The process of inference is wholly dependent on an adequate supply of world knowledge. Without the knowledge about what can happen in the world, it is impossible to make sense out of what has happened. Small children are capable of witnessing earth-shaking events without making sense of them because of their lack of knowledge about the implications of these events. That is, they didn't know what they were seeing.

Rieger (Rieger, 1975) has classified the process of inference into sixteen distinct inference classes. It is his thesis that every input sentence is subjected to the mechanisms that are linked to these classes to produce inferences every time a sentence is received. Below are Rieger's classes of inferences:

1. Specification – What parts of the meaning underlying a sentence were implicit and must be filled in?
2. Causative – What caused the action or state in the sentence to come about?
3. Resultative – What are the likely results of an input action or state in terms of its effect on the world?
4. Motivational – Why did the actor perform the action? What did he intend to happen?
5. Enablement – What states of the world must have been true for the actor to perform his action?
6. Function – What is the value or use of a given object?
7. Enablement/prediction – If a person wants a state of the world to exist, what action will it then be possible to perform?
8. Missing enablement – If a person can't do what he wants, what state will have to change in order to permit it?
9. Intervention – What can an actor do to prevent an undesirable state from occurring?
10. Action/prediction – Knowing a person's needs and desires, what actions is he likely to perform?
11. Knowledge propagation – Knowing that a person knows certain things, what else is he likely to know?
12. Normative – What things that are normal in the world should be assumed in the absence of being told them specifically?
13. State-duration – How long will a given state or action last?

14. Feature – What can be predicted about an entity when a set of facts is known about it?
15. Situation – What other information can be assumed about a given situation?
16. Utterance-intent – Why did the speaker say what he said?

It is memory's responsibility to establish the inferences that are likely to be true in a given situation. These inferences are made partially because it is often the inferences themselves that are the point of the input. Inferences are also made because they will lead to questions and knowledge that will lead to the understanding of a sequence of sentences as a whole.

The basic mechanism in understanding is the inference process. People understand more than what they have explicitly been told. They do this by tying together what they have heard with what they know about what they have heard. We can illustrate this interaction of knowledge and inference in understanding by considering the following sentence:

John saw a large boulder rolling down the mountain towards his house.

followed by each of these sentences as the second sentence in a two-sentence paragraph:

1. He ran inside to get his pet mouse.
2. He started digging a path that led away from his house.
3. He hid under his bed.
4. He picked up the telephone.
5. He started to cry.
6. He ran inside to get a mattress.
7. He yelled for his wife.
8. He called the geological society.
9. He started to write his will.

The process of understanding is considerably more than just deciding upon the appropriate meaning for a given individual sentence. The meaning of a paragraph is more than the sum of the meaning of each individual sentence contained in it. People leave out from a story or sequence of sentences the reasons for each action in the sequence and the way in which the individual items in the sequence relate to each other. These things are left out largely because they are obvious. We could ask a person who has just heard any of the nine sequences above why the action communicated in the second sentence was done and they would probably give answers of the following sort:

1r. Because he wanted to save the mouse's life.
2r. To cause the boulder to veer away from the house.
3r. He was frightened and irrational.

4r. He intended to call for help.
5r. He was frightened and gave up hope of saving himself.
6r. Maybe he though he could cushion the blow of the boulder by putting the mattress in front of the impact point.
7r. He wanted to make his wife aware of the problem so she could leave the house and save herself.
8r. Perhaps he thought the rock was an interesting specimen and he wasn't aware of the danger.
9r. He assumed he was going to die and wanted his possessions dispersed according to his wishes.

These responses indicate an understanding of the situation being discussed apart from and in addition to what has actually been related. Within the process of understanding, memory is responsible for finding the knowledge that it has that is relevant to what it has heard and using that knowledge to make inferences about the intentions, motivations, and effects of the actions in the sequence that it has heard.

What is obvious to a person is of course not obvious to a computer. The problem of getting a computer to understand is at least partially the problem of finding out what information is necessary when. In order to understand the sequence above using sentence (1) as the second sentence, it is necessary to access the facts that:

a. Large objects going at a sufficient rate of speed can destroy objects of insufficient rigidity.
b. A boulder rolling down a mountain is an instance of the first part of fact (a) and a house is an instance of the second part.
c. Things inside objects about to be destroyed often get destroyed as well.
d. People try to prevent actions that have bad consequences for them.
e. People value their possessions and dislike losing them.
f. Taking something away from a situation that will destroy it will prevent that destruction.
g. Pets are valued possessions.

A combination of these seven facts will help to process this two-line sequence. Only facts (a) and (b) are used in every sequence. The other eight sequences require facts of their own to make sense out of them. Thus, it is obvious that part of the problem of understanding is organizing the knowledge that one has so that it is possible to find what is needed in a given situation. A model of the understanding process must necessarily be extremely complicated. The base of any solution must contain a memory for facts that will apply as inferences in a given situation.

Once natural language sentences can be reduced to the conceptualizations underlying them with the use of primitive actions, the inference process is simplified. We are guaranteed to have activated all parts of the semantic equivalence class if any of its members is activated. The problem of inference is by no means completed by the use of these primitives. What we have done is to reduce the number of inferences that need be stored by rewriting, so to speak, the verb into an ACT from which we can draw inferences. Certain inferences are simply not taken care of by this. For example, if we have 'Mary kissed John', our mapping of kiss into 'MOVE lips towards' will not simplify the problem one bit (most inferences fall into this category, in fact). One must be careful not to lose information in doing a conceptual analysis. (That is, 'kiss' is really more than just 'MOVE lips towards'.) However, the mapping of the various verbs into ATRANS, for example, eliminates the problem of having to make the same inference over and over again.

The value of these primitive ACTs is that certain things are true whenever a given ACT is present and thus large amounts of information that is true for a given verb can be written only once for that underlying ACT. These equivalence classes, then, are probably much more like what people learn than would be an exhaustive list of what is true for every verb.

In addition, verb paraphrasing is explained by the use of these primitives. The core of the MARGIE program for paraphrase and inference was, of course, the notion of primitive ACTs.

MARGIE

The MARGIE program was written at Stanford to demonstrate the parser, inference and generation programs that were written based on Conceptual Dependency Theory. The MARGIE (Memory, Analysis, Response Generation, and Inference on English) computer program was designed and built at the Stanford Artificial Intelligence Project as a first attempt at dealing with the problems of handling meaning and inference. The program had three separate pieces. The conceptual analyzer (Riesbeck, 1975) took a subset of English strings and mapped them into a deep conceptual representation of their meaning. The memory and inference program (Rieger, 1975) stored new inputs, established references, and made inferences about the input it had just received. The third piece was the generator (Goldman, 1975) which encoded conceptual representation into syntactic structures. A modified version of a program written by Simmons and Slocum (Simmons and Slocum, 1972) was then used to map the syntactic structures into English sentences.

MARGIE had two modes, PARAPHRASE and INFERENCE. In PARAPHRASE mode, sentences were analyzed into a deep representation of their meaning and then that meaning was "read out" in as many ways as were available to the generator for expressing that meaning in English. Since the deep representation used constructs that were not part of English, the problem was one of coding ideas into language, rather than that of transforming input sentences in a variety of ways. In fact, the actual input sentence was not retained in the paraphrase program after its meaning had been ascertained.

In INFERENCE mode, input sentences were analyzed into their meaning representation and then operated on to find out what other information might have been true or arisen as a consequence of the input. Inferences were established and used as the basis for forming still more inferences. These inferences were then encoded into English and output.

MARGIE took up about 75K of core in its compiled version. It ran on the PDP 6/10 at Stanford. The response time was a few seconds in PARAPHRASE mode and a few minutes in INFERENCE mode.

The significance of MARGIE lay in its possibilities for expansion rather than any present-day technological advance. It was limited to somewhat stilted sentences and its vocabulary and world knowledge were very small. We have presented this work because the basic components of MARGIE are likely to be the basic components of any future computer understanding system, and it is in this sense that this work is best understood. Paraphrasing and inferencing as we had defined them in MARGIE are non-tasks. That is, they are not things that people do, nor that are important for machines to do. However, the individual pieces of MARGIE were things that are important to model. That is, meaning analysis, inferencing, and generation are important sub-tasks on the road to computer understanding.

Examples

It is helpful, in order to ascertain the scope of the program, to look at some examples of what it could do. The examples presented here were all run on the Stanford computer. The sentences presented to MARGIE were in no sense random in that the user was of course aware of the vocabulary of the program.

INFERENCE MODE

INPUT: John told Mary that Bill wants a book.
OUTPUT1: A book about what?
OUTPUT2: Mary knows that Bill wants a book.
OUTPUT3: Bill wants to come to have a book.
OUTPUT4: Bill wants someone to cease to have a book.
OUTPUT5: Bill wants to read a book.

INPUT: John gave Mary a beating with a stick.
OUTPUT1: A stick touched Mary.
OUTPUT2: Mary became hurt.
OUTPUT3: John wanted Mary to become hurt.
OUTPUT4: What did Mary do to make John angry?

INPUT: John gave Mary an aspirin.
OUTPUT1: John believes that Mary wants an aspirin.

OUTPUT2: Mary is sick.
OUTPUT3: Mary wants to feel better.
OUTPUT4: Mary will ingest the aspirin.

INPUT: John is going to go to the store.
OUTPUT1: John wants to be at the store.
OUTPUT2: John wants to buy something.
OUTPUT3: John will trade money to the store for something.

PARAPHRASE MODE

INPUT: John advised Mary to drink the wine.
OUTPUT: John told Mary that drinking the wine would benefit her.

INPUT: John killed Mary by choking Mary.
OUTPUT1: John strangled Mary.
OUTPUT2: John choked Mary and she died because she could not breathe.
OUTPUT3: Mary died because she was unable to inhale some air and she was unable to inhale some air because John grabbed her neck.

INPUT: Reading the book reminded Rita to loan the book to Bill.
OUTPUT: Rita remembered to give the book to Bill and she expects him to return it to her because someone read the book.

INPUT: John prevented Bill from giving a banana to Mary by selling a banana to Rita.
OUTPUT1: Bill was unable to give a banana to Mary because Rita traded John some money for a banana.
OUTPUT2: Because Rita bought a banana from John Mary could not get a banana from Bill.

CAUSALITY

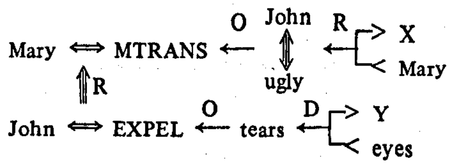
After MARGIE was finished, we began to work on the problems of the representation of text. It was quite unreasonable to be working on individual sentences as an end in themselves. The first problem we encountered was that of causality.

Causality problems are, of course, present in individual sentences as well, so as a first step in processing text, we went right back to dealing with individual sentences. The real problem is connecting up of ideas and events.

Consider the following sentence:

John cried because Mary said he was ugly.

It is possible to consider this sentence as having underlying it the conceptualizations 'John cried' and 'Mary said John was ugly'. We have treated this in Conceptual Dependency as an example of what we had called 'reason causation' (denoted by ⇑R). Thus, we would have treated the above sentence as:



A very simple and obvous thing that is incomplete about this representation is that we must assume that John heard this statement of Mary's, either from Mary or from someone else, yet this is not explicitly stated.

So the first inference that must be made is that 'John is ugly' was MTRANSed to John. What else do we know about this sentence? We know that something must have gone on in Johns's mind concerning Mary's statement. We can guess that the statement must have caused him to think about the fact that he was ugly and thus made him unhappy or perhaps that Mary didn't like him and he wanted her to, and this made him unhappy. Thus, though the above analysis is superficially acceptable, conceptually it must have been something like:

> 'Mary MTRANSed, caused John know what
> Mary MTRANSed, caused John think about some set
> of facts, caused John be unhappy, caused John EXPEL.'

Two problems come to light considering this new analysis. One is that these causals are quite different from each other. The fact that the English word 'cause' can be used here helps to cover up the problem. As will be shown later, these conceptualizations affect each other quite differently. It is necessary, then, to delimit the types of causality that can be used conceptually.

The next problem is more philosophical. Do we really need to do such an extensive analysis? What does it actually buy us? We claim that it is almost always necessary in a complex understanding task to go as deeply as possible in order to provide some universal base that is language-free with which to work. Clearly, for certain tasks, the above sentence need not be broken down any further than we have done here. However, consider the following sentence (discussed in detail with respect to a computer program that handles it in (Rieger, 1975)):

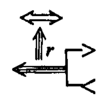
> Mary kissed John because he hit Bill.

A program that failed to notice that Mary was probably pleased by Bill's getting hit would have failed to understand what is one of the most interesting facts in that sentence. In order to be able to make such an inference, it is necessary to break up the surface causal relation into one that is more reflective of the causal connections of ideas. With this introduction we shall now proceed to the syntax of causality.

Causal types

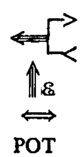
We allow four kinds of causality each occurring in the following conceptual syntactic arrangement:

Result Causation



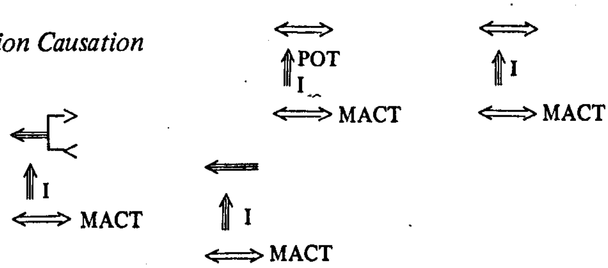
A conceptualization involving an ACT (denoted $\Longleftrightarrow$) can have a *result* ($\Uparrow r$) that is a change in value of some state of an object.

Enable Causation



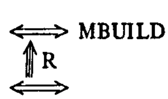
When a state change occurs, it can possibly complete sufficient conditions for an ACT to take place. This is called *enable causation* ($\Uparrow$&). Such state changes only enable potential ACTs ($\underset{POT}{\Longleftrightarrow}$). A potential ACT is an abstraction that can be realized as an ACT if the actor so desires.

Initiation Causation



Whenever an ACT or a statechange occurs, or whenever a state of a POT-ACT exists, it is possible that an actor may be made aware of it, and made to think about it. This is called *initiation causation* ($\Uparrow$I). Initiation causation accounts for people thinking about things.

Reason Causation



Once people have started thinking about things, they are likely to decide to do something (MBUILD). Deciding to do something is the *reason* for doing it. This is *reason causation* ($\Uparrow$R).

Lead-to causation: One event can be said to *lead-to* another event. This is solely an abbreviation for one of the above causals. It is not always important to know how to expand a causal so we must provide some means of representing this conceptually: (⇑L).

Causals causing: Certain causals can themselves cause or be caused.

Negatives causing: A negated conceptualization cannot really cause anything. However, a non-event can initiate a thought, as can anything else, so we allow negatives to cause MBUILDs in Initiation Causation.

A classification of causal relations according to conceptual syntactic type is desirable for both psychological and computational reasons. Certainly it is true that since the different causal arrows postulated related different combinations of conceptualizations, then we would logically consider that there is only one causal arrow. It has been our philosophy to try to represent differences that seem psychologically plausible without regard to their most logical representation. We feel that the ease of psychological (as opposed to logical) inferencing which such a method provides is important.

The causal types above can be considered to be different relational arrows between conceptualizations. We could treat these essentially syntactic distinctions as conceptual semantic distinctions. That is, we could allow only one causal arrow, with complicated rules relating what must be on one side of the arrow to what was on the other side. From the point of view of prediction in analysis such an approach would be computationally less efficient. Certain words in English (and other languages) refer to only certain of the causal arrows and not others ('allow' and 'prevent' refer to "enable causation," for example). Knowing the conceptual syntactic type of a causal (from the input word) thus delimits the possible semantics. Information predicted on the basis of the presence of one causal type about what other causals would be present allows for a better top down search in the analysis. Thus, if we know that two conceptualizations have been related by an English 'cause' word, then the syntax of causality predicts what type of conceptualization must intervene between them in a causal chain. This enables us to search for them either by looking in the sentence or by asking questions.

Such a causal classification makes inference simpler as well. Given X causes Y, where the causal type has been determined, certain inferences are valid for one causal and not for others. For example, if enable causation has been determined on conceptual syntactic ground, it can be inferred that the other conditions that contributed to the final sufficient condition in the causal were also present. This can be done directly from the syntactic type without having to first infer that the most recent state change completed the sufficiency conditions.

Other causal types

The above causal relations can have certain changes made to them that cause them to behave differently. Any causal can be stated as being potential. This is symbolized by a 'c' on the causing conceptualization. A (c)⇐B is considered to be 'if A then B would occur'.

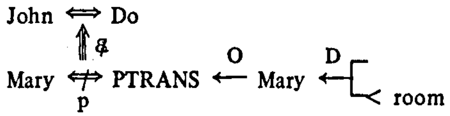
In addition to this, enable causation can be negated (symbolized ⇑̸). This means that a state change can remove the sufficient conditions for an ACT to take place. This, in effect, is another kind of causation (unenable causation); but since all the conditions and properties of enable causation are the same as unenable causation, we will treat them as being the same here.

It is not possible to place time modifications on causals themselves because causals are merely relations between conceptualizations. The conceptualizations denote events which themselves have times. Thus event 1 can be causally related to event 2 even if event 2 is a future event, and event 1 is a past event. In that case, the causality is hypothetical (will cause).

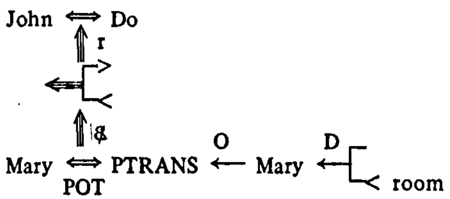
Use

Using this syntax, then, certain causal relations can be shown to need further explication. Thus, above we treated 'prevent' as:

John prevented Mary from leaving the room.



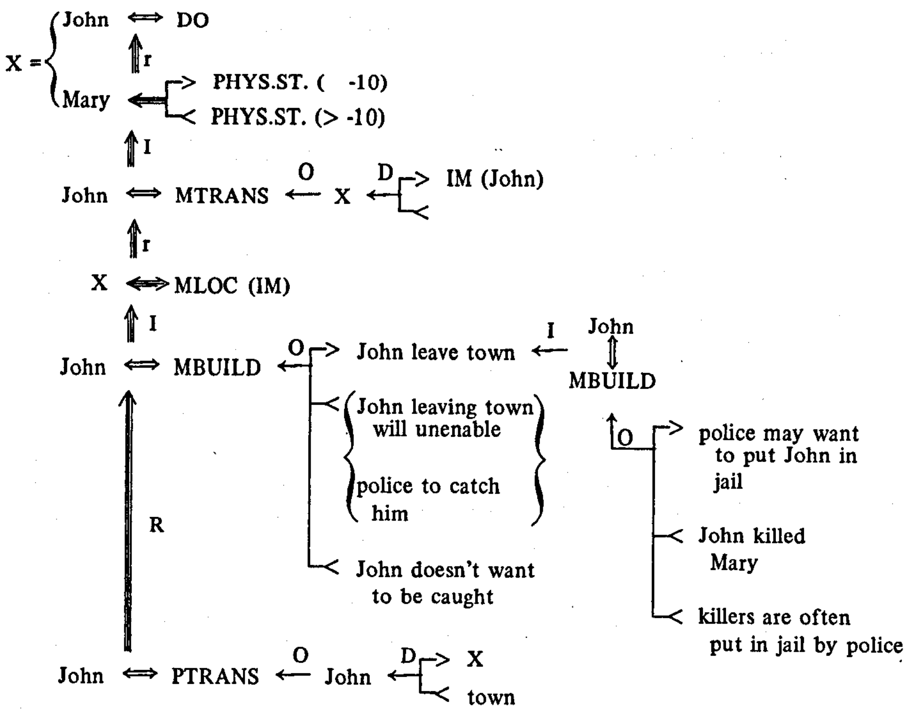
Now it can be seen that since conceptualizations of action (⟺) cannot cause other (⟺'s) (with the exception of Mental ACTs), this conceptual diagram must be further expanded. The actual diagram should be:



That is, it is the result of the ACT that John did that served to end the sufficient conditions for Mary's leaving the room. (For example, he might have broken her legs. Her non-movable legs would have ended the sufficient conditions for her leaving the room.)

John left town because he killed Mary.

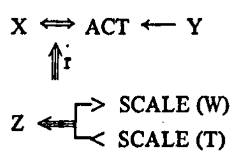
would be diagrammed as:



This example may seem fanciful, but we would claim that something like it is understood by hearers of the sentence and the mechanisms for creating such a conceptual structure are an integral part of an understanding system.

The most important part of the delimitation of how causals can connect rests with the semantics of causation. Thus, it is necessary to list every ACT and state combination that is permissible. By doing this, we are invoking a world model that in no sense can be said to be universal. Different cultures perceive different reasons for why things happen. As much as possible, then, these semantics are intended to indicate Western culture at best or my own personal culture at least.

Result Causation



a) The state change occurs in the object only (Y = Z) with the following exceptions:
   1) If the ACT is MOVE or GRASP the actor (X) can be Z.
   2) If the ACT is INGEST the actor (X) can be Z.

b) The following ACTs are the only ACTs that can cause a state change. The SCALES that can be changed are listed for each ACT.

PTRANS	LOCATION	(LOC)
	CONTAINMENT	(CONT)
ATRANS	POSSESSION	(POSS)
	OWNERSHIP	(OWN)
	CONTROL	(CTL)
MTRANS	MENTAL LOC	(MLOC)
PROPEL	PHYSICAL STATE	(PHST)
INGEST	PHST, HEALTH	
GRASP	PHST	
MBUILD	MLOC	

Examples where only result causation is present are:

John hit Mary $\xrightarrow{\text{result}}$ Mary is hurt.
Mary went to New York ⟶ Mary is in New York.
John dropped the glass ⟶ The glass is broken.

Mary kissed John ⟶ Mary is happy.

Enable Causation

⟺
⇑ &
⟺

The state that enables an action must, of course, relate to that action. Thus, only the following states may enable the ACTs listed with them.

LOC	PTRANS, PROPEL, GRASP, MOVE, EXPEL, INGEST
POSS	ATRANS
CONTROL	ATRANS
OWN	ATRANS
PHYS. ST.	any ACT
HEALTH	any ACT
MLOC	MTRANS, CONC, MBUILD
ANGER	PROPEL

Enable causation is involed in the following examples:

He picked up the ball and then he threw it.
Mary told John what she had read in her book.
John found a ring and he gave it to Mary.

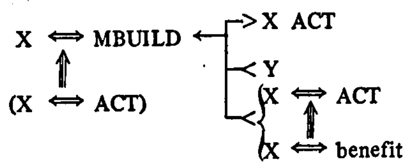
Initiation Casuation

⟺ (⟺)
⇑ I
⟺ ACT

Since it is possible for any event or state to cause someone to think of just about anything, there really exists no semantics for initiation causation, except the fact that the only ACTs that can be initiated are MTRANS and MBUILD. It should be possible though to trace a path from the initiating event or state to the new thought. Examples:

John reminded me of Bill.
A falling apple was responsible for Newton's Gravitation Law.

Reason Causation



The semantics of initiation causation are severely restricted since only MBUILD can serve as the ACT in the causing event. The only other restriction is that the causal event must have been present in the output object of MBUILD and that some set of conceptualizations (Y) must have been used in the MBUILDing.

Included within Y must be either a REASON or a BELIEF. A REASON is a conceptualization of the form 'One should do Z if W'. A reason must be present if an ACT is caused.

If a state change is caused, a BELIEF must be present within Y. A BELIEF is of the form 'Z is bad' or 'What is bad for W is bad for X'. Examples:

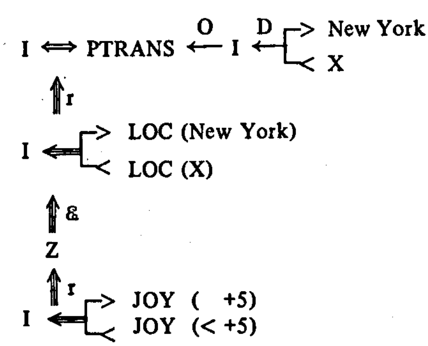
Mary hit John because she dislikes him.
John was unhappy because Mary died.

What all this means

Suppose we had the following sentences:

I am happy because I went to New York.
John's cold improved because I gave him an apple.
John died because Mary gave him a book.

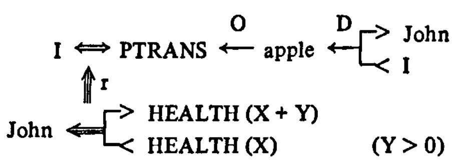
The conceptual representations of these sentences will now be more than the simple connecting of the stated events. For the first we have:



That is, PTRANS cannot change results having to do with JOY. Rather, only LOC can be affected. It must be inferred therefore, that the new location enabled the other ACT to take place which could affect the JOY scale. The only ACT which can affect the JOY scale (JOY is a MENT-ST) is MBUILD. Thus, we can infer that some other ACT initiated the MBUILD.

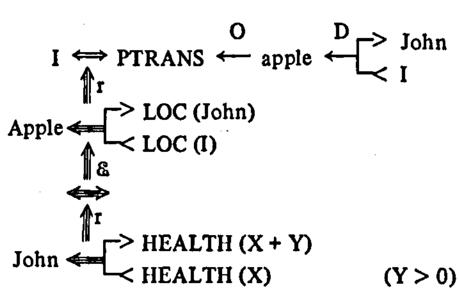
In the 'apple' sentence we are faced with a sentence that again 'really' means something different from its surface meaning. In order to interpret it we shall make use of the distinction between conceptual syntax and conceptual semantics.

It is important to realize that what has been presented so far is really idiosyncratic. That is not to say that many speakers may not share the particular world view adopted here, but it is clear that it is not necessary to do so in order to speak. The conceptual semantics is the idiosyncratic part rather than the conceptual syntax. Thus, the syntax says that PTRANS is an ACT and ACTs can have results. Thus, according to the conceptual syntax only, it would be possible to diagram this sentence as:

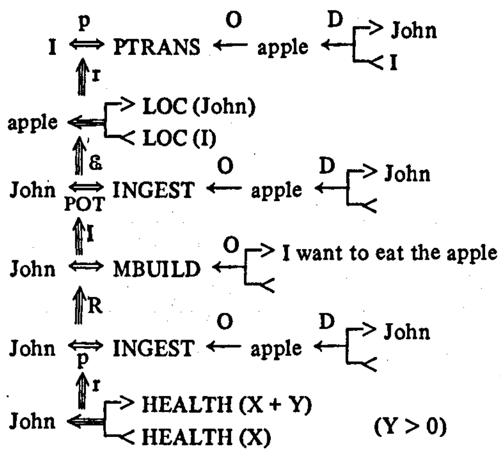


But, in fact, our conceptual semantics of causality explicitly disallows such a result causation. PTRANS can only affect LOCATION and CONTAINMENT in our particular world. It is certainly possible that there exist worlds where the mere possession or nearness of something can affect health, but in order to understand a person it is usual practice to impose our own world view on what he says. Thus, our analysis of this sentence does not permit PTRANS to result in HEALTH changes. Rather, the effect of a PTRANS is to change LOCATION, which can then enable certain ACTs to occur.

Now, then, the final parse of this sentence would be:



It is now possible to pass such an analysis to an inference program (such as that of Rieger [Rieger, 1975]) that would attempt to make educated guesses about what would fill in the blanks. Since there is a result causal path from INGEST to HEALTH according to the semantics, and since there is a path along the enable causal path from LOC to INGEST, we can now create a more accurate picture of what was *probably* meant:



For the third sentence it can be seen that part of understanding this sentence is being able to react to it in a sort of 'how come' fastion. That is, a hearer would want to know why the first thing should possibly cause the second. We might be able to trace this from the syntax and semantics of causals, but then we might not. Our proposal is that the syntax and semantics of causals should be used to set up as much as can be figured out about a situation, and then to infer the rest. At some point this all must stop, however. Thus, we allow for a simplification that includes an unspecified causal (⇑L) for one thing leading to another in a yet undetermined manner. This lead-to causation is purely bogus and is used only when something better cannot be gotten. It should be clear that in most of our previous use of causals in other papers and in most everyone else's, 'lead to' causation is what is being used. We wish to make clear that this is an abbreviation and should only be used as such.

It is also true that we shall need more abbreviations for the complications arising from trying to write these complicated diagrams and from trying to use them in computer programs. Thus, in the above example, it would be simplest to write ⇑I R r for the causal between the POT-ACT INGEST and the result of the real ACT INGEST (HEALTH change). This abbreviation is always expandable and is just a shorthand.

Thus, we put ourselves in the ambivalent situation of creating a strict syntax for conceptual representation of causality and then only sometimes using it. For a practical usage of causality chains, it is necessary to create abbreviations but to know that these are abbreviations and thus to have the apparatus available to expand them when this is desired. We use the following for example:

ER – enable plus reason causation. This denotes that a POT-ACT was thought about and did occur.

rE – result plus enable causation. This denotes that some state was resulted which enabled some new ACT.

IR – mental causation plus reason causation. This indicates that an ACT was thought about and used as a reason for another ACT.

SCRIPTS

We view the process of understanding as the fitting in of new information into a previously organized view of the world. Input sentences (like input words in intra-sentence analysis) set up expectations about what is likely to follow in the text. These expectations characterize the world knowledge that pertains to a given situation.

The concept of a script, as we shall use it here, is a structure that is made up of slots and requirements on what can fill those slots. The structure is an interconnected whole, and what is in one slot affects what can be in another. The entire structure is a unit that describes a situation as a whole and makes sense to the user of that script, in this case the language understander.

Whenever a story is understood, inferences must be made that will connect each input conceptualization to those that relate to it in the story. This connecting process is difficult and dependent upon the making of inferences to tie together seemingly unrelated pieces of text. However, it is a process that can be facilitated tremendously by the use of scripts.

A script is a giant causal chain of conceptualizations that have been known to occur in that order many times before. Scripts can be called up from memory by various words in the correct context, by visual inputs, or by expectations generated by inferences. What a script does is to set up expectations about events that are likely to follow in a given situation. These events can be predicted because they have occurred in precisely this fashion before. Scripts are associated then with static everyday events such as restaurants, birthday parties, classrooms, bus riding, theater going, and so on. A simple test for when a script is helpful we call the *reference test*. The reference test consists of trying to introduce a new object by use of 'the' and seeing whether the text makes sense. Some examples:

John went to a restaurant.
He asked the waitress to tell the chef to cook him a hot dog.

John went to a restaurant.
He asked the bus driver to talk to the midget.

John went to a birthday party.
First Bill opened the presents and then they ate the cake.

John went to a birthday party.
He asked the waitress to tell the chef to cook him a hot dog.

Every 'the' in the first and third paragraphs makes perfect sense because a script has been introduced and the objects prefaced by 'the' have all been implicitly referenced by the script. In English it is usually incorrect to preface an object by 'the' unless it has been referred to before. It is all right here because the script effectively 'says' that these items were present before they were actually mentioned. When these rules are violated, as they were in the second and fourth paragraphs, it makes the hearer uneasy and he is forced to fill in the missing sentence that introduces these objects by himself. Thus in the fourth paragraph we are surprised that a waitress and chef were present at the party, but we can cope with this information by augmenting our 'birthday party' script to be a 'very fancy party' script.

We are saying that new inputs are handled by hearers in terms of the old knowledge that is lying around about the words and participants in a situation. More important, we are saying that we would expect that, since a script is called up to interpret new inputs and the script contains 'blank events' (events that are expected but not necessarily explicitly mentioned), then hearers will infer the actual existence of these blank events and will often confuse what they have inferred with what they have actually been told. Consider the following two paragraphs:

> John went into a restaurant and ordered a hamburger. He enjoyed it and left the waitress a big tip.
>
> John got on the bus to go to Grant Avenue. The driver wouldn't make change for him but an old lady helped him out.

We would expect that people would have no trouble answering the questions 'Who served John the hamburger?' and 'Did John pay the fare?' Yet neither of these events was explicitly in the stories. Rather, they were transmitted implicitly by the scripts attached to those stories. We would claim that scripts take over the understanding task in everyday situations to such a degree as to blur the distinction between what one hears and what one infers.

The general form of a script is a causal chain that connects the expected events of a given situation. A restaurant script might have in it a set of different scenes, each dealing with the kinds of interactions that take place in a restaurant. For example, we might have an 'enter' scene, followed by a 'sitdown' scene, followed by an 'order' scene, followed by an 'eat' scene, followed by a 'pay' scene, and so on. There would be many possible variations that such scenes could have. One possible variety of the 'order' scene might be

MTRANS desire for a menu to waitress
ATRANS menu to customer
MTRANS read menu
MBUILD decide on order
MTRANS order to waitress

Of course, possible variations would have to be built into such a scene. The enabling conditions for each act would have to be checked and if found to be unsatisfied would have to be rectified. (For example, if the waitress isn't near you she will have to be signaled in order to MTRANS the order.)

Such a script is useful because it fills in the blanks in our understanding. (This is similar to the Frames idea suggested by Minsky [Minsky, 1974]). So if we are told that John sat down and later we find that he ate, we can assume that the 'order' scene took place. Alternatively, if we hear that John asked the waitress for the menu and then ordered a steak, we would want to assume that he read the menu, that a steak was on it, and that his order was transmitted to the waitress. All these pieces of information come from the 'order' scene and would be complicated to figure out without such a piece of scriptlike information.

Plans

We might ask where scripts come from. One answer is that scripts are simply plans that have been used a lot.

Plans are sets of information that are attached to the various goals that people have. Some goals we can assume people have are to eat, to be warm, to possess certain things, to have respect, to have power, to know things, to love, and so on. Attached to each of these goals are methods known to the person that are useful for achieving the goals. These methods are possibly explicit sequences of actions that will achieve these goals together with the conditions necessary for the performance of those actions. Or they may be general, only partly formed action lists that have crucial parts either unknown or unachievable that prevent their being put into effect.

However, whether or not a person is capable of achieving a certain goal has little to do with whether he can understand sentences that make use of such goal-achieving or planful knowledge. Consider the following sequences:

John wanted to become chairman of the department. He went and got some arsenic.

John wanted to be chairman of the department. He invited the Dean over for dinner.

John was lonely. He thought if he could find a cab the driver might be able to help him out.

People have general knowledge about plans. They know that to achieve the power goal one has several options. You can do favors for people currently in power, get powerful people to like you, or get rid of the competition, as three (possibly interrelated) options. People use this information to interpret other people's behavior. If you don't have the information that pertains to someone's plan then it is impossible to interpret what you perceive. Specialized knowledge about what is being referred to is needed for the third paragraph above. In the absence of this knowledge about one type of plan, this paragraph is incomprehensible.

To give a general idea about how plans work, consider an anthropomorphic bear. This bear lives in a bear world but he and the animals he interacts with can talk. Such a bear has some simple goals: to satiate his hunger, his sex drive, and his need for rest and to preserve his health. Suppose our bear is hungry. He must get some food. Having food in his control will enable the desired INGEST act. The bear uses the plan GET(X) where X is food. GET(X) is a plan that will, he hopes, translate into a sequence of acts that the bear can perform that will result in the desired state (CONTROL(food)), which enables the INGEST. GET(X) is rewritten into two subplans (FIND(X), which will enable TAKE(X). FIND(X) requires knowing where X is located, so this is translated into ΔKNOW (LOC(X)). (Here we borrow the change of state notation called deltacts [Abelson, 1975a]).

The completion of the change of knowledge will enable a PTRANS to the new location, so

$$\text{FIND(LOC(X))} = \Delta\text{KNOW(LOC(X))} + \text{PTRANS(TO LOC(X))}.$$

Many possible plans come from ΔKNOW. If someone else might know, the plan called ASK might be used. The ASK plan consists of the act to be done (an MTRANS of the question 'Where is the food?') and the preconditions that must be satisfied in order to make that plan have a positive outcome. For example, in order to ask, the locations of the asker and the asked must be the same (in the bear world). In addition, the asked must want to convey the answer. The ASK plan is responsible for telling the user how to overcome any unsatisfied preconditions. For example, if the asked doesn't wish to tell, then the asker may wish to THREATEN. The result of the ASK plan is that the asker now knows where the food is. This enables a PTRANS and now the problem of TAKING the food must be met. This comes under ΔCONTROL(X). Under ΔCONTROL(X) there are many possible plans, one of which is BARGAIN OBJECT. If this plan is successful, then control is gotten and the INGEST can take place.

Planning sequences such as that above can be used for both telling and understanding stories. Using the structure and choices above, a generated story might be:

> Joe Bear was hungry. He went to Irving Bird and asked him whether he knew of any bees' nests. Irving said he wouldn't tell him, so Joe threatened to bust him in the beak. Irving said there was a nest two trees down. Joe Bear went there and offered the bees a bunch of flowers in exchange for their honey. The bees agreed and Joe sat down and ate.

The structures above could be used for understanding sequences such as:

> Joe Bear was hungry.
> He found Irving Bird.

Where it was known that birds were not sources of food for bears it could be determined that Irving might be a source of information.

Planning structures are an integral part of memory processes. People know how others handle the world and thus can understand pieces of a plan that they are told about. The theoretical entities that are part of these plans (such as ASK and THREATEN) are simply names of possible sequences of events that will yield the desired result. (They should not be confused with the conceptual representation of the meaning of these words, which is something apart from the intention of an act and its place within a plan. You needn't use the words 'ask' or 'threaten' in order to ASK or THREATEN. The conceptual entities underlying these and other words call up plans that might have been intended.)

When a plan is used often enough it becomes a script. What are scripts for some people are not necessarily scripts for others. What is important here is that language understanding simply cannot take place in the absence of the knowledge that is in memory. Higher level structures such as plans and scripts are just glorified inference techniques. The basis of understanding is the assignment of new inputs to previously stored episodes in memory that will make sense of them. If relevant knowledge is found in memory, understanding can be achieved. Otherwise it cannot be.

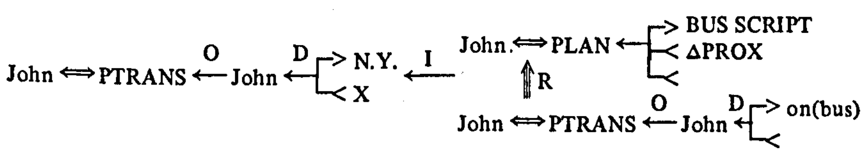
REPRESENTATION OF STORIES

Now let us reconsider the problem of the representation of sentences with respect to their place in a story. Consider the sentence

John went to New York by bus.

This sentence refers to a PTRANS from a sentence point of view, and the bus script called by ΔPROX from the point of view of stories. Should the entire bus script be instrument for the PTRANS here? The issue is entirely representational since the question is not whether John got on the bus or waited for it or whether there was a bus driver, all of which were certainly true. The question is whether these items ought to be part of the representation.

Certainly what must be present as a representation of this sentence is at least the initial PTRANS to New York and probably an instrumental PTRANS to on a bus. This PTRANS to on a bus calls up the BUS SCRIPT. What must be in a representation are the conceptual elements of which we are certain along with a ready access to those element that are somewhat less certain or detail-like that we can expand and peruse when the occasion arises (i.e., when a question is asked). So we propose as the analysis of this sentence:



The above is intended to show that John accomplished his PTRANS by means of getting on a bus which was the first step in a PLAN of his. His plan was to satisfy the goal ΔPROX by means of using the BUS SCRIPT.

We are creating a new primitive ACT–PLAN which has two types of input objects–the goal (a deltact often) and some external circumstances relating to the selection of a plan that will realize that goal. (For example, poverty or not knowing how to drive, might be reasons for selecting BUS SCRIPT here). The output of PLAN is the particular plan selected.

BUS-SCRIPT is the name of such a plan. It and ΔPROX are the first non-primitive elements we have allowed in a Conceptual Dependency representation. For years, we have argued that 'sweat', 'dance', 'kiss' and other complicated concepts would have to be broken down into primitive elements. Other researchers have argued that this is too complicated to do and work with, but we have felt there was little choice, given that concepts like 'perspire,' 'foxtrot' and 'embrace' etc. all need reference to the similar elements that underlie them all. So here we are creating non-primitive concepts. What is the difference?

One difference is that these non-primitives do not come directly from words. Rather, they are names of organized bits of knowledge. Often we need to know what knowledge group has been referenced in order to have access to it later on. We may access it when we are asked a question, in which case we can leave the non-primitive unexpanded until that time. However, if some specific piece of the unexpanded plan is mentioned, it must be expanded to accommodate it. This would occur in a story when the first line introduced a plan or script and the second line made explicit some piece of it.

The new primitive ACT PLAN is responsible for PLANning to do something. It serves to relate bodies of knowledge together so as to make explicit the relationship between a goal and the plan to realize that goal. We define a goal as anything that somebody wants. Consequently there can be an infinite possibility of goals for any person. However, there exists a standard set of goals that we can safely assume all people have from time to time. Some of these goals are very low level, like ΔPROX. That is, ΔPROX probably exists only as a subgoal in the realization of some higher level goal. The cognizing or MBUILDing of a goal leads to (and is often the instrument of) a PLAN. Thus we represent 'John wanted to go to New York' as

$$\text{John} \Longleftrightarrow \text{MBUILD} \leftarrow \begin{cases} \rightarrow \text{GOAL (John)} = \Delta\text{PROX (John, N.Y.)} \\ < \end{cases}$$

This is just a shorthand for the old 'going to New York would cause him to be happy'.

Consider the problem of possible sentences following 'John went to New York by bus'.

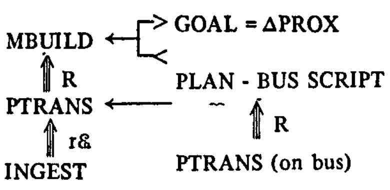
1) After he got there he had some cheese cake.
2) First, he bought a ticket.

Sentence 1) represents the class of sentences that continue the causal chain set up by the PTRANS to New York. The PTRANS results in a location change which enables a PTRANS to wherever they sell cheese cake. Thus, a causal chain can be built according to the rules in Schank (Schank, 1973, 1975) which indicates that this story and the analysis of it is okay.

But for 2) a different problem exists. 2) does not follow in a causal chain from the initial sentence. John's going to New York did not result in or enable 2). Rather 2) is an elaboration of a detail of the initial sentence. The question is how to represent this. The answer must rely on the notion of a GOAL within the ACT PLAN. It is not possible to elaborate upon a description without essentially spelling out the details of a plan to achieve a goal. However, in order to connect 2) to the ΔPROX in the PLAN ACT, we must distinguish between the PLANning of a ΔPROX and the doing of those ACTs that are in the PLAN. That is, there is a difference between thinking about something and doing it.

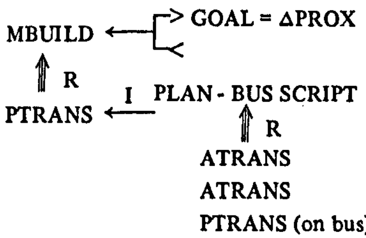
So what we are saying is that sentence 2) is represented twice in the instrument of the main PTRANS. It appears once as the first step in the ΔPROX that is part of the PLAN ACT. It appears in its physical realization as the first ACT caused by this PLAN as reason. Thus the representations of the two stories are as follows:

John went to New York by bus.
After he got there he had some cheesecake.



Here, we indicate the desire to go to New York (MBUILD) was a reason for going (PTRANS) which was done by PLANning a Bus Script, the first step of which we are sure was getting on the bus (PTRANS). The main PTRANS resulted in a location change which enabled an INGEST to a restaurant (abbreviated as *r*&). Hidden in the *r*& is perhaps another PTRANS to a restaurant which we could infer later if need be.

For sentence 2 the representation is:



Here we have that the MBUILD of the goal was the reason for the PTRANS to New York which was done by PLANning it out and executing the steps of getting a ticket and getting on the bus. Any continued description of events connected with the bus script would appear as instrument to the main PTRANS.

This is a major theoretical point and it is worth going into it in detail. We have stated that conceptualizations are tied together by causal chains. Since every ACT has a set of instuments and one of the main causal chain pieces is *enable* causation, there has been some confusion between causality and instrumentality. This confusion existed primarily because of the somewhat artificial problem of analyzing sentences in isolation.

We can see from the above that what is the case is that instruments are a *fuller expansion* of the ACT that they are instrumental to. With respect to causal chains they replace the main ACT that they are instrumental to. The purpose of the main ACT then is to give a handle on the large amount of information in the instrumental chain. We now explain.

From the point of view of causal chains, we wish to connect together every event that results in a state which enables an event and so on. It was our intent (Schank, 1973 and 1975) to connect only those items that could actually have caused each other. Thus, PTRANS can't cause INGEST, it must result in a location change which is near food, which together with some other conditions can enable an INGEST. Suppose the sentence preceding our story had been "John was sitting at home in New Haven when he decided to go to New York and get some cheese cake." This sentence would have filled out the decision process in the initial MBUILD as well as filling in the initial condition and starting point for the bus trip. However, to go from sitting in a New Haven house to a place in New York is a complicated process. By our own rule we would have to specify all the enablements and results in order to enable (from the initial sitting condition) the PTRANS to New York. Now in fact, nothing we could do could enable this PTRANS to New York that would not include specifying at least the entire bus script. What is actually the case is that the entire instrumental chain (including the PLAN and the bus script) should complete the correct causal chain between the initial position (in New Haven) and the final position (in New York). That is, we seem to not need PTRANS at all, rather we just replace it with its instrumental realization.

So shall we replace the PTRANS here (and perhaps all PTRANSes and other such global actions)?

To answer this we must consider what it is we are trying to do. Our aims are two. First, we wish to build a computer program that can understand reasonably complex stories. Second, we wish to come up with a good theory of human processing of such stories.

Consider, for a moment, a human story understander. When a person reads a 300 page novel he does not (unless he is very unusual) remember all the conceptualizations stated in the story in the form of a giant causal chain. Rather he remembers the gist of the book. Maybe 5 or 10 pages of summary could be extracted from him after reading the book. Previously we have said that Conceptual Dependency Theory will account for memory for gist of sentences. But it cannot be seriously proposed that this is all that is needed for gist of long and complex stories. Some other explanation must be given.

One small part of this explanation lies in the above problem. The concept PTRANS is used as an information organizer. Its presence says something about the goals and intentions of the actor and the overall results and enabling conditions that are important to the story. Without a main PTRANS here, we would be unable to separate the importance of the step onto the bus from the arrival in New York. The main PTRANS serves to focus that importance. It is an abstraction that names the entire sequence. It thus can be remembered that way. It allows for the hearer to concentrate on the main flow and retrace the details later (here, knowing there was a PTRANS allows for tracing what instruments could have been chosen, and recalling the bus, remembering the purchase of the ticket).

We are saying that this is what happens in an example of large text understanding. In a three line story, the process is probably quite different. But embedded in 10,000 kinds of text, such organizational schemes make it possible to remember at all and not get flooded by a morass of conceptualizations.

In a recent experiment, Abelson (Abelson, 1975b) showed that people remember stories better when they are asked to take some particular point of view (of one of the participants or of an observer in a particular place), and that what they remember is contingent on which point of view they had. The ramifications of this experiment for a theory of language understanding have to be that when people have a clue of what to forget they do better at remembering. In other words, good forgetting is the key to remembering. Likewise, if we want to build programs that remember, we had best teach them how to forget. One method of forgetting is simply not noticing levels of detail that are there. This can be done by treating the instruments for an action at a different level of detail than the main ACTs that they explain. When looking at a story at one level of detail we would not see the level of detail underneath it unless specifically called upon to do so (e.g., to answer the question 'Did he pay money to get to N.Y.?').

The principle here is a very powerful and important one. It has ramifications at other levels of understanding as well. Before we get into these, however, we shall attempt to finish our discussion of the representation of stories using scripts and plans.

What we are saying then is that the primitive ACTs essentially perform two functions. First, they represent the concept of action that was stated in a sentence. Second, some of the primitive ACTs, in particular the TRANS ACTs, also serve as names of larger pieces of information. 'John's PTRANS to New York' can be considered both the conceptual realization of a sentence such as 'John went to New York' and the handle by which all of the events that took place on the trip itself can be organized.

Stories are connected in memory of means of causal chains. We are saying now that there are many kinds of chaining. For example, a story might be represented at the highest level as a PTRANS to New York, an INGEST of cheese cake and a PTRANS back to New Haven. But intermediate between these ACTs would be a lot of information. This information is about the actual events that made up the causal chain between the three principal ACTs of the story.

The instruments themselves are causal chains which could also have main parts and subparts. In order to get to the lowest level of detail might require going through several subprincipal ACTs (the fight in the subway on the way to the restaurant.)

At the end of the lowest level of instrumentality would provide a complete causal chain. This chain would never be looked at all at once, but rather the various subhandles would tell the story at various levels of detail.

For example, suppose we had the following story:

> John wanted some cheesecake. He decided to go to New York. He went to New York by bus. On the bus he met a nice old lady who he talked to about the prices in the supermarket. When he left the bus he thanked the driver for the ride and found the subway to go to Lindy's. On the subway he was reading the ads when suddenly he was robbed. He wasn't hurt though and he got off the train and entered Lindy's and had his cheesecake. When the check came, he said he couldn't pay and was told he would have to wash dishes. Later he went back to New Haven.

This story is represented in the new format as:

PTRANS (to N.Y.)	ATRANS (ticket)	
	PTRANS (on bus)	MOVE (foot)
PTRANS (to restaurant)	PTRANS (to on subway)	
	ATTEND (look at fight)	MTRANS (fare)
	ATRANS (get robbed	MTRANS (talk to robber)
	PTRANS (to restaurant)	
INGEST (cheesecake)	not ATRANS (no money to pay check)	
	DO cause CLEAN (wash dishes)	
	PTRANS (to bus station)	
PTRANS (to New Haven)		

Obviously there is more to marking the gist of a story than looking at the principal ACTs without noticing instruments. It is also necessary to mark unusualness at any place that it occurs, so that it is not ignored. The gist of the above story would probably have to mention the robbery and its consequence of having to wash dishes. This event is more important than the general chain of events.

The point to be made here is this. People learn how to ignore details. Scripts can be added into stories representation and understanding by only looking at the important parts and keeping the rest under the surface. Similarly, any event can be handled by looking at the principal ACT that names the sequence and by looking at the sequence underlying it when necessary.

The process of understanding then, is one of constructing a list of pointers (primitive ACTs) into a flow of events. The list of principal primitive ACTs that is thus constructed is the 'gist' of the story minus any problems that come up in the normal flow of events. Thus, the 'gist' of the above story could be 'John went to New York for cheese cake and then returned home' (PTRANS, INGEST, PTRANS). This gist will work whenever no unusual or problem events occur in a story. But, if a story has these unusual events (as do most interesting stories), these too would be remembered. Thus there must be more than just a straight conceptual dependency causal chain as a representation of a story. There must be an addition to the record of the flow of events (indexed by the principal ACTs as pointer into the causal chain).

This addition we call the WEIRD LIST. The WEIRD LIST is a combination of conceptualizations that were found to be weird by various metrics, together with the belief structures that caused them to be understood as weird. Events on a weird list are given a special place so that they can be tied to their consequences as exemplified later in a story. Thus, here, getting robbed is weird (not on the script for subway riding) so it is placed on the weird list. Its consequence is a lack of money and now we expect a request (or demon) to look for any new events whose enabling conditions require money. 'Going to a restaurant' is one of those and it gets tied to the weird list. This is done by the standard means of causal chaining, triggered in this case by the request for money-needing events. That is, an unenable causal connection is established between 'no money' (the result of the ATRANS in 'rob') and the necessary condition (of money) for the ATRANS in the restaurant script.

Immediately, a problem is created and this problem is put on the weird list too. The problem is 'What will happen next in the restaurant script as a result of no ATRANS?' This problem and its resolution are put on the weird list too regardless of the resolution ('He found money in this shoe' would serve as a resolution too and would appear on the weird list). A problem being resolved is even more weird and would be specifically marked that way on the weird list. (Paying for the cheesecake as if nothing had happened would be an example).

The end result of the story understanding process would thus be: (1) A complete causal chain of conceptualizations connected at the level of the principal primitive ACTs with instrumentality available as necessary and (2) a weird list tabulating problems and their resolution.

The process of summarization would thus be done automatically by consultation of the principal ACTs and the weird list. Such a summary would be:

John went to New York for some cheesecake. He got robbed on the subway and had to wash dishes in the restaurant when he couldn't pay for the cheesecake. Then he went home.

The serious question here is, of course, how is this all to be computed? The answer is that as long as scripts are involved (and we aver that they are involved in a tremendous part of understanding (see also Abelson [Abelson, 1975b]); the answer is relatively easy. First, scripts are computed and stored as instruments of principal actions (based on the goals of the actor (the deltacts and others). As long as scripts are referenced they can be reconstructed by the hearer and are thus easily forgotten. Thus, the process of understanding, where scripts are involved, is one of ascertaining what script is being referred to and establishing therefore what the actors' goals are. The scripts are thus 'forgotten' in the sense that they are stored elsewhere than in the main causal chain. The exceptions to this are events that are not predicted by the script. These events are placed within the instrument but are also added to the weird list. The method for doing this is simply lack of correspondence with predicted script parts. When such disjointness is noted, it is put on the weird list and from that point the computed probable consequences of the weird event are 'kept in mind' (as requests) whenever a new script is entered. New scripts are, from that point, expected to have deviances if their entry conditions were affected by the weird event. They are thus paid closer attention to (kept within the main causal chain).

One of the major issues in Artificial Intelligence research must be the creation of a theory of forgetting. It simply is not possible to assume that people do, or that machines should, remember everything they encounter. In listening to a speaker, reading a book, or engaging in a conversation, people could not possibly remember everything they are told verbatim. In attempting to get the gist of a sequence, they must employ what we call *forgetting heuristics*. As part of these forgetting heuristics are heuristics that search out items of major importance. The selection of these major items is the key to forgetting. We don't really wish to assert that people couldn't possibly remember everything they hear. Rather we wish to find a procedure that will let us see only the major items, yet also find, with some difficulty, the thoughts or statements that underlie them, and the ideas that underlie those, and so on.

Thus, the key to understanding must be, in order to facilitate search among what has been understood, an organization of the new information, in such a fashion as to seem to forget the unimportant material and to highlight the important material. Forgetting heuristics must do this for us. So the first task before us is to establish what the most significant items in a text are likely to be, and then to establish the heuristics which will extract and remember exactly those items.

REFERENCES

Abelson, R.P. (1975a) Concepts for representing mundane reality in plans. *Representation and Understanding: Studies in Cognitive Science* (eds. Bobrow, D. and Collins, A.), Academic Press.

Abelson, R.P. (1975b) Does a story understander need a point of view? *Using Knowledge to Understand.* (eds. Schank, R. and Nash-Webber, B.) Proceedings of the Conference on Theoretical Issues in Natural Language Processing.

Goldman, N. (1975) Conceptual generation. *Conceptual Information Processing,* (ed. Schank, R.), North Holland Publishing, Amsterdam.

Katz, J. (1967) Recent issues in semantic theory. *Foundations of Language,* III.

Minsky, M. (1974) Frame-systems. *AI memo*, MIT.

Rieger, C. (1975) Conceptual memory. *Conceptual Information Processing,* (ed. Schank, R.) North Holland Publishing, Amsterdam.

Riesbeck, C. (1975) Conceptual Analysis. *Conceptual Information Processing,* (ed. Schank, R.) North Holland, Publishing, Amsterdam.

Simmons, and Slocum (1972) Generating English discourse from semantic networks. *Comm. of the ACM.* 15, 10.

Schank, R. (1972) Conceptual dependency: a theory of natural language understanding. *Cognitive Psychology,* 3, 4.

Schank, R. (1973) Causality and reasoning. *Technical Report No. 1.* Instituto per gli studi Semantici e Cognitivi. Castagnola, Switzerland.

Schank, R. (1975) The structure of episodes in memory. *Representation and Understanding: Studies in Cognitive Sciences*, (eds. Bobrow, D. and Collins, A) Academic Press.

Schank, R., Tesler, L. and Weber, S. (1970) Spinoza II: Conceptual Case-Based Natural Language Analysis. *AI Memo, No. AIM-109,* Computer Science Department, Stanford University, Stanford, California.

Winograd, T. (1971) Procedures as Representation for Data in a Computer Program for Understanding Natural Language. *TR-84,* MIT Project MAC.