Nick Cercone Gordon McCalla
Editors

The Knowledge Frontier

Essays in the Representation of Knowledge

With 93 illustrations

Springer-Verlag
New York Berlin Heidelberg
London Paris Tokyo

© 1987

11. SNePS Considered as a Fully Intensional Propositional Semantic Network

Stuart C. Shapiro
William J. Rapaport

Department of Computer Science
University of Buffalo
State University of New York
Buffalo, New York 14260

Abstract

SNePS, the Semantic Network Processing System, is a semantic network language with facilities for building semantic networks to represent virtually any kind of information, retrieving information from them, and performing inference with them. Users can interact with SNePS in a variety of interface languages, including a LISP-like user language, a menu-based screen-oriented editor, a graphics-oriented editor, a higher-order-logic language, and an extendible fragment of English.

This article discusses the syntax and semantics for SNePS considered as an intensional knowledge representation system and provides examples of uses of SNePS for cognitive modelling, database management, pattern recognition, expert systems, belief revision, and computational linguistics.

11.1. Introduction

This chapter presents a formal syntax and semantics for SNePS, the *S*emantic *Ne*twork *P*rocessing *S*ystem (Shapiro, 1979b).[30] The syntax shows the emphasis placed on SNePS's *propositional* nature. The semantics, which is based on Alexius Meinong's theory of intentional objects (the objects of thought), makes SNePS's *fully intensional* nature precise: as a fully intensional theory, it avoids possible worlds and is appropriate for AI considered as "computational philosophy" - AI as the study of how intelligence is possible - or "computational psychology" - AI with the goal of writing programs as models of *human* cognitive behavior. We also present a number of recent AI research and applications projects that use SNePS, concentrating on one of these, a use of SNePS to model (or construct) the mind of a cognitive agent, referred to as CASSIE (the *C*ognitive *A*gent of the *S*NePS *S*ystem-an *I*ntelligent *E*ntity).

11.1.1. The SNePS environment

A semantic network is a data structure typically consisting of labeled nodes and labeled, directed arcs. SNePS can be viewed as a semantic network language with facilities for

1. building semantic networks to represent virtually

[30]This research was supported in part by the National Science Foundation under Grant No. IST-8504713 and SUNY Buffalo Research Development Fund grants No. 150-9216-F and No. 150-8537-G (Rapaport), and in part by the Air Force Systems Command, Rome Air Development Center, Griffiss Air Force Base, NY 13441-5700, and the Air Force Office of Scientific Research, Bolling AFB, DC 20332 under contract No. F30602-85-C-0008 (Shapiro). We wish to thank Michael Almeida, James Geller, João Martins, Jeannette Neal, Sargur N. Srihari, Jennifer Suchin, and Zhigang Xiang for supplying us with descriptions of their projects, and Randall R. Dipert, the members of SNeRG (the SNePS Research Group), and three anonymous reviewers for comments and discussion.

any kind of information or knowledge.

2. retrieving information from them, and

3. performing inference with them, using SNIP (the *S*NePS *I*nference *P*ackage) and path-based inference.

Users can interact with SNePS in a variety of interface languages, including: SNePSUL, a LISP-like SNePS User Language; SENECA, a menu-based, screen-oriented editor; GINSENG, a graphics-oriented editor; SNePSLOG, a higher-order-logic language (in the sense in which PROLOG is a first-order-logic language) (McKay and Martins, 1981), (Shapiro, McKay, Martins, and Morgado, 1981); and an extendible fragment of English, using an ATN parsing and generating grammar (Shapiro, 1982), see Figure 11-1.

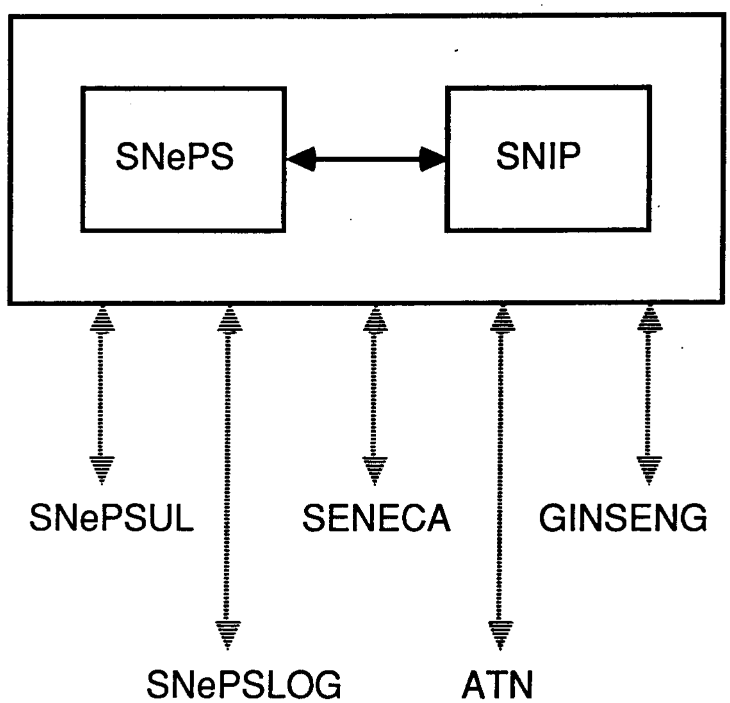


Figure 11-1: SNePS, SNIP and Their User Interfaces.

SNePS, SNIP (the SNePS Interface Package) and their user interfaces. When the arcs, case frames, and ATN grammar are those of SNePS/CASSIE, then the system is being used to model CASSIE. When the arcs are for the database (see Section 11.4.1), then the system is being used as a database management system, etc.

SNePS is the descendent of SAMENLAQ (Shapiro, Woodmansee, and Kreuger, 1968) (Shapiro and Woodmansee, 1969) and MENTAL (Shapiro, 1971b), (Shapiro, 1971c). It was developed with the help of the SNePS Research Group at Indiana University and at the University at Buffalo. The current version is implemented in Franz LISP and runs on VAX 11/750s and 780s in the Department of Computer Science at Buffalo. An earlier version was implemented in ALISP on a CDC Cyber 730; and an updated version is being implemented in Common LISP on Symbolics LISP machines, TI Explorers, and a Tektronix 4406. There are additional installations at other universities in the U.S. and Europe.

11.1.2. SNePS as a knowledge representation system

Some researchers, for example, (Levesque and Brachman, 1985), view a knowledge representation (KR) system as a subsystem that manages the knowledge base of a knowledge-based system by storing information and answering questions. In contrast, we view SNePS as the entire knowledge-based system, interacting with a user/interlocutor through one of its interfaces. Of course, the user/interlocutor could be another computer program using SNePS as a subsystem, but that is not the way we use it.

A basic design goal of SNePS and its ancestors was to be an environment within which KR experiments could be performed, that is, to be a semantic network at the "logical" level, to use Brachman's term (Brachman, 1979), see Section 11.5, below. This has been effected by providing a rather low level interface, SNePSUL. Using SNePSUL, a KR designer can specify a syntax: individual arc labels and sets of arc labels (or case frames) that will be used to represent various objects and information about them. It is also the designer's obligation to supply a semantics for these case frames. As is the case for any provider of a language or "shell", we cannot be responsible for what use others make of the facilities we provide. Nevertheless, we have our own preferred use.

In this chapter, we try to do two things. First, we try to provide an understanding of SNePS and of some of the uses to which it has been put. Second, and most importantly, we present our own preferred use: this is to use SNePS, with a particular set of arc labels and case frames, and a particular parsing/generating grammar for a fragment of English, as (a model of) the cognitive agent, CASSIE. We shall refer to SNePS with these arcs, case frames, and grammar as SNePS/CASSIE. SNePS/CASSIE forms CASSIE's mind and stands as our current theory of KR at the "conceptual" level (cf. Section 11.5, below, and (Brachman, 1979)). The purpose of the central part of this paper is to present this theory by explaining the entities represented by structures in SNePS/CASSIE, by giving a formal syntax and semantics for those structures, and by showing and explaining a sample conversation with CASSIE.

11.1.3. Informal description of SNePS

Regardless of the intentions of a KR-system designer, SNePS, as a KR formalism, provides certain facilities and has certain restrictions. The facilities (for example, for building, finding, and deducing nodes) are best understood as those provided by SNePSUL, but we shall not give a complete description of SNePSUL here. [For an example, cf. Section 11.4.1, below; for details, see (Shapiro, 1979b).] The restrictions, however, are important to understand, because they distinguish SNePS from a general labelled, directed graph and from many other semantic network formalisms.

SNePS is a *propositional* semantic network. By this is meant that all information, including propositions, "facts", etc., is represented by nodes. The benefit of representing propositions by nodes is that propositions about propositions can be represented with no limit. (In the formal syntax and semantics given in Section 11.3, the propositions are the nodes labelled '*m*' or '*r*'.)

Arcs merely form the underlying syntactic structure of SNePS. This is embodied in the restriction that one cannot add an arc between two existing nodes. That would be tantamount to telling SNePS a proposition that is not represented as a node. There are a few built-in arc labels, used mostly for rule nodes. *Paths* of arcs can also be defined, allowing for *path-based* inference, including property inheritance within generalization hierarchies [see Section 11.3.4, below; cf. Shapiro (Shapiro, 1978), (Srihari, 1981), and (Tranch, 1982).] All other arc labels are defined by the user, typically at the beginning of an interaction with SNePS, although new labels can be defined at any time.

For purposes of reasoning, propositions that are asserted in SNePS must be distinguished from those propositions that are merely represented in SNePS but not asserted. This could happen in the case of a proposition embedded in another (for example, "Lucy is rich" embedded in "John believes that Lucy is rich"). SNePS interprets a proposition node to be asserted if and only if it has no arcs pointing to it.[31]

Another restriction is the *Uniqueness Principle:* There is a one-to-one correspondence between nodes and represented concepts. This principle guarantees that nodes will be shared whenever possible and that nodes represent intensional objects.[32] We next consider the nature of these objects.

11.2. Intensional knowledge representation

SNePS can be used to represent propositions about entities in the world having properties and standing in relations. Roughly, nodes represent the propositions, entities, properties, and relations, while the arcs represent structural links between these.

SNePS nodes *might* represent *extensional* entities. Roughly, extensional entities are those whose "identity conditions" (the conditions for deciding when "two" of them are really the "same") do not depend on their manner of representation. They

---

[31]This is not really a restriction of SNePS, but of SNIP (the SNePS Inference Package) and path-based inference.

[32]In (Maida and Shapiro, 1982) this name was given to only half of the Uniqueness Principle as stated here: "each concept represented in the network is represented by a unique node" (page 291).

may be characterized as those entities satisfying the following rough principle:

Two extensional entities are equivalent (for some purpose) if and only if they are identical[33]

For example, the following are extensional:

> the Fregean referent of an expression;
> physical objects;
> sentences;
> truth values;
> mathematical objects such as:
> sets,
> functions defined in terms of their input–output behavior (that is, as sets of ordered pairs),
> *n*–place relations defined in terms of sets of ordered *n*–tuples.

Although SNePS *can* be used to represent extensional entities in the world, we believe that it *must* represent *intensional* entities. Roughly, intensional entities are those whose identity conditions *do* depend on their manner of representation. They are those entities that satisfy the following rough principle:

Two intensional entities might be equivalent (for some) purpose without being identical (that is, they might really be two, not one).

Alternatively, intensional entities may be characterized as satisfying the following five criteria:

1. They are non-substitutible in referentially opaque contexts.

2. They can be indeterminate with respect to some properties.

3. They need not exist.

---

[33] that is, if and only if "they" are really one entity, not two

4. They need not be possible.

5. They can be distinguished even if they are necessarily identical (for example, *the sum of 2 and 2* and *the sum of 3 and 1* are distinct objects of thought).

For example, the following are intensional:

```
the Fregean sense of an expression;
concepts;
propositions;
properties;
algorithms;
objects of thought, including:
    fictional entities (such as Sherlock Holmes),
    non-existents (such as the golden mountain),
    impossible objects (such as the round square)
```

Only if one wants to represent the relations between a mind and the world would SNePS also have to represent extensional entities [cf. (Rapaport, 1976), (Rapaport, 1978), (McCarthy, 1979)]. However, if SNePS is used just to represent a mind - that is, a mind's model of the world-then *it does not need to represent any extensional objects*. SNePS can then be used either to model the mind of a particular cognitive agent or to build such a mind - that is, to *be* a cognitive agent itself.

There have been a number of arguments presented in both the AI and philosophical literature in the past few years for the need for intensional entities, (Castaneda, 1974), (Woods, 1975), (Rapaport, 1976), (Rapaport, 1985a), (Brachman, 1977), (Routley, 1979), cf. (Rapaport, 1984a), (Parsons, 1980), cf. (Rapaport, 1985b)). Among them, the following considerations seem to us to be especially significant:

Principle of Fine-Grained Representation:

```
The objects of thought (that is, intentional objects)
are intensional: a mind can have two or more objects of
thought that correspond to only one extensional object.
```

To take the classic example, the Morning Star and the Evening Star might be distinct objects of thought, yet there is only one extensional object (viz., a certain astronomical body) corresponding to them.

Principle of Displacement:

> Cognitive agents can think and talk about non-existents: a mind can have an object of thought that corresponds to no extensional object.

Again to take several classic examples, cognitive agents can think and talk about fictional objects such as Santa Claus, possible but non-existing objects such as a golden mountain, impossible objects such as a round square, and possible but not-yet-proven-to-exist objects such as theoretical entities (for example, black holes).

If nodes only represent intensional entities (and extensional entities are not represented in the network), how do they link up to the external, extensional world? In SNePS/CASSIE, the answer is by means of a LEX arc (see syntactic formation rule SR.1 and semantic interpretation rule SI.1 in Section 11.3.3, below): the nodes at the head of the LEX arc are *our* (the user's) interpretation of the node at its tail. The network without the LEX arcs and their head-nodes displays the *structure* of CASSIE's mind [cf. (Carnap, 1967), Section 11.14].

A second way that nodes can be linked to the world is by means of sensors and effectors, either linguistic or robotic. The robotic sort has been discussed in (Maida and Shapiro, 1982). Since so many AI understanding systems deal exclusively with language, here we consider a system with a keyboard as its sense organ and a CRT screen as its only effector.

Since the language system interacts with the outside world only through language, the only questions we can consider about the connections of its concepts with reality are questions such as:

> Does it use words as we do?
> When it uses word *w*, does it mean the same thing as when I use it?
> When I use word *w*, does it understand what I mean?

The perceptual system of the language system is its parser/analyzer - the programs that analyze typed utterances and build pieces of semantic network. The motor system is the generator - the programs that analyze a section of the semantic network and construct an utterance to be displayed on the CRT. One crucial requirement for an adequate connection with the world is simple consistency of input-output behavior. That is, a phrase that is analyzed to refer to a particular node should consistently refer to that node, at least while there is no change in the network. Similarly, if the system generates a certain phrase to describe the concept represented by a node, it should be capable of generating that same phrase for that same node, as long as nothing in the network changes. Notice that it is unreasonable to require that if a phrase is generated to describe a node, the analyzer should be able to find the node from the phrase:: The system might know of several brown dogs and describe one as "a brown dog"; it could not be expected to find that node as the representation of "a brown dog" consistently.

If we are assured of the simple input-output consistency of the system, the main question left is whether it uses words to mean the same thing as we do. It is the same question that we would be concerned with if we were talking with a blind invalid, although in that case we would assume the answer was 'Yes' until the conversation grew so bizzare that we were forced to change our minds. As the system (or the invalid) uttered more and more sentences using a particular word or phrase, we would become more and more convinced that it meant what we would mean by it, or that it meant what we might have described with a different word or phrase ("Oh! When you say 'conceptual dependency structure', you mean what I mean when I say 'semantic network'."), or else that we *didn't* know what was meant, or that it was not using it in a consistent, meaningful way (and hence that the system (or invalid) did not know what it was talking about). As long as the conversation proceeds without our getting into the latter situation, the system has all the connections with reality it needs.

11.3. Description of SNePS/CASSIE

In this section, we introduce CASSIE, and give the syntax and semantics for SNePS/CASSIE in terms of a philosophical theory of mental entities inspired by Alexius Meinong's Theory of Objects.

11.3.1. CASSIE - A model of a mind

SNePS nodes represent the objects of CASSIE's thoughts - the things she thinks about, the properties and relations with which she characterizes them, her beliefs, her judgments, etc. [cf. (Maida and Shapiro, 1982), (Rapaport, 1985a)]. According to the Principle of Displacement, a cognitive agent is able to think about virtually anything, including fictional objects, possible but non-existing objects, and impossible objects. Any theory that would account for this fact requires a non-standard logic, and its semantics cannot be limited to merely *possible* worlds. (Otherwise, it could not account for impossible objects. This accounts for the difficulties David Israel has in providing a possible-worlds semantics for SNePS (Israel, 1983), (cf. (Rapaport, 1985a)). Theories based on the Theory of Objects of the turn-of-the-century Austrian philosopher-psychologist Alexius Meinong are of precisely this kind.

For present purposes, it will be enough to say that Meinong held that psychological experiences consist in part of a psychological *act* (such as thinking, believing, judging, wishing, etc.) and the *object* to which the act is directed (for example, the object that is thought about or the proposition that is believed). Two kinds of Meinongian objects of thought are relevant for us:

1. The *objectum*, or object of "simple" thoughts: Santa Claus is the objectum of John's act of thinking of Santa Claus. Objecta are the meanings of noun phrases.

2. The *objective*, or object of belief, knowledge, etc.: that Santa Claus is thin is the objective of John's

> act of believing that Santa Claus is thin. Objectives are like propositions in that they are the meanings of sentences and other sentential structures.

It is important to note that objecta need not exist and that objectives need not be true. [For details, see: (Meinong, 1904), (Findlay, 1963), (Rapaport, 1976), (Rapaport, 1978), (Rapaport, 1981), (Rapaport, 1982), (Castaneda, 1974), (Castaneda, 1975a), (Castaneda, 1975b), (Casteneda, 1975), (Castaneda, 1977), (Castaneda, 1979), (Tomberlin, 1984), and (Routley, 1979); cf. (Rapaport, 1984a), (Parsons, 1980); cf. (Rapaport, 1985b), (Lambert, 1983); cf. (Rapaport, 1986a), (Zalta, 1983).]

This is, perhaps, somewhat arcane terminology for what might seem like AI common sense. But without an underlying theory, such as Meinong's, there is no way to be sure if common sense can be trusted. It is important to note that not only are all represented things inten*s*ional, but that they are all objects of CASSIE's mental acts; that is, they are all in CASSIE's mind (her "belief space") - they are all inten*t*ional. Thus, even if CASSIE represents the beliefs of someone else (for example, John's belief that Lucy is rich, as in the conversation in the next section), the objects that she represents as being in that person's mind (as being in his "belief space") are actually CASSIE's representations of those objects - that is, they are in CASSIE's mind.

11.3.2. A conversation with CASSIE

Before giving the syntax and semantics of the case frames employed in representing CASSIE's "mind", we present a conversation we had with her. An ATN parser/generator (Shapiro, 1982) was used to parse the English input into SNePS and to generate the SNePS structures in English. User input is on the lines beginning with the :-prompt; CASSIE's output is on the lines that follow. (Execution and garbage-collection times have been edited out for clarity.)

```
: Young Lucy petted a yellow dog
I understand that young Lucy petted a yellow dog

: What is yellow
```

```
a dog is yellow

: Dogs are animals
I understand that dogs are animals

: Who petted an animal
young Lucy petted a yellow dog

: Lucy is sweet
I understand that young Lucy is sweet

: Lucy is a girl
I understand that sweet young Lucy is a girl

: Girls are people
I understand that girls are people

: What was petted by a person
a yellow dog was petted by sweet young Lucy

: The dog is named Rover
I understand that Rover is a yellow dog

: What was petted by a person
yellow Rover was petted by sweet young Lucy

: John believes that Lucy is rich
I understand that John believes that Lucy is rich

: Who is rich
I don't know.

: Who is sweet
young Lucy is sweet

: John believes that Lucy is old
I understand that John believes that rich Lucy is old

: John is a boy
I understand that John is a boy

: Boys are people
I understand that boys are people

: Dogs are pets
I understand that dogs are pets

: For every p and d if p is a person and d is a pet then p
     loves d
I understand that for every d and p, if p is a person
and
   d is a pet
```

```
  then p loves d

: Who loves a pet
sweet young Lucy loves yellow Rover
and
John loves yellow Rover
```

11.3.3. Syntax and semantics of SNePS/CASSIE

In this section, we give the syntactic formation rules (SR) and semantic interpretations (SI) for the nodes and arcs used in this interaction, together with some other important ones. We return to a more detailed examination of the interaction in the next section. What we present here is our current model; we make no claims about the completeness of the representational scheme. In particular, we leave for another paper a discussion of such structured individuals as the golden mountain or the round square, which raise difficult and important problems with predication and existence. [For a discussion of these issues, see (Rapaport, 1978), (Rapaport, 1985a).]

Information is represented in SNePS by means of *nodes* and *arcs*. Since the meaning of a node is determined by what it is connected to in the network, there are no isolated nodes. Nodes that only have arcs pointing *to* them are considered to be unstructured or *atomic*. They include:

(A1) *sensory* nodes, which represent interfaces with the external world (in the examples that follow, they will represent words, sounds, or utterances);

(A2) *base* nodes, which represent constant individual concepts and properties;

(A3) *variable* nodes, which represent arbitrary individuals (cf. (Fine, 1983)) or arbitrary propositions.

Molecular nodes, which have arcs emanating *from* them, include:

(M1) *structured individual* nodes, which represent structured individual concepts or properties (that is, concepts and properties represented in such a way that their internal structure is exhibited; see the discussion of structured information in (Woods, 1975));

(M2) *structured proposition* nodes, which represent

> propositions; those with no incoming arcs represent *beliefs* of the system.[34] (Note that structured proposition nodes can also be considered to be structured individuals.) Proposition nodes are either *atomic* (representing atomic propositions) or are *rule nodes*. Rule nodes represent deduction rules and are used by SNIP (the SNePS Inference Package) for node-based deductive inference.[35]

For each of the three categories of molecular nodes (structured individuals, atomic propositions, and rules), there are *constant* nodes of that category and *pattern* nodes of that category representing arbitrary entities of that category.

The rules labeled 'SR', below, should be considered as syntactic formation rules for a *non-linear* network language. The semantic interpretations, labeled 'SI', are in terms of Meinongian objecta and objectives, which are intentional objects, that is, objects of thought. Since intentional objects are intensional, our Meinongian semantics is an *extensional* semantics over a domain of *intensional* entities (Meinongian objects).

We begin with a few definitions.[36]

Definition 1

A node *dominates* another node if there is a path of directed

---

[34]There is a need to distinguish structured proposition nodes with no incoming arcs from structured individual nodes with no incoming arcs; the latter, of course, are not beliefs of the system. This is handled by the syntactic formation rules and their semantic interpretations. There is also a need to distinguish between beliefs of the system and those propositions that the system is merely contemplating or "assuming" temporarily [cf. (Meinong, 1983)]. We are currently adding this capability to SNePS by means of an *assertion* operator ('!').

[35]For details, see (Shapiro, 1977), (Shapiro, 1978), (McKay and Shapiro, 1980), (McCarty and Sridharan, 1981), (Shapiro and McKay, 1980), (Shapiro, Martins, and McKay, 1982), (Martins, 1983a).

[36]These are actually only rough definitions; the interested reader is referred to (Shapiro, 1979b), Section 2.1, for more precise ones.

arcs from the first node to the second node.

Definition 2

A *pattern* node is a node that dominates a variable node.

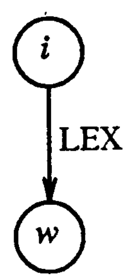
Definition 3

An *individual* node is either a base node, a variable node, or a structured constant or pattern individual node.

Definition 4

A *proposition* node is either a structured proposition node or an atomic variable node representing an arbitrary proposition.

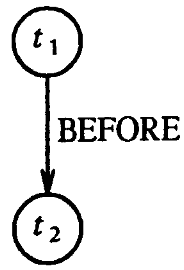
SR.1 If "*w*" is an English word and "*i*" is an identifier not previously used, then



is a network, *w* is a sensory node, and *i* is a structured individual node.

SI.1 *i* is the Meinongian objectum corresponding to the utterance of *w*.

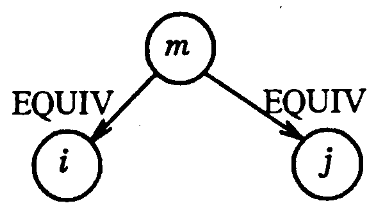
SR.2 If either "t_1" and "t_2" are identifiers not previously used, or "t_1" is an identifier not previously used and t_2 is a temporal node, then



is a network and t_1 and t_2 are *temporal* nodes, that is individual nodes representing times.

SI.2 t_1 and t_2 are Meinongian objecta corresponding to two time intervals, the former occurring before the latter.

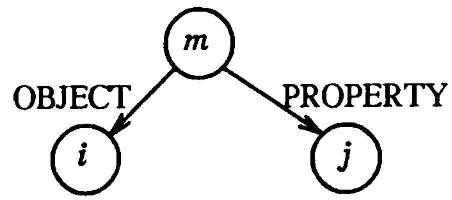
SR.3 If *i* and *j* are individual nodes, and "*m*" is an identifier not previously used, then



is a network and *m* is a structured proposition node.

SI.3 *m* is the Meinongian objective corresponding to the proposition that Meinongian objecta *i* and *j* (are believed by CASSIE to) correspond to the same actual object. (This is not used in the conversation, but is needed for fully intensional representational systems; cf. (Rapaport, 1978;RAPA84b) and (Castaneda, 1974;CAST75b) for analyses of this sort of relation, and (Maida and Shapiro, 1982) for a discussion of its use.)

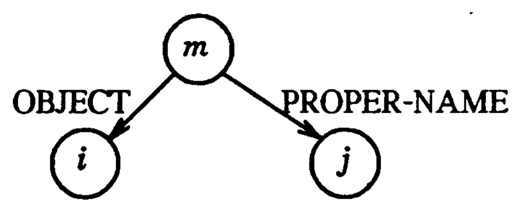
SR.4 If *i* and *j* are individual nodes and "*m*" is an identifier not previously used, then



is a network and m is a structured proposition node.

SI.4 m is the Meinongian objective corresponding to the proposition that i has the property j.

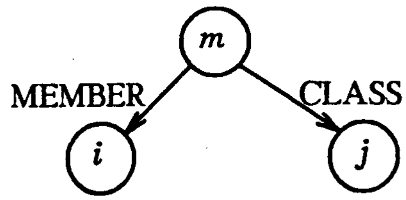
SR.5 If i and j are individual nodes and "m" is an identifier not previously used, then



is a network and m is a structured proposition node.

SI.5 m is the Meinongian objective corresponding to the proposition that Meinongian objectum i's proper name is j. (j is the Meinongian objectum that is i's proper name; its expression in English is represented by a node at the head of a LEX-arc emanating from j.)

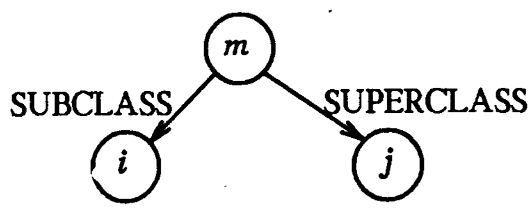
SR.6 If i and j are individual nodes and "m" is an identifier not previously used, then



is a network and m is a structured proposition node.

SI.6 m is the Meinongian objective corresponding to the proposition that i is a (member of class) j.

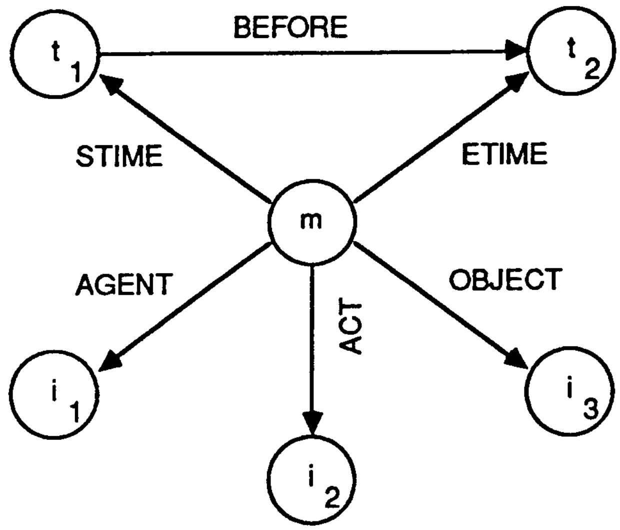
SR.7 If i and j are individual nodes and "m" is an identifier not previously used, then



is a network and m is a structured proposition node.

SI.7 m is the Meinongian objective corresponding to the proposition that (the class of) is are (a subclass of the class of) js.

SR.8 If i_1, i_2, i_3 are individual nodes, t_1, t_2, are temporal nodes, and "m" is an identifier not previously used, then

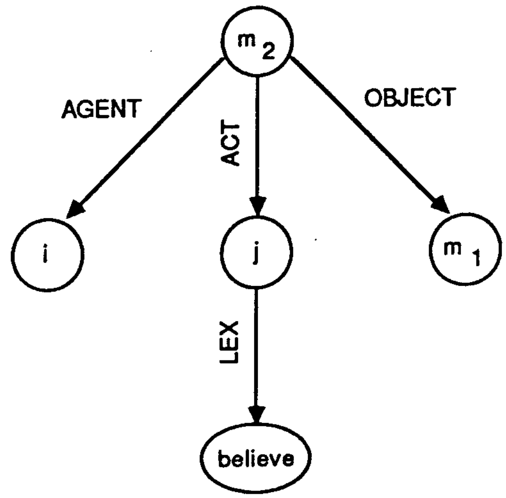


is a network and m is a structured proposition node.

SI.8 m is the Meinongian objective corresponding to the proposition that agent i_1 performs act i_2 to or on i_3 starting at time t_1 and ending at time t_2, where t_1 is before t_2.

It should be noted that the ETIME and STIME arcs are optional and can be part of any proposition node. They are a provisional technique for handling the representation of acts and events; our current research on temporal representation is much more complex and is discussed in Section 11.4.7, below.

SR.9 If m_1 is a proposition node, i is an individual node, j is the (structured individual) node with a LEX arc to the node, believe, and "m_2" is an identifier not previously used, then

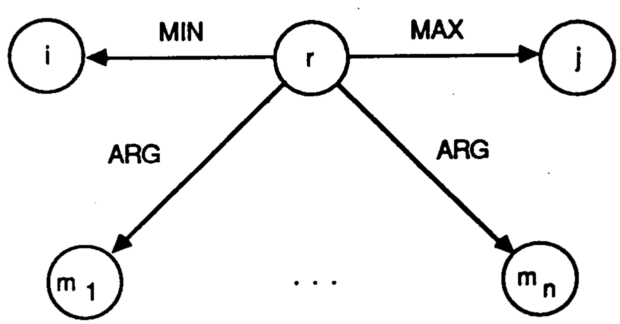


is a network and m_2 is a structured proposition node.

SI.9 m_2 is the Meinongian objective corresponding to the proposition that agent i believes proposition m_1.

Two special cases of SR.9 that are of interest concern *de re* and *de dicto* beliefs; they are illustrated in Figure 11-2 and Figure 11-3. [For details, see (Rapaport and Shapiro, 1984) and (Rapaport, 1984b), (Rapaport, 1986b).]

SR.10 If $m_1, \ldots, m_n$ are proposition nodes ($n \geqslant 0$), "i" and "j" are integers between 0 and n, inclusive, and "r" is an identifier not previously used, then



is a network, and r is a rule node.

SI.10 r is the Meinongian objective corresponding to the proposition that there is a relevant connection between

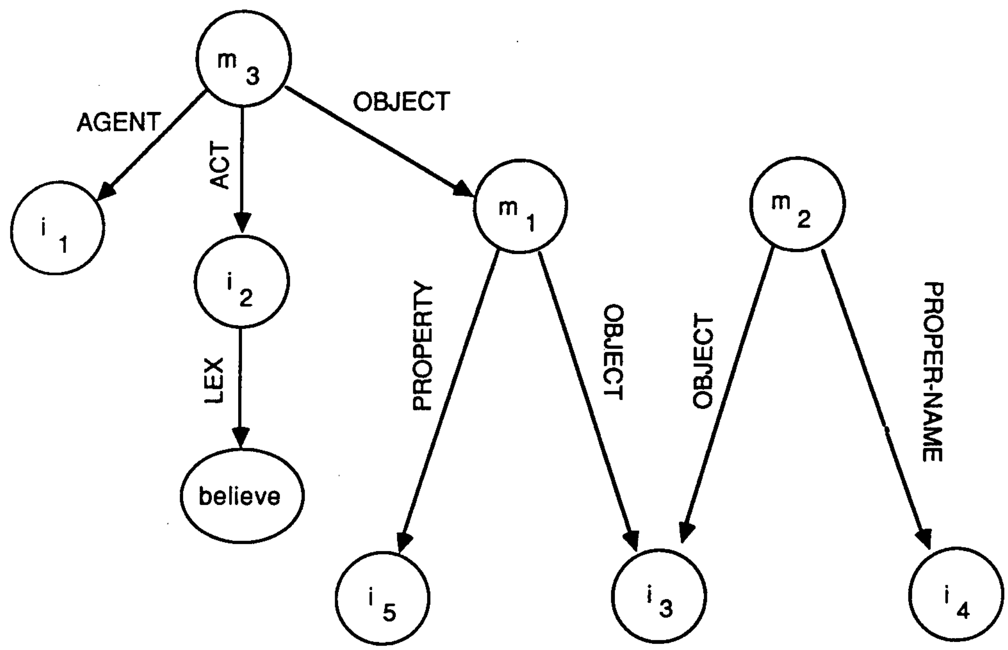


Figure 11-2: Meinongian Objective - *de re* Reading

m_3 is the Meinongian objective corresponding to the proposition that agent i_1 believes *de re* of objectum i_3 (who is believed by CASSIE to be named i_4) that it has the property i_5.

propositions $m_1, \ldots, m_n$ such that at least i and at most i(j) of them are simultanenously true.

Rule r of SR/SI.10 is called *AND-OR* and is a unified generalization of negation ($i = j = 0$), binary conjunction ($i = j = 2$), binary inclusive disjunction ($i = 1$, $j = 2$), binary exclusive disjunction ($i = 0$, $j = 1$), etc.

SR.11 If $m_1, \ldots, m_n$ are proposition nodes (n $\leqslant$ 0), is an integer between 0 and n, inclusive, and "r" is an identifier not previously used, then

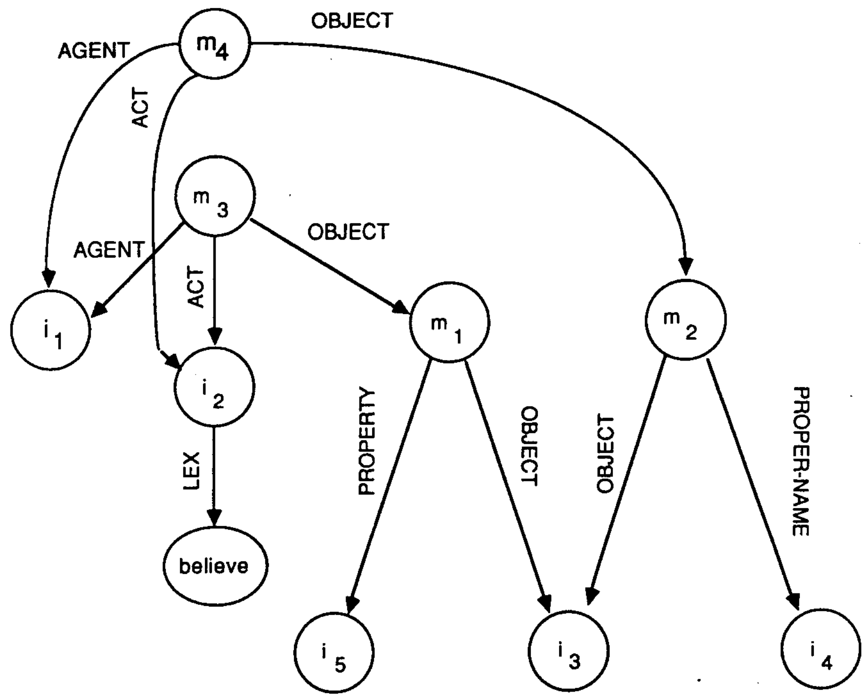
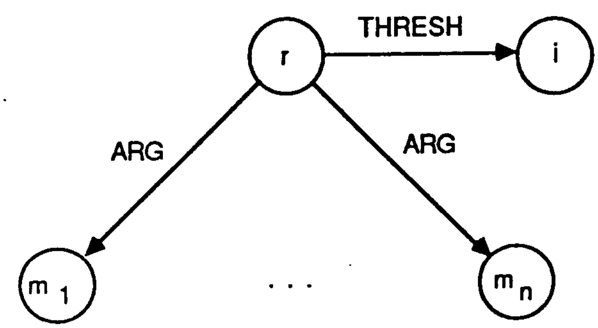


Figure 11-3: Meinongian Objective - *de dicto* Reading

m_4 is the Meinongian objective corresponding to the proposition that agent i_1 believes *de dicto* that objectum i_3 (who is believed by i_1 to be named i_4) has the property i_5.

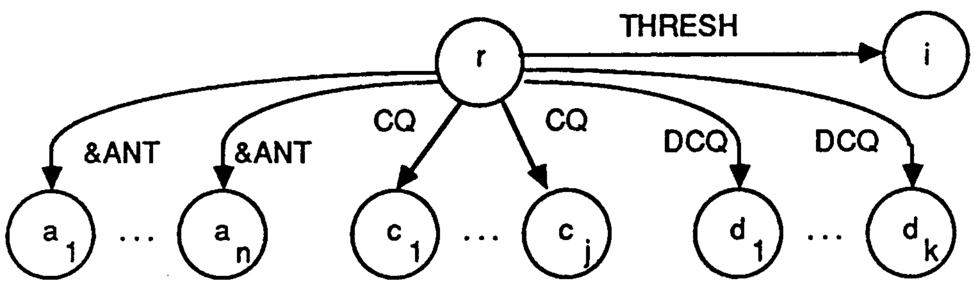


is a network, and r is a rule node.

SI.11 r is the Meinongian objective corresponding to the proposition that there is a relevant connection between propositions m_1, ... , m_n such that either fewer than i of them are true or they all are true.

Rule r of SR/SI.11 is called *THRESH* and is a generalization of the material biconditional (i = 1).

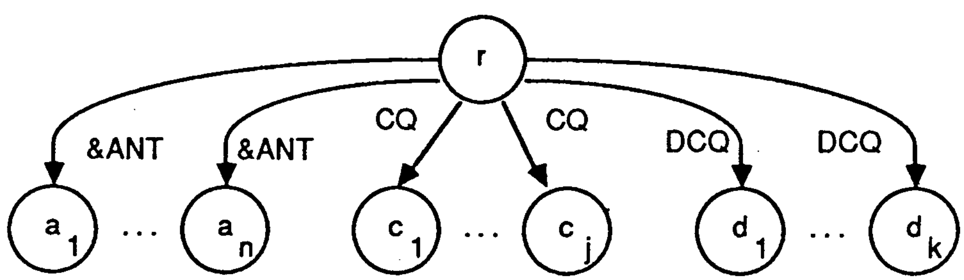
SR.12 If $a_1, \ldots, a_n$, $c_1, \ldots, c_j$, and $d_1, \ldots, d_k$ are proposition nodes $(n \geqslant 1;\ j, k \geqslant 0;\ j + k \geqslant 1)$, "$i$" is an integer between 1 and n, inclusive, and "r" is an identifier not previously used, then



is a network, and r is a rule node.

SI.12 r is the Meinongian objective corresponding to the proposition that the conjunction of any i of the propositions $a_1, \ldots, a_n$ relevantly implies each c_l $(1 \leqslant l \leqslant j)$ and relevantly implies each d_l $(1 \leqslant l \leqslant k)$ for which there is not a better reason to believe it is false.

SR.13 If $a_1, \ldots, a_n$, $c_1, \ldots, c_j$, and $d_1, \ldots, d_k$ are proposition nodes $(n, j, k \geqslant 0)$, and "r" is an identifier not previously used, then

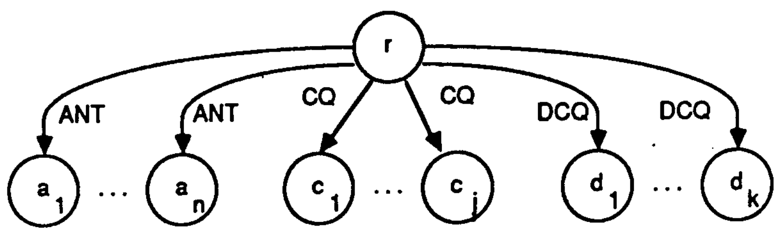


is a network, and r is a rule node.

SI.13 r is the Meinongian objective corresponding to the proposition that the conjunction of the propositions $a_1, \ldots, a_n$ relevantly implies each c_l $(1 \leqslant l \leqslant j)$ and relevantly implies each d_i $(1 \leqslant l \leqslant k)$ for which there is not a better reason to believe it is false.

The d_i are *default* consequences, in the sense that each is implied only if it is neither the case that CASSIE already believes *not* d_l nor that *not* d_i follows from non-default rules.

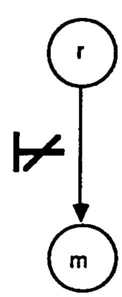
SR.14 If $a_1, \ldots, a_n, c_1, \ldots, c_j,$ *and* $d_1, \ldots, d_k$ are proposition nodes $(n \geqslant 1;\ j, k \geqslant 0;\ j + k \geqslant 1)$, and "$r$" is an identifier not previously used, then



is a network, and r is a rule node.

SI.14 r is the Meinongian objective corresponding to the proposition that any a_i, $1 \leqslant i \leqslant n$, relevantly implies each c_l $(1 \leqslant l \leqslant j)$ and relevantly implies each d_l $(1 \leqslant l \leqslant k)$ for which there is not a better reason to believe it is false.

SR.15 If m is a proposition node, and "r" is an identifier not previously used, then



is a network, and r is a rule node.

SI.15 r is the Meinongian objective corresponding to the proposition that there is no good reason for believing proposition m.

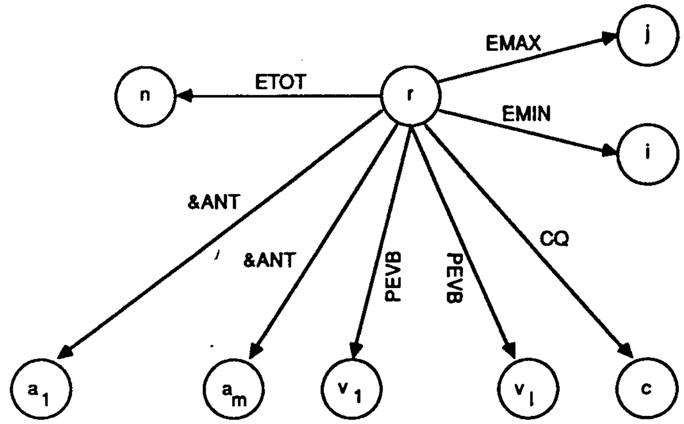
SR.16 If r is a rule node as specified by SR.10-SR.15, and r dominates variable nodes $v_1, \ldots, v_n$, and, in addition, arcs labeled "AVB" go from r to each v_i, then r is a quantified rule node.

SI.16 r is the Meinongian objective corresponding to the proposition that the rule that would be expressed by r without the AVB arcs holds after replacing each v_i by any Meinongian object in its range.

SR.17 If r is a rule node as specified by SR.10-SR.15, and r dominates variable nodes $v_1, \ldots, v_n$, and, in addition, arcs labeled "EVB" go from r to each v_i, then r is a quantified rule node.

SI.17 r is the Meinongian objective corresponding to the proposition that the rule that would be expressed by r without the EVB arcs holds after replacing each v_i by some Meinongian object in its range.

SR.18 If $a_1, \ldots, a_m$ and c are proposition nodes; $v_1, \ldots, v_l$ are variable nodes dominated by one or more of $a_1, \ldots, a_m, c$; "i", "j", and "n" are integers $(0 \leqslant i \leqslant j \leqslant mn)$; and "$r$" is an identifier not previously used; then



is a network, and r is a rule node.

SI.18 r is the Meinongian objective corresponding to the proposition that, of the n sequences of Meinongian objects which, when substituted for the sequence $v_1, \ldots, v_l$, make all the a_i believed propositions, between i and j of them also satisfy c. (For further details on such numerical quantifiers, see (Shapiro, 1979c).)

11.3.4. The conversation with CASSIE, revisited

In this section, we shall review the conversation we had with CASSIE, showing the network structure as it is built - that is, showing the structure of CASSIE's mind as she is given information and as she infers new information. (Comments are preceded by a dash.)

```
: Young Lucy petted a yellow dog
I understand that young Lucy petted a yellow dog

 - CASSIE is told something, which she now believes. Her
   entire belief  structure is shown in Figure 11-4 (a).
   The node labeled "now" represents the current time, so
   the petting is clearly  represented as being in the past.
   CASSIE's response is "I understand that" appended to her
   English description of the proposition just entered.


: What is yellow
a dog is yellow

 - This response shows that CASSIE actually has some
   beliefs; she  did not just parrot back the above
   sentence.

: Dogs are animals
I understand that dogs are animals

 - CASSIE is told a small section of a class hierarchy.

: Who petted an animal
young Lucy petted a yellow dog

 - CASSIE can answer the question using the class
   hierarchy, because, prior to the conversation,
   the inheritance rule
```

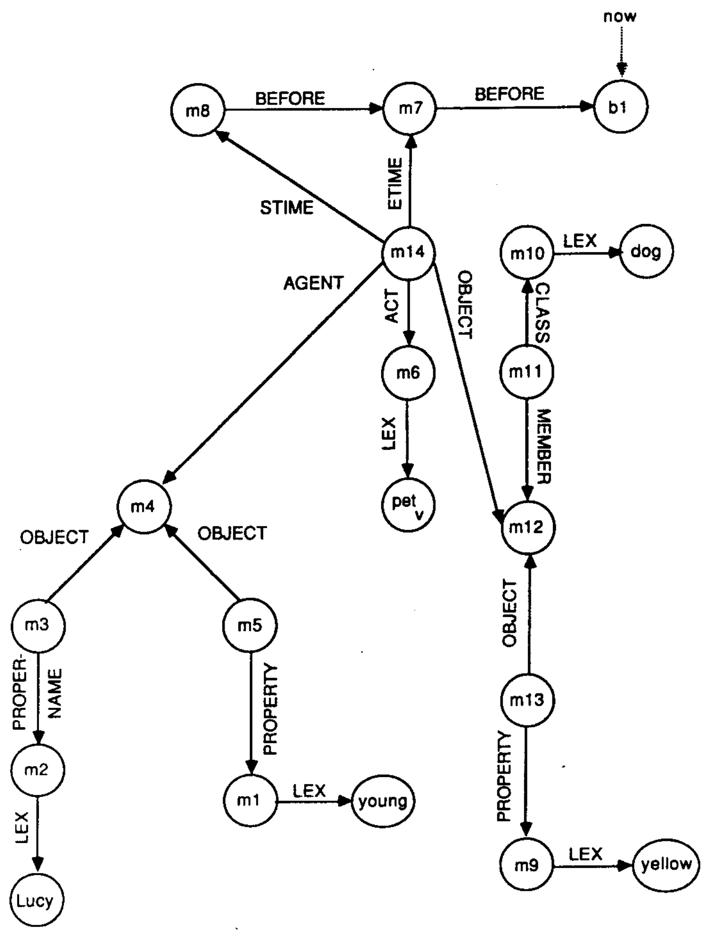


Figure 11-4: Fragment of CASSIE's Belief Structure

Fragment of CASSIE's belief structure after being told that young Lucy petted a yellow dog.

```
(def-path class (compose class (kstar
            (compose subclass- superclass))))
```

was given to SNePS. This rule says that the CLASS arc is implied by the path consisting of a CLASS arc followed by zero or more occurrences of the two-arc path consisting of the converse SUBCLASS arc followed by the SUPERCLASS arc [see (Shapiro, 1978), (Srihari, 1981)]. The dog was called "a yellow dog" rather than "a yellow animal" because the redundant CLASS arc is not built. Figure 11-5 shows the current state of CASSIE's belief structure about the dog's classification and color.

```
: Lucy is sweet
I understand that young Lucy is sweet
```

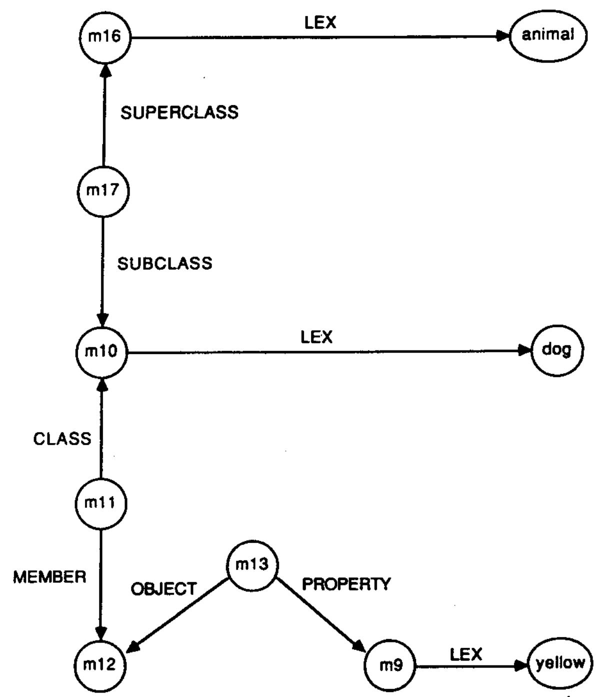


Figure 11-5: CASSIE's Belief Structure

CASSIE's belief structure about the dog's classification and color. (Node m12 represents the dog.)

- CASSIE's response shows that she identifies this Lucy with the previous Lucy.

```
: Lucy is a girl
I understand that sweet young Lucy is a girl
```

- The beginning of a class hierarchy for Lucy. Notice that all the adjectival properties of Lucy are mentioned.

```
: Girls are people
I understand that girls are people
```

- More of the class hierarchy is given.

```
: What was petted by a person
a yellow dog was petted by sweet young Lucy
```

- Again, the proposition is retrieved using the CLASS inheritance rule. The answer is expressed in the passive voice because of the way the question was asked.

```
: The dog is named Rover
I understand that Rover is a yellow dog
```

- 'the dog' refers to the only dog CASSIE knows about, who is now given a name.

: What was petted by a person
yellow Rover was petted by sweet young Lucy

- This is exactly the same question that was asked before. It is answered differently this time, because the dog now has a name, and CASSIE prefers to describe an individual by its name when it has one.

: John believes that Lucy is rich
I understand that John believes that Lucy is rich

- At this point in our development of CASSIE, she interprets 'believes that' contexts to be *de dicto*, so she assumes that the Lucy that John has in mind is a different one from the Lucy that she knows. Figure 11-6 shows CASSIE's beliefs about the two Lucies.

: Who is rich
I don't know.

- CASSIE knows no one who is rich. She only believes that *John* believes that someone (whom she believes that he believes to be named 'Lucy') is rich. The answer is 'I don't know', rather than 'no one is rich', because CASSIE doesn't use the closed-world hypothesis.

: Who is sweet
young Lucy is sweet

- This question is asked merely to demonstrate that Lucy is able to answer a "who is <property>" question when she has relevant beliefs.

: John believes that Lucy is old
I understand that John believes that rich Lucy is old

- Even though CASSIE assumes that John knows a different Lucy than she knows, she assumes that all John's beliefs about "Lucy" are about the same Lucy.

: John is a boy
I understand that John is a boy

- This and the next two inputs are given to establish more of the class hierarchy and

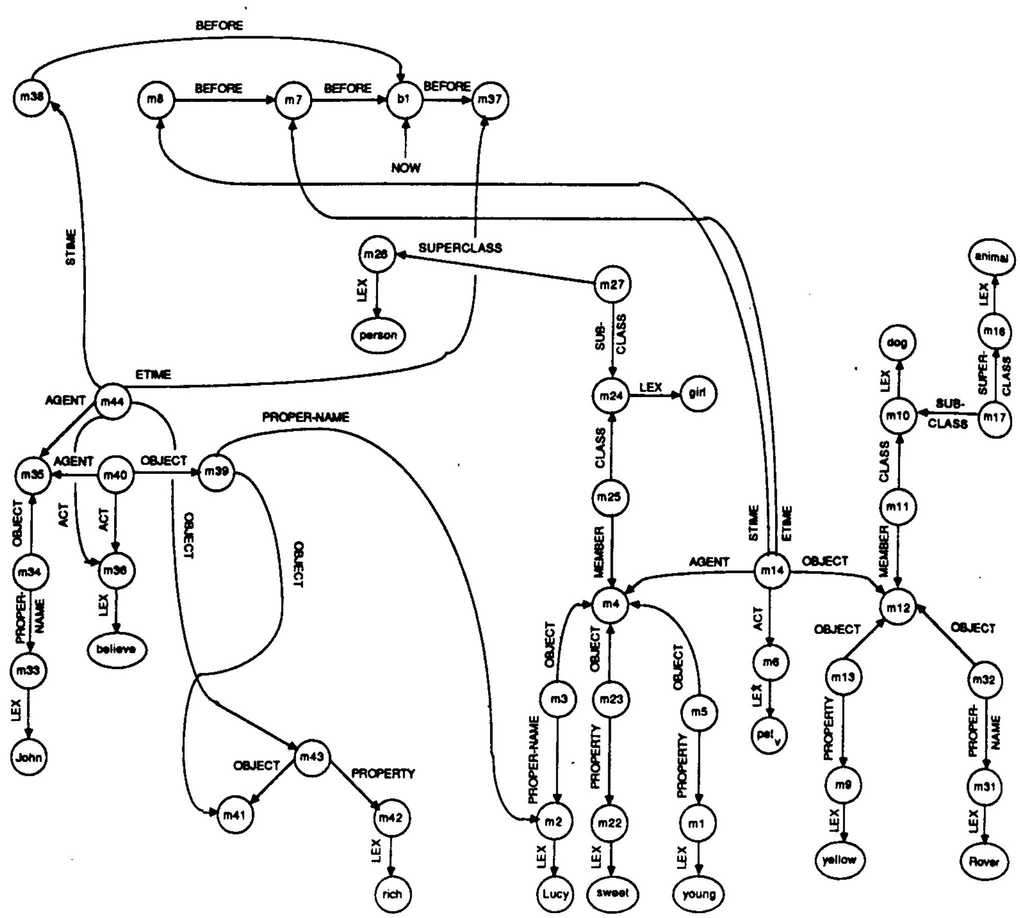


Figure 11-6: A Fragment of the Network

A Fragment of the network after CASSIE is told that John believes that Lucy is rich, showing CASSIE's beliefs about the two Lucies.

```
    to make it clear that when CASSIE answers
    the last question of this session, she is
    doing both path-based reasoning and node-
    based reasoning at the same time.

I understand that boys are people

: Dogs are pets
I understand that dogs are pets

: For every p and d if p is a person and d is a pet then p
     loves d
```

```
I understand that for every d and p, if p is a person and
   d is a pet

 - Figure 11-7 shows how this node-based
   rule fits into the class hierarchy. This is, we
   believe, equivalent to the integrated TBox/ABox
   mechanism proposed for KRYPTON
   [ (Brachman, Fikes, and Levesque, 1983),
 (Brachman, Gilbert, and Levesque, 1985)].
```

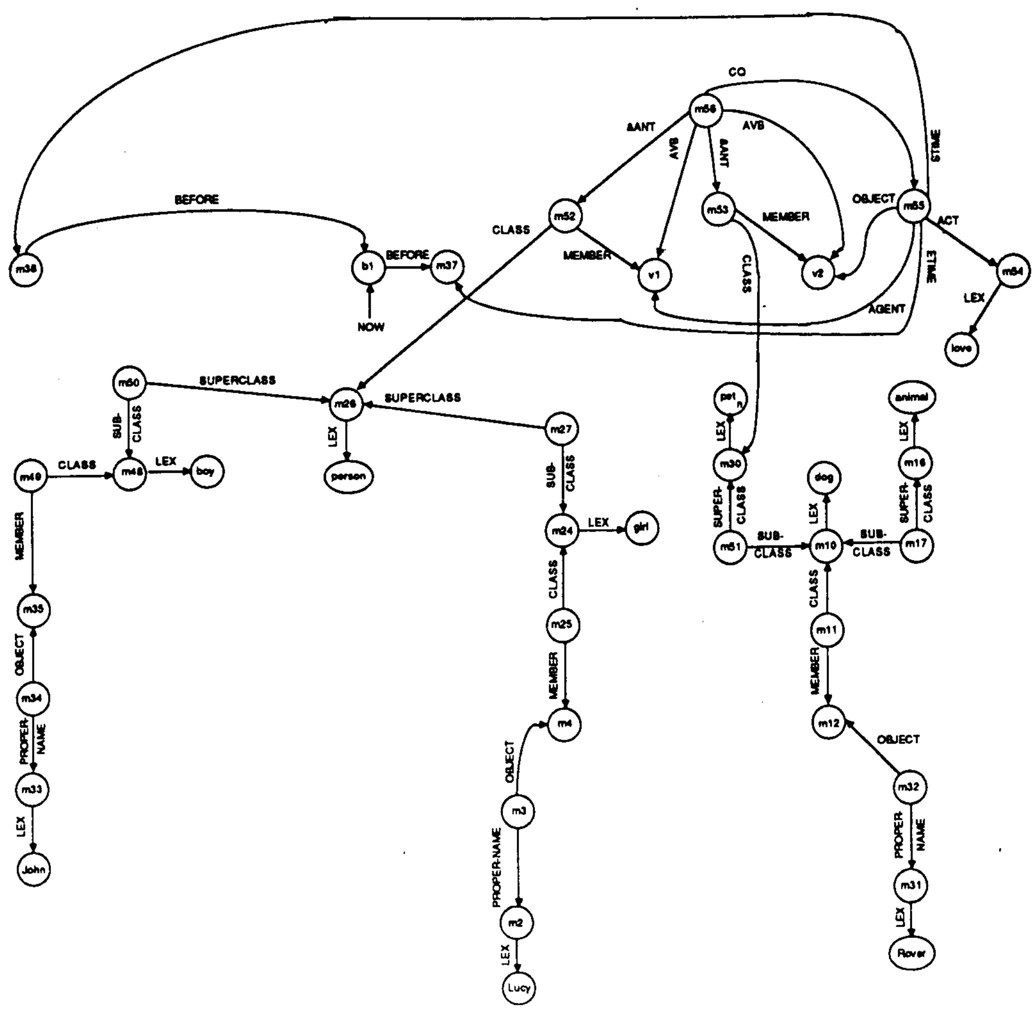


Figure 11-7: A Node-based Rule in a Class Hierarchy

```
: Who loves a pet
sweet young Lucy loves yellow Rover
and
```

John loves yellow Rover

- The question was answered using path-based inferencing to deduce that Lucy and John are people and that Rover is a pet, and node-based inferencing to conclude that, therefore, Lucy and John love Rover.

- The full network showing CASSIE's state of mind at the end of the conversation is given in Figure 11-8.

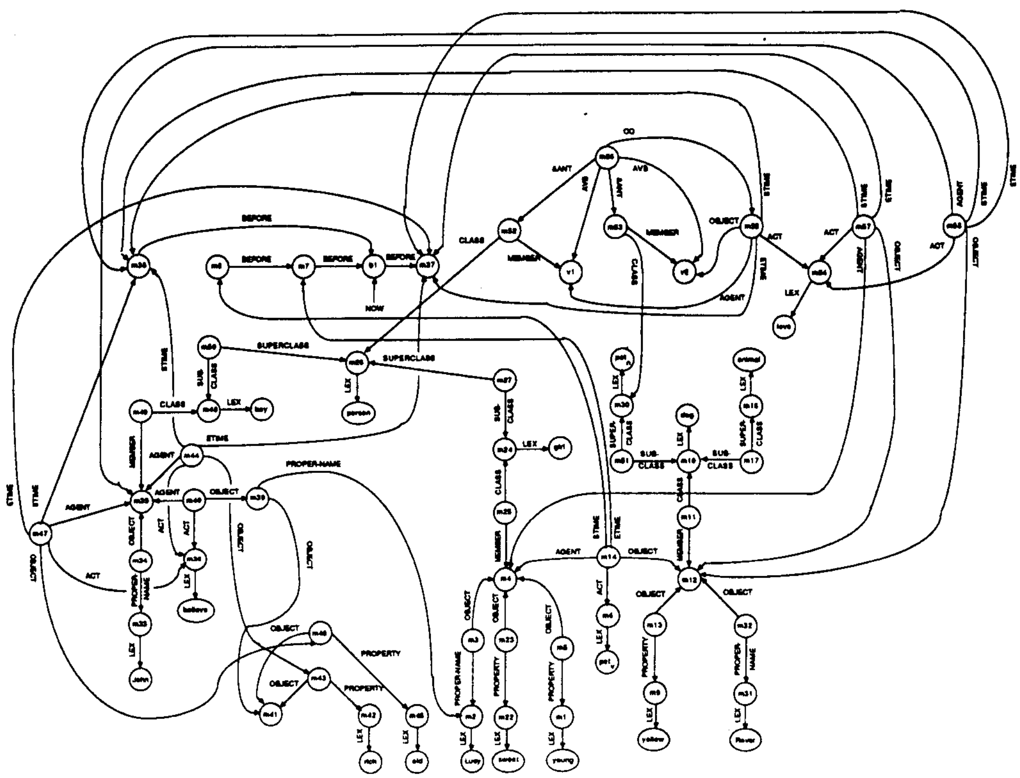


Figure 11-8: CASSIE's Beliefs at the End of the Conversation

11.4. Extensions and applications of SNePS

In this essay, we have been advocating the use and interpretation of SNePS networks to model (the beliefs of) a cognitive agent. SNePS, however, is of much wider and more general applicability. In this section, we give examples of recent and current research projects using SNePS in belief-revision, as a database management system, for developing several expert systems, and for representing temporal information in narratives. Even though most of these uses of SNePS do not explicitly involve a cognitive agent, it should be noted that in each case the asserted nodes can be treated as "beliefs" of the system: beliefs about the database, beliefs about the various domains of the expert systems, beliefs about linguistics, etc.

11.4.1. SNePS as a database management system

SNePS can be used as a network version of a relational database in which every element of the relational database is represented by an atomic node, each row of each relation is represented by a molecular node, and each column label (attribute) is represented by an arc label. Whenever a row *r* has an element *e* in column *c*, the molecular node representing *r* has an arc labeled *c* pointing to the atomic node representing *e*. Relations (tables) may be distinguished by either of two techniques, depending on the particular relations and attributes in the relational database. If each relation has an attribute that does not occur in any other relation, then the presence of an arc labeled with that attribute determines the relationship represented by the molecular node. A review of the syntax of the CASSIE networks will show that this technique is used there. The other technique is to give every molecular node an additional arc (perhaps labeled "RELATION") pointing to an atomic node whose identifier is the name of the relation. Table 11-1 shows the Supplier-Part-Project database of (Date, 1981), p 114). Notice that the SNAME and STATUS attributes only occur in the SUPPLIER relation; PNAME, COLOR, and WEIGHT only occur in the PART relation; JNAME only occurs in the PROJECT relation; and QTY only occurs in the SPJ relation. Figure 11-9 shows the SNePS network for part of this database.

Table 1: SUPPLIER

S#	SNAME	STATUS	CITY
s1	Smith	20	London
s2	Jones	10	Paris
s3	Blake	30	Paris
s4	Clark	20	London
s5	Adams	30	Athens

Table 2: PART

P#	PNAME	COLOR	WEIGHT	CITY
p1	nut	red	12	London
p2	bolt	green	17	Paris
p3	screw	blue	17	Rome
p4	screw	red	14	London
p5	cam	blue	12	Paris
p6	cog	red	19	London

Table 3: PROJECT

J#	JNAME	CITY
j1	sorter	Paris
j2	punch	Rome
j3	reader	Athens
j4	console	Athens
j5	collator	London
j6	terminal	Oslo
j7	tape	London

Table 4: SPJ

S#	P#	J#	QTY
s1	p1	j1	200
s1	p1	j4	700
s2	p3	j1	400
s2	p3	j2	200
s2	p3	j3	200
s2	p3	j4	500
s2	p3	j5	600
s2	p3	j6	400
s2	p3	j7	800
s2	p5	j2	100
s3	p3	j1	200
s3	p4	j2	500
s4	p6	j3	300
s4	p6	j7	300
s5	p2	j2	200
s5	p2	j4	100
s5	p5	j5	500
s5	p5	j7	100
s5	p6	j2	200
s5	p1	j4	1000
s5	p3	j4	1200
s5	p4	j4	800
s5	p5	j4	400
s5	p6	j4	500

Table 11-1: Tables **Supplier Part Project** and **SPJ**

Many database retrieval requests may be formulated using the **find** command of SNePSUL, the SNePS User's Language. The syntax of **find** is **(find r_1 n_1 ... r_m n_m)**, where r_i is either an arc or a path, and n_i is either a node or a set of nodes (possibly the value of a nested call to **find**). The value of a call to find is the set of all nodes in the network with an r_1 arc to any node in the set n_1, an r_2 arc to any node in

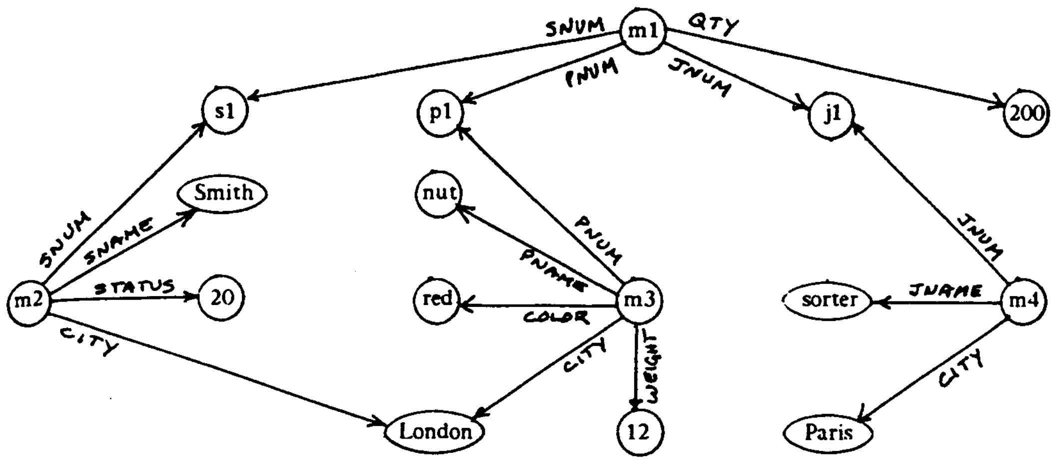


Figure 11-9: Fragment of SNePS Network for the Supplier-Part-Project Database.

the set n_2, ... , and an r_m arc to any node in the set n_m. Free variables are prefixed by "?". An infix '-' sign between **finds** represents the set difference operator.

The session below shows some of the queries from (Date, 1981); pp 141-142 translated into **find** commands, and the results on the database shown above. (In each interaction, comments are preceded by semicolons, user input follows the '*'-prompt, and SNePS responses are on succeeding lines. Execution and garbage collection times have been edited out for clarity.)

```
; Get full details of all projects in London.
* (dump (find jname ?x city London))
(m18 (city (London)) (jname (tape)) (jnum (j7)))
(m16 (city (London)) (jname (collator)) (jnum (j5)))
(dumped)

; Get SNUM values for suppliers who supply project J1
; with part P1.
* (find snum- (find jnum j1 pnum p1))
(s1)

; Get JNAME values for projects supplied by supplier S1.
```

```
* (find (jname- jnum jnum- snum) s1)
(console sorter)

; Get S# values for suppliers who supply both projects
; J1 and J2.
* (find (snum- jnum) j1 (snum- jnum) j2)
(s3 s2)

; Get the names of the suppliers who supply project J1
; with a red part.
* (find (sname- snum snum-)  (find jnum j1 (pnum pnum-
      color) red))
(Smith)

; Get S# values for suppliers who supply a London or Paris
;       project with a red part.
* (find snum- (find (jnum jnum- city) (London Paris)
                    (pnum pnum- color) red))
(s4 s1)

; Get P# values for parts supplied to any project by
;       a supplier in the same city.
* (find pnum- (find (jnum jnum- city) ?city (snum snum-
      city) ?city))
(p5 p4 p1 p2 p6 p3)

; Get J# values for projects not supplied with any red part
;       by any London supplier.
* ((find jnum- ?x)-(find jnum- (find (pnum pnum- color) red
                               (snum snum- city) London)))
(j6 j5 j2)

, Get S# values for suppliers supplying at least one part
; supplied by at least one supplier who supplies at least
; one red part.
* (find (snum- pnum pnum- snum snum- pnum pnum- color) red)
(s3 s4 s2 s5 s1)

; Get J# values for projects which use only parts which are
;      available from supplier S1.
* ((find jnum- (find qty ?q))
   - (find (jnum- pnum) (find pnum- ?r) - (find (pnum- snum)
      s1)))
nil
```

11.4.2. Address recognition for mail sorting

A research group led by Sargur N. Srihari is studying address recognition techniques for automated mail sorting (Srihari, Sargur, Jonathan, Palumbo, Niyogi, and Wang, 1985).

Computer determination of the sort-destination of an arbitrary piece of letter-mail from its visual image is a problem that remains far from solved. It involves overcoming several sources of ambiguity at both the spatio-visual and linguistic levels: The location of the destination address has to be determined in the presence of other text and graphics; relevant address lines have to be isolated when there are irrelevant lines of text in the address block; the iconic shapes of characters have to be classified into words of text when numerous types of fonts, sizes, and printing media are present; and the recognized words have to be verified as having the syntax and semantics of an address.

Spatial relationships between objects are essential knowledge sources for vision systems. This source extends naturally to the postal-image understanding problem, because of strong directional expectations. For example, the postage mark is usually above and to the right of the destination address, and the return address is usually to the left of the postage. A semantic network is a natural representation for geometric relations.

An envelope image is segmented into blocks, and a SNePS network is built that represents the geometric relations between blocks and information about the relative and absolute area occupied by each block. A preliminary set of geometric relations are the eight compass points. Relative area occupancy is expressed as the percentage of each block that falls in each of nine equal rectangular subdivisions of the envelope image, and absolute area is given in terms of the number of pixels covered by each block. The program constructs an exhaustive representation of all the geometric relations present in the image. Given the image produced by an initial segmentation procedure, a rough, intuitive output, shown in Figure 11-10 with some arc labels removed for clarity) was produced.

Future work in this area includes refinement of the data structure to represent more information more efficiently and the addition of inferencing capabilities whose objective is to present the control structure with tentative decisions about the address block based only on the information provided by the initial segmentation.

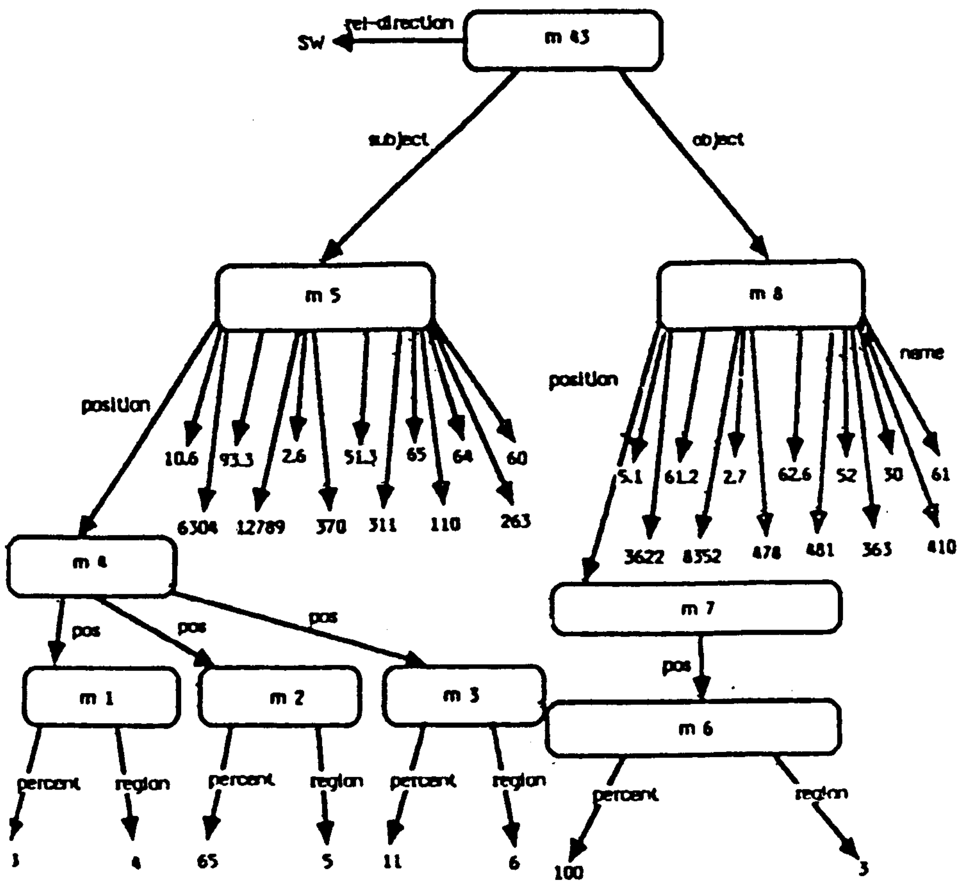


Figure 11-10: SNePS Network Representation of Initial Segmentation of Envelope Image (from Srihari, Hull et al. 1985)

11.4.3. NEUREX

The NEUREX project (Cheng, 1984), (Xiang and Srihari, 1985), (Xiang, Srihari, Shapiro and Chutkow, 984"), (Suchin, 1985) is a diagnostic expert system for diseases of the central and peripheral nervous systems; it also deals with information about neuroaffectors, neuroreceptors, and body parts. SNePS is used to represent spatial structures and functions propositionally. Entities are represented topologically by means of proposition nodes expressing an entity's shape, position, etc., and spatial relations are represented by proposition nodes expressing adjacency, connectivity, direction, etc. This approach integrates structural and functional neuroanatomical information. Moreover, the representation is both propositional and analog. For the peripheral nervous system, there are nodes representing such propositions as that, for example, a sequence of nerve segments are linked at junctions, and that the whole sequence forms a (peripheral) nerve; the network that is built is itself an analog representation of this nerve (and ultimately, together with its neighbors, of the entire peripheral nervous system). See Chapter 15 for further discussion of analog representations. For the central nervous system, there are coordinates in the network representation that can be used to support reasoning by geometrical computation or graphical interfaces.

As one example, the network of Figure 11-11 can be used by the system to determine which muscles are involved in shoulder-joint flexion, using the SNePS User Language request

```
(find (ms- cn) (find jt shoulder-joint mv flexion)),
```

which returns the following list of four nodes:

```
(deltoid pectoralis_major_clavicular_head
coracobrachialis biceps_brachii)
```

Furthermore, rules, like that shown in Figure 11-12, can be employed and can even include probabilistic information. (Note that node *r* in Figure 11-12 is the SNePS implementation of the IF-THEN rule; cf. (SR.13).)

11.4.4. Representing visual knowledge

The goal of the Versatile Maintenance Expert System (VMES) project is to develop an expert maintenance system that can reason about digital circuits represented graphically (cf. (Shapiro, Srihari, Geller, and Taie, 1986;SSTG86)). A similar perspective on the need for visual knowledge representation is taken by Tsotsos and Shibahara (Chapter 10) and Havens and Mackworth (Chapter 16). The representation is not pixel-oriented; this is a project in visual knowledge representation integrated with more traditional conceptual and propositional knowledge representation. The graphical form of

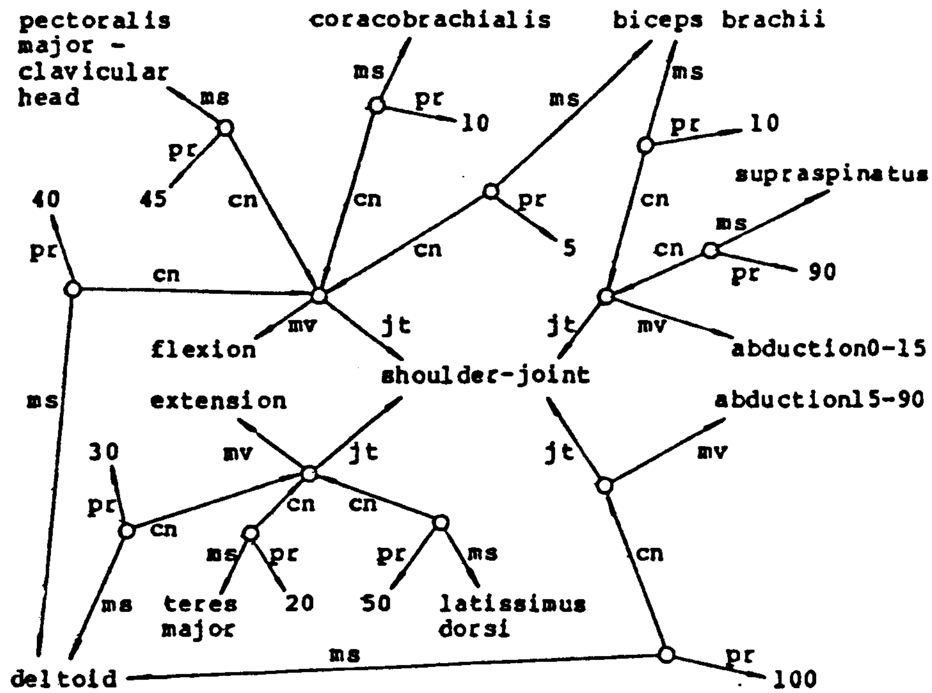


Figure 11-11: Four of the Shoulder-Joint Movements

Four of the shoulder-joint movements with musscles involved and their contribution to each relevant movement. (Meaning of the arc labels: jt=joint; mv=movement; ms=muscle; cn=contribute; pr=percentage.) (From Xiang and Srihari 1985)

an object is a LISP function that, when evaluated, draws the object on the screen. Propositional nodes express information about (1) the relative or absolute position of the object and (2) attributes of the object. Visual knowledge can also be distributed among nodes in traditional hierarchies: for example, the knowledge of how to display a particular hammer may be stored at the level of the class of hammers; the knowledge of how to display a person may be distributed among the nodes for heads, arms, etc.

For example, Figure 11-13 shows a set of three assertions. Node m233 represents the assertion that the object TRIANGLE-1 is 100 units to the right and 20 units below the object SQUARE-1. The MODALITY arc permits the selection of different modes of display; here, we want to display TRIANGLE-1 in "functional" mode. Node m220 states that every member of the class TRIANGLE displayed in functional mode has the form DTRIANG associated with it. Finally, node m219 asserts that TRIANGLE-1 is a TRIANGLE.

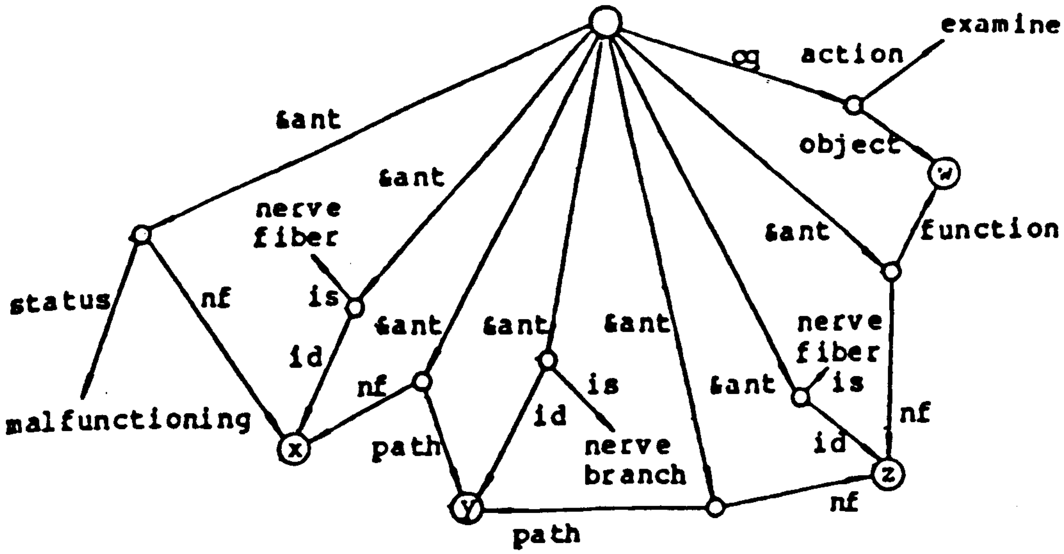


Figure 11-12: SNePS Network for a NEUREX Rule. .
(From Xiang and Srihari 1985)

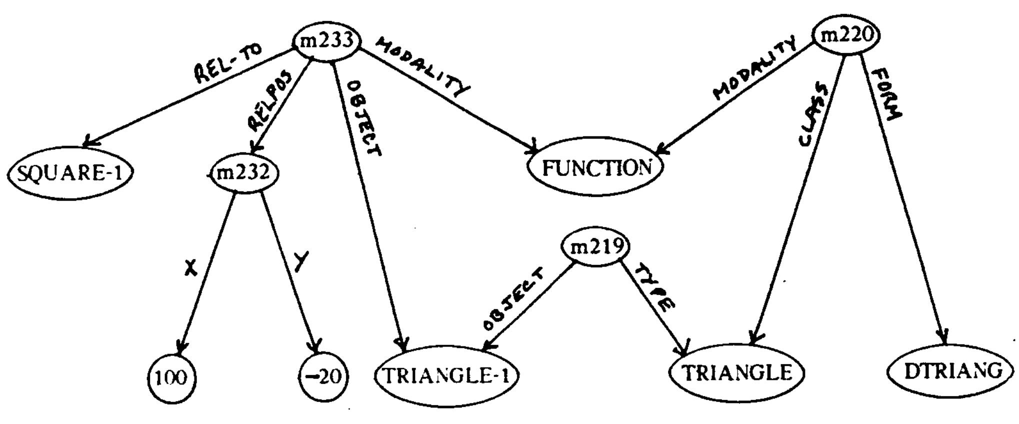


Figure 11-13: SNePS Network in VMES for the Form and Relative Position of TRIANGLE-1.

Figure 11-14 contains four assertions, of which node m246 is the most complex. It links the object GATE-1 to an absolute position at 100/400 and to the class of all AND-gates. Node m244 asserts that GATE-1 is a part of BOARD-1. Node m248 asserts that INP1-GATE1 is a PART-OF GATE-1 and belongs to the class AINP1. The label 'PART' actually stands for "has part". Node m239 links the attribute BAD to GATE-1. Every attribute belongs to an attribute class, and the arc ATTRIBUTE-CLASS points to the class STATE.

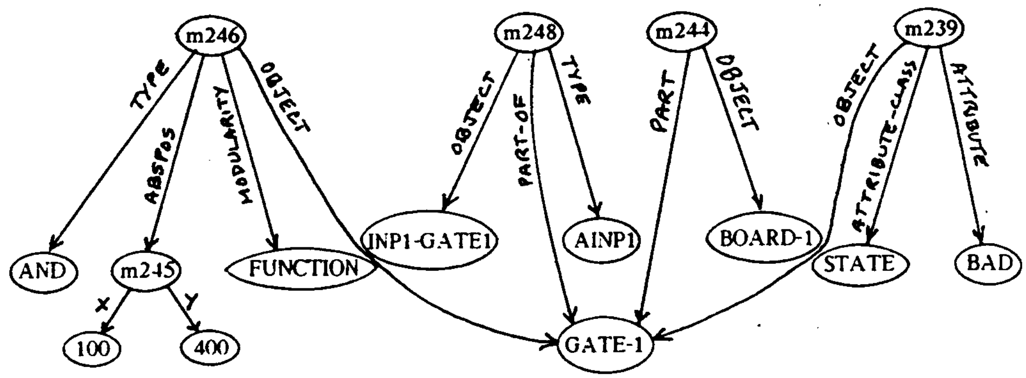


Figure 11-14: SNePS Network in VMES for the Location, Structure, and State of GATE-1.

11.4.5. SNeBR: A belief revision package

The SNePS Inference Package has been extended by João Martins to handle belief revision - an area of AI research concerned with the issues of revising sets of beliefs when a contradiction is found in a reasoning system. Research topics in belief revision include the study of the representation of beliefs, in particular how to represent the notion of belief dependence; the development of methods for selecting the subset of beliefs responsible for contradictions; and the development of techniques to remove some subset of beliefs from the original set of beliefs. (For an overview of the field, see (Martins, 1987).)

SNeBR (*SNe*PS *B*elief *R*evision) is an implementation in SNePS of an abstract belief revision system called the Multiple Belief Reasoner (MBR), which, in turn, is based on a relevance logic system called SWM (after Shapiro, Wand, and Martins)

(Shapiro and Wand, 1976), (Martins, 1983b), (Martins, 1983a), (Martins and Shapiro, 1984), (Martins and Shapiro, 1986a), (Martins and Shapiro, 1986b), (Martins and Shapiro, 1986c). SWM contains the rules of inference of MBR and defines how contradictions are handled. The only aspect of SWM relevant to this description concerns the objects with which MBR deals, called *supported wffs*. They are of the form

$$A \mid t, o, r$$

where A is a well-formed formula representing a proposition, t is an *origin tag* indicating how A was obtained (for example, as a hypothesis or as a derived proposition), o is an *origin set* containing *all* and *only* the hypotheses used to derive A, and r is a *restriction set* containing information about contradictions *known* to involve the hypotheses in o. The triple t, o, r is called the *support* of the wff A. The origin tag, origin set, and restriction set of a wff are computed when the wff is derived, and its restriction set may be updated when contradictions are discovered.

MBR uses the concepts of context and belief space. A *context* is any set of hypotheses. A context determines a *belief space*, which is the set of all the hypotheses defining the context together with all propositions derived exclusively from them. The propositions in the belief space defined by a given context are characterized by having an origin set that is contained in the context. At any point, the set of all hypotheses under consideration is called the *current context*, which defines the *current belief space*. The only propositions that are retrievable at a given time are the ones belonging to the current belief space.

A contradiction may be detected either because an assertion is derived that is the negation of an assertion already in the network, or because believed assertions invalidate a rule being used (particularly an AND-OR or a THRESH rule; see (SR/SI.10-11)). In the former case, the contradiction is noted when the new, contradictory, assertion is about to be built into the network, since the Uniqueness Principle guarantees that the contradictory assertions will share network structure. In the latter case, the contradiction is noted in the course of applying the rule. In the former case, it may be that the contradictory assertions are in different belief spaces (only the new one being in the current belief space). If so, the restriction sets are updated to reflect the contradictory sets of hypotheses, and nothing else happens. If the contradictory assertions are both in the current belief space (which will be the case when one of them is a rule being used), then, besides updating the restriction sets, the user will be asked to delete at least one of the hypotheses underlying the contradiction from the current context. Management of origin sets according to SWM guarantees that, as long as the current context was originally not known to be contradictory, removal of any one of the hypotheses in the union of the origin sets of the contradictory assertions from the current context will restore the current context to the state of not being known to be inconsistent.

11.5. Knowledge-based natural language understanding

Jeannette Neal has developed an AI system that can treat knowledge of its own language as its discourse domain, (Neal, 1985). The system's linguistic knowledge is represented declaratively in its network knowledge base in such a way that it can be used in the dual role of "program" to analyze language input to the system and "data" to be queried or reasoned about. Since language forms (part of) its domain of discourse, the system is also able to learn from the discourse by being given instruction in the processing and understanding of language. As the system's language knowledge is expanded beyond a primitive kernel language, instructions can be expressed in an increasingly sophisticated subset of the language being taught. Thus, the system's language is used as its own metalanguage.

The kernel language consists of a relatively small collection of predefined terms and rewrite rules for expressing syntax and for expressing the mapping of surface strings to the representation of their interpretations.

The knowledge representations include representations for surface strings and for relations such as: (a) a lexeme being a member of a certain lexical category, (b) bounded string B being in category C and this phrase structure being represented by concept N, (c) a structure or parsed string expressing a certain concept, and (d) one phrase structure being a constituent of another structure.

In order to talk about both the syntax and semantics of language, the network representations distinguish between a word or string and its interpretation. In one experiment, the statements

```
(1)    A WOMAN IS A HUMAN
(2)    'WOMAN' IS SINGULAR
```

were input to the system. The first makes a claim about women; the second makes a claim about the *word* 'woman'. Nodes m40 and m50 of Figure 11-15, respectively, represent the propositions expressed by these statements. The concept or class expressed by 'WOMAN' is represented by node b22; the entity represented by node b22 is a participant in the subset-superset proposition expressed by (1). However, in the representation of (2), the word 'WOMAN' itself is the entity having the property SINGULAR.

Additional statements, such as:

```
(R) IF THE HEAD-NOUN OF A NOUN-PHRASE X
    HAS NUMBER Y, THEN X HAS NUMBER Y.
```

were input to the system to demonstrate the use of a subset of English as its own metalanguage in building up the system's language ability from its primitive predefined language. Figure 11-16 illustrates the representation of the system's interpretation of rule (R) as well as the representation of certain linguistic relations. Node m87 represents the proposition that some bounded string represented by variable node v4 is in the category HEAD-NOUN, and this phrase structure is represented by variable node v3. Node m88 represents that the phrase structure represented by node v3 is a constituent of v1, which represents a NOUN-PHRASE structure. (In this figure, the AVB arcs have been eliminated for clarity; cf. (SR/SI.16).) As soon as any rule such as (R) is parsed and interpreted, it is immediately available for use in subsequent processing. Thus, the system is continuously educable and can use its language as its own metalanguage.

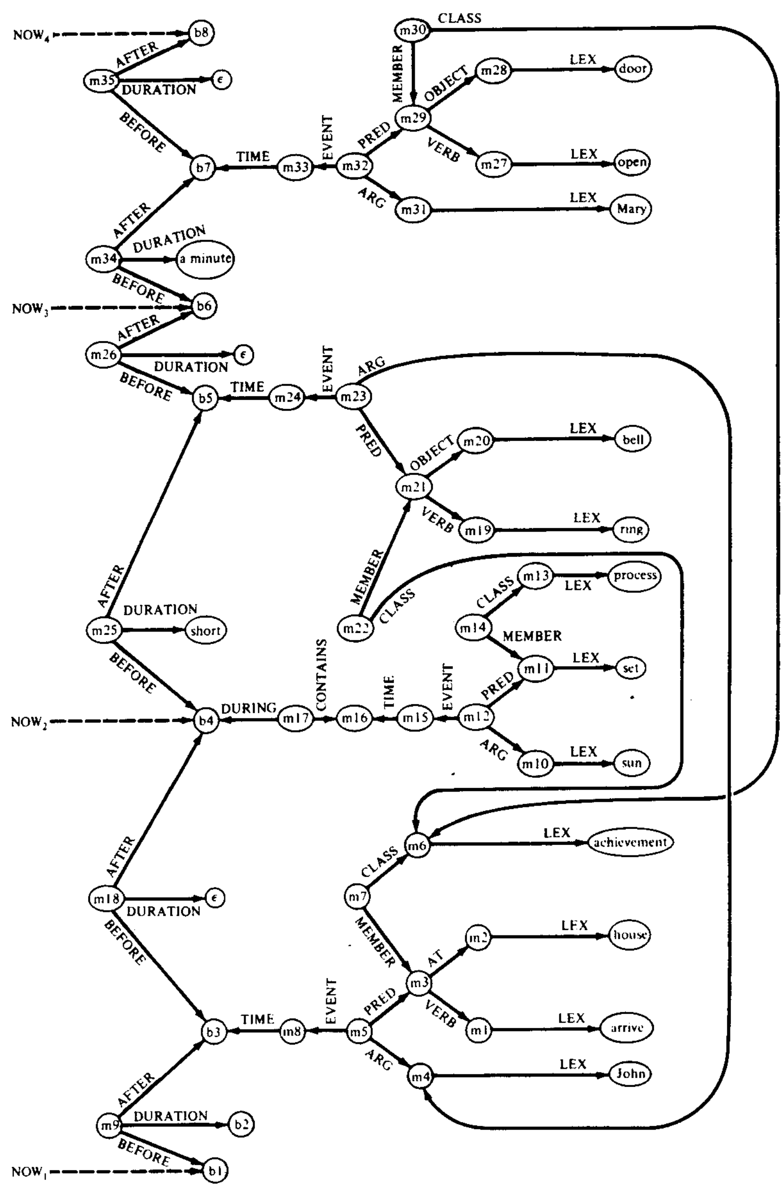


Figure 11-17: SNePS Network for a Short Narrative.

Subscripts are used in the figure to show the successive values of NOW.

The BEFORE-AFTER-DURATION case frame is used to indicate that the period of time pointed to by the BEFORE-arc temporally precedes the period of time pointed to by the AFTER-arc by the length of time pointed to by the DURATION-arc. These durations are usually not known precisely. The value <epsilon> stands for a very short interval; whenever an event occurs in the narrative line, it has the effect of moving NOW an interval of <epsilon> beyond it.

The DURING-CONTAINS case frame is used to indicate that the period of time pointed to by the DURING-arc is during (or contained in) the period of time pointed to by the CONTAINS-arc. Notice that the progressive sentence, "The sun was setting", created an event that contains the then-current NOW. If the system knows about such things as sunsets, then it should infer that the event of the sun's setting also contains John's arrival, his ringing of the bell, and probably also Mary's opening of the door.

11.6. Conclusion: SNePS and SNePS/CASSIE as Semantic Networks

We shall conclude by looking at SNePS and SNePS/CASSIE from the perspective of Brachman's discussions of structured inheritance networks such as KL-One and hierarchies of semantic network formalisms (Brachman, 1977, Brachman, 1979).

11.6.1. Criteria for semantic networks

Brachman offers six criteria for semantic networks:

A semantic network must have a *uniform notation*. SNePS provides some uniform notation with its built-in arc labels for rules, and it provides a uniform procedure for users to choose their own notation.

A semantic network must have an *algorithm for encoding information*. This is provided for by the interfaces to SNePS, for example, by the parser component of our ATN parser-generator that takes English sentences as input and produces SNePS networks as output.

A semantic network must have an *"assimilation" mechanism* for building new information in terms of stored information. SNePS provides for this by the Uniqueness Principle, which enforces node sharing during network building. The assimilation is demonstrated by the generator component of our ATN parser-generator, which takes SNePS nodes as input and produces English output expressing those nodes: Our conversation with CASSIE illustrated this the node built to represent the new fact, 'Lucy is sweet', is expressed in terms of the already existing node for Lucy (who had previously been described as young) by 'young Lucy is sweet'.

A semantic network should be *neutral* with respect to network formalisms at higher levels in the Brachman hierarchy. SNePS is a semantic network at the "logical" level, whereas SNePS/CASSIE is at the "conceptual" level. SNePS is neutral in the relevant sense; it is not so clear whether SNePS/CASSIE is. But neutrality at higher levels may not be so important; a more important issue is the reasons why one formalism should be chosen over another. Several possible criteria that a researcher might consider are: *efficiency* (including the ease of interfacing with other modules; for example, our ATN parser-generator has been designed for direct interfacing with SNePS), *psychological adequacy* (irrelevant for SNePS, but precisely what SNePS/CASSIE is being designed for), *ontological adequacy* (irrelevant for SNePS/CASSIE-see below), *logical adequacy* (guaranteed for SNePS because of its inference package), and *natural language adequacy* (a feature of SNePS's interface with the ATN grammar).

A semantic network should be *adequate* for any higher-level network formalism. SNePS meets this nicely: KL-One can be implemented in SNePS (Tranch, 1982).

A semantic network should have a *semantics*. We presented that in Section 11.3. But it should be observed that there are at least two very different sorts of semantics. In SNePS, nodes have a meaning *within the system* in terms of their links to other nodes; they have a meaning *for users* as provided by the nodes at the heads of LEX arcs. Arcs, on the other hand, only have meaning within the system, provided by node- and path-based inference rules (which can be thought of as procedures that operate on the arcs). In both cases, there is an "internal", system semantics that is holistic and structural: the meaning of the nodes and arcs are not given in isolation, but in terms of the entire network. This sort of "syntactic" semantics differs from a semantics that provides links to an external interpreting system, such as a user or the "world" - that is, links between the network's way of representing information and the user's way. It is the latter sort of semantics that we provided for SNePS/CASSIE with respect to an ontology of Meinongian objects.

11.6.2. SNePS and SNePS/CASSIE vs. KL-One

SNePS and SNePS/CASSIE can be compared directly to KL-One. Unlike KL-One, which is an *inheritance*-network formalism for representing concepts, instances of concepts, and properties and relations among them, SNePS is a *propositional*-network formalism for representing propositions and their constituents (individuals, properties, and relations).

Nevertheless, SNePS can handle inheritance. We have already seen an example of inheritance by *path*-based inference in the conversation with CASSIE. In that example, inheritance could also have been accomplished through node-based inference by, for example, representing 'dogs are animals' as a universally-quantified rule rather than by a SUBCLASS-SUPERCLASS case frame. That is, where an inheritance network might express the claim that dogs are animals by a single arc (say, a subclass-arc) from a dog-node to an animal-node, SNePS could express it by a proposition (represented by node m17 in Figure 11-5.).

One advantage of the propositional mode of representation and, consequently, of the second, or *rule*-based, form of property inheritance is that the proposition (m17) expressing the relationship can then become the objective of a proposition representing an agent's belief or it can become the antecedent or consequent of a node-based rule. In some inheritance networks, this could only be done by choosing to represent the entire claim by either the dog-node, the animal-node, the subclass-arc, or (perhaps) the entire structure consisting of the two nodes and the arc. The first two options seem incorrect; the third and fourth either introduce an anomaly into the representation (since arcs can then point either to nodes or to other arcs or to structures), or it reduces to what SNePS does: SNePS, in effect, trades in the single arc for a node with two outgoing arcs. In this way, the arcs of inheritance networks become information-bearing nodes, and the semantic network system becomes a propositional one.

Second, KL-One uses "epistemologically primitive links".

But why does KL-One use the particular set of links that it does, and not some other set; that is, what is the ontological justification for KL-One's links? There have been many philosophical and logical theories of the relations of the One to the Many (part-whole, member-set-superset, instance-concept, individual-species-genus, object-Platonic Form, etc.). KL-One's only motivation seems to be as a computationally efficient theory that clarifies the nature of inheritance networks; but it does not pretend to ontological or psychological adequacy. Indeed, it raises almost as many questions as it hopes to answer. For example, in KL-One, instances of a general concept seem to consist of *instances* of the attributes of the general concept, each of which instances have *instances* of the values of those attributes. But this begs important philosophical questions about the relations between properties of concepts (or of Forms, or of ...) and properties of individuals falling under those concepts (or participating in those Forms, or ...; some of these issues are discussed in (Brachman, 1983), but not from a philosophical point of view): Are they the same properties? Are the latter "instances" of the former? Are there such things as concepts (or Forms, or ...) of properties? And do instance nodes represent individuals? Do they represent individual concepts? [cf. (Brachman, 1977): 148.]

Now, on the one hand, SNePS/CASSIE's arcs are also taken to be "primitive"; but they are justified by the Meinongian philosophy of mind briefly sketched out above and explored in depth in the references cited. On the other hand, SNePS's arcs, by contrast to both SNePS/CASSIE's and KL-One's, are not restricted to any particular set of primitives. We believe that the interpretation of a particular use of SNePS depends on the *user's* world-view; the user should not be required to conform to *ours*.

And, unlike KL-One, the entities in the ontology for SNePS/CASSIE are not to be taken as representing things in the world: SNePS/CASSIE's ontology is an *epistemological ontology* [cf. (Rapaport, 1985a), (Rapaport, 1985b), (Rapaport, 1986a)] consisting of the purely intensional items that enable a cognitive agent to have beliefs (about the world). An epistemological ontology is a theory of what there must be in order for a cognitive agent to have beliefs (about what there is).

SNePS CONSIDERED AS A FULLY INTENSIONAL PROPOSITIONAL SEMANTIC NETWORK

(The Knowledge Frontier: Essays in the Representation of Knowledge,
edited by Nick Cercone and Gordon McCalla,
New York: Springer-Verlag, 1987, pages 262-315)

Stuart C. Shapiro and William J. Rapaport

Department of Computer Science
State University of New York at Buffalo
Buffalo, NY 14260
shapiro@cs.buffalo.edu, rapaport@cs.buffalo.edu

REFERENCES

(1) Almeida, Michael J. (1987), "Reasoning about the Temporal Structure of Narratives," Technical Report No. 87-10 (Buffalo, NY: SUNY Buffalo Dept. of Computer Science).

(2) Almeida, Michael J., & Shapiro, Stuart C. (1983), "Reasoning about the Temporal Structure of Narrative Texts," *Proceedings of the Fifth Annual Conference of the Cognitive Science Society (University of Rochester)* (Hillsdale, NJ: Lawrence Erlbaum Associates).

(3) Brachman, Ronald J. (1977), "What's in a concept: structural foundations for semantic networks," *International Journal of Man-Machine Studies* 9: 127-152.

(4) Brachman, Ronald J. (1979), "On the Epistemological Status of Semantic Networks," in Findler 1979: 3-50; reprinted in Brachman & Levesque 1985: 191-215.

(5) Brachman, Ronald J. (1983), "What IS-A Is and Isn't: An Analysis of Taxonomic Links in Semantic Networks," *Computer* 16.10 (October 1983) 30-36.

(6) Brachman, Ronald J.; Fikes, Richard E.; & Levesque, Hector J. (1983), "KRYPTON: Integrating Terminology and Assertion," *Proceedings of the National Conference on Artificial Intelligence (AAAI-83; Washington, DC)* (Los Altos, CA: Morgan Kaufmann): 31-35.

(7) Brachman, Ronald J.; Gilbert, Victoria P.; & Levesque, Hector J. (1985), "An Essential Hybrid Reasoning System: Knowledge and Symbol Level Accounts of KRYPTON," *Proceedings of the Ninth International Joint Conference on Artificial Intelligence (IJCAI-85; Los Angeles)*

(Los Altos, CA: Morgan Kaufmann): 532-539.

(8) Brachman, Ronald J., & Levesque, Hector J. (1985), *Readings in Knowledge Representation* (Los Altos, CA: Morgan Kaufmann).

(9) Carnap, Rudolf (1928), *The Logical Structure of the World*, R. A. George (trans.) (Berkeley: University of California Press, 1967).

(10) Castañeda, Hector-Neri (1974), "Thinking and the Structure of the World," *Philosophia* 4 (1974) 3-40; originally written in 1972 and reprinted in 1975 in *Critica* 6 (1972) 43-86.

(11) Castañeda, Hector-Neri (1975a), "Individuals and Non-Identity: A New Look," *American Philosophical Quarterly* 12: 131-140.

(12) Castañeda, Hector-Neri (1975b), "Identity and Sameness," *Philosophia* 5: 121-150.

(13) Castañeda, Hector-Neri (1975c), *Thinking and Doing: The Philosophical Foundations of Institutions* (Dordrecht: D. Reidel).

(14) Castañeda, Hector-Neri (1977), "Perception, Belief, and the Structure of Physical Objects and Consciousness," *Synthèse* 35: 285-351.

(15) Castañeda, Hector-Neri (1979), "Fiction and Reality: Their Basic Connections," *Poetica* 8: 31-62.

(16) Chen, Kweijen (1984), "Representation of Neurological Paths by Semantic Networks: User Interface," Project Report (Buffalo, NY: SUNY Buffalo Dept. of Computer Science).

(17) Date, C. J. (1981), *An Introduction to Database Systems* (Reading, MA: Addison-Wesley, 3rd ed.).

(18) Findlay, J. N. (1963), *Meinong's Theory of Objects and Values* (Oxford: Clarendon Press, 2nd ed.).

(19) Findler, Nicholas V. (ed.) (1979), *Associative Networks: Representation and Use of Knowledge by Computers* (New York: Academic Press).

(20) Fine, Kit (1983), "A Defence of Arbitrary Objects," *Proceedings of the Aristotelian Society*, Supp. Vol. 58: 55-77.

(21) Israel, David J. (1983), "Interpreting Network Formalisms," in N. Cercone (ed.), *Computational Linguistics* (Oxford: Pergamon Press): 1-13.

(22) Lambert, Karel (1983), *Meinong and the Principle of Independence* (Cambridge, Eng.: Cambridge University Press).

(23) Levesque, Hector J., & Brachman, Ronald J. (1985), "A Fundamental Tradeoff in Knowledge Representation and Reasoning (Revised Version)," in Brachman & Levesque 1985: 41-70.

(24) Maida, Anthony S., & Shapiro, Stuart C. (1982), "Intensional Concepts in Propositional Semantic Networks," *Cognitive Science* 6: 291-330; reprinted in Brachman & Levesque 1985: 169-189.

(25) Martins, João (1983), "Reasoning in Multiple Belief Spaces," Technical Report No. 203 (Buffalo, NY: SUNY Buffalo Dept. of Computer Science).

(26) Martins, João (1987), "Belief Revision," in S. C. Shapiro (ed.), *Encyclopedia of Artificial Intelligence* (New York: John Wiley): 58-62.

(27) Martins, João, & Shapiro, Stuart C. (1983), "Reasoning in Multiple Belief Spaces," *Proceedings of the Eighth International Joint Conference on Artificial Intelligence (IJCAI-83; Karlsruhe, W. Germany)* (Los Altos, CA: Morgan Kaufmann): 370-373.

(28) Martins, João, & Shapiro, Stuart C. (1984), "A Model for Belief Revision," *Proceedings of the AAAI Non-Monotonic Reasoning Workshop*: 241-294.

(29) Martins, João, & Shapiro, Stuart C. (1986a), "Theoretical Foundations for Belief Revision," in J. Y. Halpern (ed.), *Theoretical Aspects of Reasoning About Knowledge* (Los Altos, CA: Morgan Kaufmann): 383-398.

(30) Martins, João, & Shapiro, Stuart C. (1986b), "Hypothetical Reasoning," *Applications of Artificial Intelligence to Engineering Problems: Proceedings of the 1st International*

Conference (Berlin: Springer-Verlag): 1029-1042.

(31) Martins, João, & Shapiro, Stuart C. (1986c), "Belief Revision in SNePS," *Proceedings of the Sixth Canadian Conference on Artificial Intelligence* (Presses de l'Université de Québec): 230-234.

(32) McCarthy, J. (1979), "First Order Theories of Individual Concepts and Propositions," in J. E. Hayes, D. Michie, & L. Mikulich (eds.), *Machine Intelligence* 9 (Chichester, Eng.: Ellis Horwood): 129-47; reprinted in Brachman & Levesque 1985: 523-533.

(33) McKay, Donald P., & Martins, João (1981), "SNePSLOG User's Manual," SNeRG Technical Note No. 4 (Buffalo, NY: SUNY Buffalo Dept. of Computer Science).

(34) McKay, Donald P., & Shapiro, Stuart C. (1980), "MULTI—A LISP-Based Multiprocessing System," *Proceedings of the 1980 LISP Conference* (*Stanford University*): 29-37.

(35) McKay, Donald P., & Shapiro, Stuart C. (1981), "Using Active Connection Graphs for Reasoning with Recursive Rules," *Proceedings of the Seventh International Joint Conference on Artificial Intelligence* (*IJCAI-81; Vancouver*) (Los Altos, CA: Morgan Kaufmann): 368-374.

(36) Meinong, Alexius (1904), "Über Gegenstandstheorie," in R. Haller (ed.), *Alexius Meinong Gesamtausgabe*, Vol. II (Graz, Austria: Akademische Druck- u. Verlagsanstalt, 1971): 481-535. English translation ("The Theory of Objects") by I. Levi et al., in R. M. Chisholm (ed.), *Realism and the Background of Phenomenology* (New York: Free Press, 1960): 76-117.

(37) Meinong, Alexius (1910), *On Assumptions*, J. Heanue (ed. & trans.) (Berkeley: University of California Press, 1983).

(38) Neal, Jeannette G. (1985), "A Knowledge Based Approach to Natural Language Understanding," Technical Report No. 85-06 (Buffalo, NY: SUNY Buffalo Dept. of Computer Science).

(39) Parsons, Terence (1980), *Nonexistent Objects* (New Haven: Yale University Press).

(40) Rapaport, William J. (1976), *Intentionality and the Structure of Existence*. Ph.D. dissertation (Indiana University Dept. of Philosophy).

(41) Rapaport, William J. (1978), "Meinongian Theories and a Russellian Paradox," *Noûs* 12: 153-180; errata, *Noûs* 13 (1979) 125.

(42) Rapaport, William J. (1981), "How to Make the World Fit Our Language: An Essay in Meinongian Semantics," *Grazer Philosophische Studien* 14: 1-21.

(43) Rapaport, William J. (1982), "Meinong, Defective Objects, and (Psycho-)Logical Paradox," *Grazer Philosophische Studien* 18: 17-39.

(44) Rapaport, William J. (1984a), Critical Notice of Routley 1979, *Philosophy and Phenomenological Research* 44: 539-552.

(45) Rapaport, William J. (1984b), "Belief Representation and Quasi-Indicators," Technical Report No. 215 (Buffalo, NY: SUNY Buffalo Dept. of Computer Science).

(46) Rapaport, William J. (1985a), "Meinongian Semantics for Propositional Semantic Networks," *Proceedings of the 23rd Annual Meeting of the Association for Computational Linguistics (University of Chicago)* (Morristown, NJ: Association for Computational Linguistics): 43-48.

(47) Rapaport, William J. (1985b), "To Be and Not to Be," *Noûs* 19: 255-271.

(48) Rapaport, William J. (1985/1986), "Non-Existent Objects and Epistemological Ontology," *Grazer Philosophische Studien* 25/26: 61-95.

(49) Rapaport, William J. (1986a), Review of Lambert 1983, *Journal of Symbolic Logic* 51: 248-252.

(50) Rapaport, William J. (1986b), "Logical Foundations for Belief Representation," *Cognitive Science* 10: 371-422.

(51) Rapaport, William J., & Shapiro, Stuart C. (1984), "Quasi-Indexical Reference in Propositional Semantic Networks," *Proceedings of the 10th International Conference on Computational Linguistics (COLING-84; Stanford University)* (Morristown, NJ: Association for Computational Linguistics): 65-70.

(52) Routley, Richard (1979), *Exploring Meinong's Jungle and Beyond* (Canberra: Australian National University, Research School of Social Sciences, Dept. of Philosophy).

(53) Shapiro, Stuart C. (1971a), *The MIND System: A Data Structure for Semantic Information Processing*, Report No. R-837-PR (Santa Monica, CA: The Rand Corporation); also AD No. 733 560, Defense Documentation Center, Alexandria, VA.

(54) Shapiro, Stuart C. (1971b), "A Net Structure for Semantic Information Storage, Deduction and Retrieval," *Proceedings of the Second International Joint Conference on Artificial Intelligence (IJCAI-71; London)* (Los Altos, CA: Morgan Kaufmann): 512-523.

(55) Shapiro, Stuart C. (1977), "Representing and Locating Deduction Rules in a Semantic Network," *Proceedings of the Workshop on Pattern-Directed Inference Systems, SIGART Newsletter*, No. 63: 14-18.

(56) Shapiro, Stuart C. (1978), "Path-Based and Node-Based Inference in Semantic Networks," in D. Waltz (ed.), *Tinlap-2: Theoretical Issues in Natural Language Processing* (New York: ACM): 219-225.

(57) Shapiro, Stuart C. (1979b), "The SNePS Semantic Network Processing System," in Findler 1979: 179-203.

(58) Shapiro, Stuart C. (1979c), "Numerical Quantifiers and Their Use in Reasoning with Negative Information," *Proceedings of the Sixth International Joint Conference on Artificial Intelligence (IJCAI-79; Tokyo)* (Los Altos, CA: Morgan Kaufmann): 791-796.

(59) Shapiro, Stuart C. (1982), "Generalized Augmented Transition Network Grammars for Generation from Semantic Networks," *American Journal of Computational Linguistics* 8.1 (January-March 1982) 12-25.

(60) Shapiro, Stuart C.; Martins, João; & McKay, Donald P. (1982), "Bi-Directional Inference," *Proceedings of the Fourth Annual Conference of the Cognitive Science Society (University of Michigan)* (Hillsdale, NJ: Lawrence Erlbaum Associates): 90-93.

(61) Shapiro, Stuart C., & McKay, Donald P. (1980), "Inference with Recursive Rules," *Proceedings of the First Annual National Conference on Artificial Intelligence (AAAI-80; Stanford University)* (Los Altos, CA: Morgan Kaufmann): 151-153.

(62) Shapiro, Stuart C.; McKay, Donald P.; Martins, João; & Morgado, Ernesto (1981), "SNePSLOG: A 'Higher Order' Logic Programming Language," presented at the 1981 Workshop on Logic Programming for Intelligent Systems, Long Beach, CA; SNeRG Technical Note No. 8 (Buffalo, NY: SUNY Buffalo Dept. of Computer Science).

(63) Shapiro, Stuart C.; Srihari, Sargur N.; Geller, James; & Taie, Ming-Ruey (1986), "A Fault Diagnosis System Based on an Integrated Knowledge Base," *IEEE Software* 3.2 (March 1986) 48-49.

(64) Shapiro, Stuart C.; Srihari, Sargur N.; Taie, Ming-Ruey; & Geller, James (1986), "VMES: A Network-Based Versatile Maintenance Expert System," *Proceedings of the 1st International Conference on Applications of Artificial Intelligence to Engineering Problems* (Berlin: Springer-Verlag): 925-936.

(65) Shapiro, Stuart C., & Wand, Mitchell (1976), "The Relevance of Relevance," Technical Report No. 46 (Bloomington, IN: Indiana University Dept. of Computer Science).

(66) Shapiro, Stuart C., & Woodmansee, G. H. (1969), "A Net Structure Based Relational Question Answerer: Description and Examples," *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI-69; Washington, DC)* (Los Altos, CA: Morgan Kaufmann): 325-346.

(67) Shapiro, Stuart C.; Woodmansee, G. H.; & Kreuger, M. W. (1968), "A Semantic Associational Memory Net That Learns and Answers Questions (SAMENLAQ)," Technical Report No. 8 (Madison, WI: University of Wisconsin Computer Sciences Dept.).

(68) Srihari, Rohini K. (1981), "Combining Path-Based and Node-Based Inference in SNePS," Technical Report No. 183 (Buffalo, NY: SUNY Buffalo Dept. of Computer Science).

(69) Srihari, Sargur N.; Hull, Jonathan J.; Palumbo, Paul W.; Niyogi, Debashish; & Wang, Ching-Huei (1985), "Address Recognition Techniques in Mail Sorting: Research Directions," Technical Report No. 85-09 (Buffalo, NY: SUNY Buffalo Dept. of Computer Science).

(70) Suchin, Jennifer (1985), "A Semantic Network Representation of the Peripheral Nervous System," Project Report (Buffalo, NY: SUNY Buffalo Dept. of Computer Science).

(71) Tomberlin, James E. (ed.) (1984), *Agent, Language, and the Structure of the World* (Indianapolis: Hackett).

(72) Tranchell, Lynn M. (1982), "A SNePS Implementation of KL-ONE," Technical Report No. 198 (Buffalo, NY: SUNY Buffalo Dept. of Computer Science).

(73) Woods, William A. (1975), "What's in a Link: Foundations for Semantic Networks," in D. G. Bobrow & A. M. Collins (eds.), *Representation and Understanding: Studies in Cognitive Science* (New York: Academic Press): 35-82; reprinted in Brachman & Levesque 1985: 217-41.

(74) Xiang, Zhigang, & Srihari, Sargur N. (1985), "Spatial Structure and Function Representation in Diagnostic Expert Systems," *Proceedings of the 5th International Workshop on Expert Systems and Their Applications*: 191-206.

(75) Xiang, Zhigang; Srihari, Sargur N.; Shapiro, Stuart C.; & Chutkow, Jerry G. (1984), "Analogical and Propositional Representations of Structure in Neurological Diagnosis," *Proceedings of the 1st Conference on Artificial Intelligence Applications* (Silver Spring, MD: IEEE Computer Society Press): 127-132.

(76) Zalta, Edward (1983), *Abstract Objects* (Dordrecht: D. Reidel).