Bolt Beranek and Newman Inc.

Report No. 4190

Research in Natural Language Understanding

Quarterly Progress Report No. 7, 1 March 1979 to 31 May 1979

Prepared for:
Advanced Research Projects Agency

BEST
AVAILABLE COPY

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER BBN Report No. 4190	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) RESEARCH IN NATURAL LANGUAGE UNDERSTANDING Quarterly Technical Progress Report No. 7 1 March 1979 - 31 May 1979		5. TYPE OF REPORT & PERIOD COVERED Quarterly Progress Report
		6. PERFORMING ORG. REPORT NUMBER BBN Report No. 4190
7. AUTHOR(s) Ronald J. Brachman		8. CONTRACT OR GRANT NUMBER(s) N00014-77-C-0378
9. PERFORMING ORGANIZATION NAME AND ADDRESS Bolt Beranek and Newman Inc. 50 Moulton Street Cambridge, MA 02138		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Office of Naval Research Department of the Navy Arlington, VA 22217		12. REPORT DATE 31 May 1979
		13. NUMBER OF PAGES 25
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Distribution of this document is unlimited. It may be released to the Clearinghouse, Department of Commerce, for sale to the general public.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Natural language, structured inheritance network, knowledge representation, taxonomic lattice, parsing, semantic interpretation, speech acts, pragmatics.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report describes a natural language understanding system based on a taxonomic knowledge representation system. The system makes use of a structured inheritance network, which serves as a taxonomy of sentence types to which semantic interpretation procedures are attached at varying levels of generality. The same taxonomic representation system is used to represent kinds of 'speech act interpretations' of input utterances and to represent the kinds of actions required in response to those utterances.
cont'd.

DD FORM 1473 1 JAN 73 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. Abstract (cont'd.)

The context of the system is the intelligent manipulation of grapnic displays.

RESEARCH IN NATURAL LANGUAGE UNDERSTANDING

Quarterly Technical Progress Report No. 7

1 March 1979 - 31 May 1979

ARPA Order No. 3414

Program Code No. 8D30

Name of Contractor:
Bolt Beranek and Newman Inc.

Effective Date of Contract:
1 September 1977

Amount of Contract:
$712,572

Contract No. N00014-77-C-0378

Contract Expiration Date:
31 August 1979

Short Title of Work:
Natural Language Understanding

Principal Investigator:
Dr. William A. Woods
(617) 491-1850, x4361

Scientific Officer
Gordon D. Goldstein

Sponsored by
Advanced Research Projects Agency
ARPA Order No. 3414

This research was supported by the Advanced Research Projects Agency of the Department of Defense and was monitored by ONR under Contract No. N00014-77-C-0378.

TABLE OF CONTENTS

Taxonomy, Descriptions, and Individuals in Natural Language Understanding*

Ronald J. Brachman

KLONE is a general-purpose language for representing conceptual information. Several of its prominent features - semantically clean inheritance of structured descriptions, taxonomic classification of generic knowledge, intensional structures for functional roles (including the possibility of multiple fillers), and procedural attachment (with automatic invocation) - make it particularly useful in computer-based natural language understanding. We have implemented a prototype natural language system that uses KLONE extensively in several facets of its operation. This report describes the system and points out how it uses KLONE for representation in natural language processing.

Our system is the beneficiary of two kinds of advantage from KLONE. First, the taxonomic character of the structured inheritance net facilitates the **processing** involved in analyzing and responding to an utterance. In particular, (1) it helps guide parsing by ruling out semantically meaningless paths, (2) it provides a general way of organizing and invoking semantic

* This report is a slightly revised version of a paper presented at the 17th Annual Meeting of the Association for Computational Linguistics, La Jolla, CA, August 11, 1979.

interpretation rules, and (3) it allows algorithmic determination of equivalent sets of entities for certain plan-recognition inferences. Second, KLONE's **representational structure** captures some of the subtleties of natural language expression. That is, it provides a general way of representing exactly the quantificational import of a sentence without over-committing the interpretation to scope or multiplicity not overtly specified.

In this report, we first present a brief overall description of the natural language system. Then, prior to describing how we use KLONE in the system, we discuss some of the language's features at a general level. Finally, we look in detail at how KLONE affords us the advantages listed above.

1. The Task and the System

Our general task is to provide a natural interface to an intelligent display system in a command and control environment. The component of our system that manipulates the (bit-map) display - the 'Advanced Information Presentation System' (AIPS) - represents explicitly (in KLONE) all objects (ships, etc.) to be presented, their presentation forms (circles, text, etc.), descriptions of view surfaces on which to project presentations of the objects, and coordinate mappings between those surfaces. This explicit representation allows the user to flexibly alter at will the picture s/he sees by adding or moving display windows, changing size, shape, etc. of display forms, and adding and removing objects or object detail. The user changes the subject and form of what s/he sees by **describing** what s/he wants displayed.

In the particular system to be described in this report, we have taken as our domain of discourse the Augmented Transition Network (ATN) Grammar from the LUNAR natural language understanding system [Woods, Kaplan and Nash-Webber, 1972]. Thus, the objects to be displayed are the states and arcs of the ATN, including state names, arc types, conditions, actions, etc. Our particular display setup has three windows - for prompts, text interaction, and grammar display. At the moment, the size and placement of these windows is fixed; but these could be easily changed using the AIPS facility.

The addition of a natural language interface to AIPS yields more than just a convenient way to state explicit display changes. Now the display can be altered in response to a question (e.g., highlighting a ship to mean "there!" in response to a "where" question), or to an indirect speech act (e.g., "I want to see it" produces a display of the appropriate object). Further, natural language provides a convenient way to express standing orders of various types (e.g., "Display ships with radar as flashing triangles"; "whenever three ships are in the same convoy, and within 6 miles of each other, use a single task force symbol to stand for the set of ships").

A simple dialogue will serve to show the blend of natural language and intelligent knowledge-based graphics that we envision in the command and control environment (note the use of user-pointing input as well as language):

1) Show me the clause level network.
   [System displays states and arcs of the S/ network]

2) Show me S/NP.
   [System highlights state S/NP]

3) Focus in on the preverbal constituents.
   [System shifts scale and centers the display on the preverbal states]

4) No. I want to be able to see S/AUX.
   [System "backs off" display so as to include state S/AUX]

5) Remove the highlight from this <user points> state.
   [System removes highlight from S/NP]

At the same time, we would like to ask factual questions about the states, arcs, etc. of the ATN (e.g., "What are the conditions on this <user points> arc?"). Questions and commands addressed to the system typically (1) make use of elements of the preceding dialogue, (2) can be expressed indirectly so that the surface form does not reflect the real intent, and (3) given our graphical presentation system, can make reference to a shared non-linguistic context. The issues of anaphora, (indirect) speech acts, and deixis are thus of principal concern.

1.1 System Organization

The natural language system is organized as illustrated in Figure 1. The user sits at a bit-map terminal equipped with a keyboard and a pointing device. Typed input from the keyboard (possibly interspersed with coordinates from the pointing device) is analyzed by a version of the **RUS System** [Bobrow, 1978] - an ATN-based incremental parser that is closely coupled with a "case-frame dictionary". In our system, this dictionary is

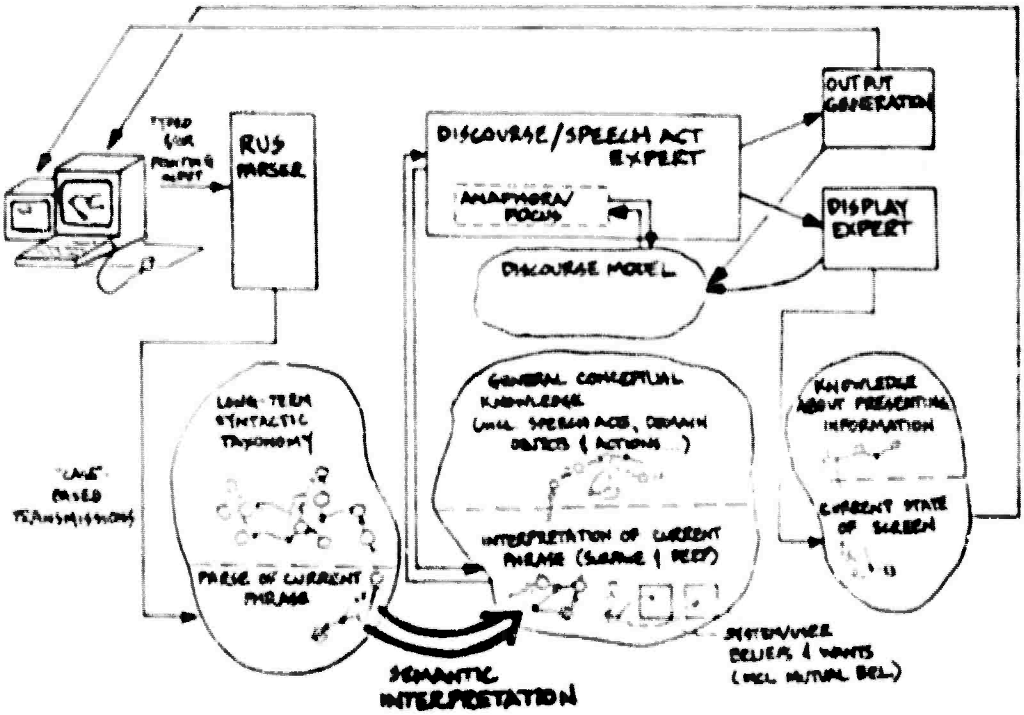

Figure 1. System structure
(highlighting types of knowledge involved).

embodied in a syntactic taxonomy represented in KLONE. The parser produces a KLONE representation of the syntactic structure of an utterance. Incrementally along with its production, this syntactic structure triggers the creation of an interpretation. The interpretation structure - the literal (sentential) semantic content of the utterance - is then processed by a discourse expert that attempts to determine what was really meant. In this process, anaphoric expressions must be resolved and indirect speech acts recognized. Finally, on the basis of what is determined to be the intended force of the utterance, the discourse component decides how the system should respond. It plans its own speech or display actions, and passes them off to the language generation component (not yet implemented) or display expert. Some of these operations will be discussed in more detail in Section 3.

2. The Representation Language

KLONE is a uniform language for the explicit representation of conceptual information based on the idea of **structured inheritance networks** [Brachman, 1978, 1979]. The principal representational elements of KLONE are **Concepts**, of which there are two major types - Generic and Individual. Generic Concepts are arranged in an inheritance structure, expressing long-term generic knowledge as a taxonomy. A single Generic Concept is a description template, from which individual descriptions (in the form of Individual Concepts) are formed. Generic Concepts can be built as specializations of other Generic Concepts, to which they are attached by **inheritance Cables**. These Cables form the backbone of the network (a Generic Concept can have many "superConcepts" as well as many "subConcepts"). They carry structured descriptions from a Concept to its subConcepts.

KLONE Concepts are highly structured objects. A subConcept inherits a **structured** definition from its parent* and can modify it in a number of structurally consistent ways. The main elements of the structure are **Roles**, which express relationships between a Concept and other closely associated Concepts (i.e., its properties, parts, etc.). Roles themselves have structure, including descriptions of potential fillers,** modality information, and names.*** There are basically two kinds of Roles in KLONE: **RoleSets** and **IRoles**. RoleSets have potentially many fillers and may carry a restriction on the number of possible

---

* This inheritance implies **inter alia** that, if STATE is a subConcept of ATN-CONSTITUENT, then any particular state is **by definition** also an ATN constituent.

** These limitations on the form of particular fillers are called "Value Restrictions" (V/R's). If more than one V/R is applicable at a given Role, the restrictions are taken conjunctively.

*** Names are not used by the system in any way. They are merely conveniences for the user.

vice-president, etc.; this is a relationship between RoleSets in which the more specific Roles inherit all properties of the parent Role except for the number restriction;

- particularization (of a RoleSet for an Individual Concept); e.g., the **officers** of BBN are all COLLEGE-GRADUATEs; this is the relationship between a RoleSet of an Individual Concept and a RoleSet of a parent Generic Concept.

- satisfaction (binding of a particular filler description into a particular Role in an Individual Concept); e.g., the **president** of BBN is STEVE-LEVY; this is the relationship between an IRole and its parent RoleSet.

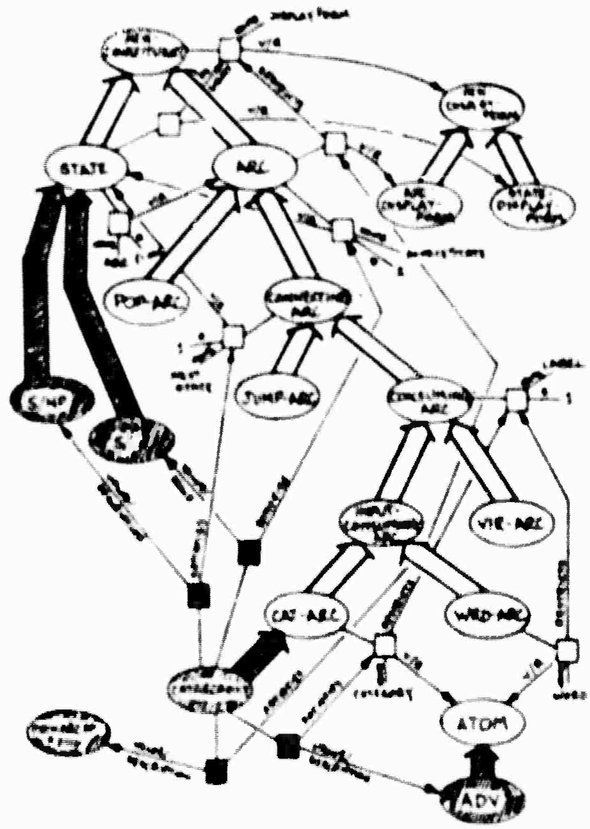

Fig. 2. A piece of a KLONE taxonomy.

Figure 2 illustrates the use of Cables and the structure of Concepts in a piece of the KLONE taxonomy for the ATN grammar. In this figure, Concepts are presented as ellipses (Individual Concepts are shaded), Roles as small squares (IRoles are filled in), and Cables as double-lined arrows. The most general Concept, ATN-CONSTITUENT, has two subConcepts - STATE and ARC. These each inherit the general properties of ATN constituents, namely, each is known to have a **displayForm** associated with it. The subnetwork below ARC expresses the classification of the various types of arcs in the ATN and how their conceptual structures vary. For example, a CONNECTING-ARC has a **nextState** (the state in which the transition leaves the parsing process), while for POP-ARCs the term is not meaningful (i.e., there is no **nextState** Role). Links that connect the Roles of more specific Concepts with corresponding Roles in their parent Concepts are considered to travel through the appropriate Cables. Finally, the structure of an Individual Concept is illustrated by CATARC#0117. Each IRole expresses the filling of a Role inherited from the hierarchy above -- because CATARC#0117 is a CAT-ARC, it has a **category**; because it is also a CONNECTING-ARC, it has a **nextState**, etc.

The structure of a Concept is completed by its set of **Structural Descriptions** (SD's). These express how the Roles of the Concept interrelate via the use of parameterized versions

("ParaIndividuals") of other Concepts in the network to describe quantified relations between the ultimate fillers of the Concept's Roles. The quantification is expressed in terms of set mappings between the RoleSets of a Concept, thereby quantifying over their sets of fillers. In addition to quantified relations between potential Role fillers, simple relations like subset and set equality can be expressed with a special kind of SD called a "RoleValueMap" (e.g., the relation that "the **object** of the **precondition** of a SEEing action is the same as the **object** of its **effect**"). SD's are inherited through cables and are particularized in a manner similar to that of Roles.

There is one important feature of KLONE that is worth pointing out, although it is not yet used in the current natural language system. The language carefully distinguishes between purely descriptional structure and assertions about coreference, existence, etc. All of the structure mentioned above (Concepts, Roles, SD's and Cables) is **definitional**. A separate construct called a **Nexus** is a used as a locus of coreference for Individual Concepts. One expresses coreference of description relative to a **Context** by placing a Nexus in that Context and attaching to it Individual Concepts considered to be coreferential. All assertions are made relative to a Context, and thus do not affect the (descriptive) taxonomy of generic knowledge. We anticipate that

Nexuses will be important in reasoning about particulars, answering questions (especially in deciding the appropriate form for an answer), and resolving anaphoric expressions, and that Contexts will be of use in reasoning about hypotheticals, beliefs, and wants.

The final feature of KLONE relevant to our discussion is the ability to attach procedures and data to structures in the network. The attached procedure mechanism is implemented in a very general way. Procedures are attached to KLONE entities by "interpretive hooks" (ihooks), which specify the set of situations in which they are to be triggered. An interpreter function operating on a KLONE entity causes the invocation of all procedures inherited by or directly attached to that entity by ihooks whose situations match the intent of that function. Situations include things like "Individuate", "Modify", "Create", "Remove", etc. In addition to a general situation, an ihook specifies when in the execution of the interpreter function it is to be invoked (PRE-, POST-, or WHEN-).

3. Use of KLONE in the Natural Language System

As mentioned previously, KLONE is used in several places in our language understanding system - these include the syntactic taxonomy used to constrain parsing and to index semantic interpretation rules, and the structures used in the syntactic/discourse interface to express the literal semantic content of an utterance. The parser uses KLONE to describe potential syntactic structures. A taxonomy of syntactic constituent descriptions, with Concepts like PHRASE, NOUN-PHRASE, LOCATION-PP, and PERSON-WORD, is used to express how phrases are built from their constituents. The taxonomy also serves as a discrimination net, allowing common features of constituent types to be expressed in a single place, and distinguishing features to cause branching into separate subnets.

Two benefits accrue from this organization of knowledge. First, shallow semantic constraints are expressed in the Roles and SD's of Concepts like LOCATION-PP. For example, the **prepObject** of a LOCATION-PP must be a PLACE-NOUN. A description of "on AI" (as in "book on AI") as a LOCATION-PP could not be constructed since AI does not satisfy the value restriction for the **head** role. Such constraints help rule out misleading parse paths, in the manner of a semantic grammar [Burton, 1976], by refusing to construct semantically anomalous constituent descriptions. In conjunction with the general (ATN) grammar of English, this is a powerful guidance mechanism which helps parsing proceed close to deterministically [Bobrow, 1978].

Second, the syntactic taxonomy serves as a structure on which to hang semantic projection rules. Since the taxonomy is an inheritance structure, the description of a given syntactic constituent inherits all semantic interpretation rules appropriate for each of the more general constituent types that it specializes, and can have its own special-purpose rules as well. In the example above, simply by virtue of its placement in the taxonomy, the Concept for "on AI" would inherit rules relevant to prepositional phrases in general and to SUBJECT-PP's in particular, but not those appropriate to LOCATION-PP's. Interpretation **per se** is achieved using the attached procedure facility, with semantic projection rules expressed as functions attached to Roles of the syntactic Concepts. The functions specify how to translate pieces of syntactic structure into "deeper" Concepts and Roles. For example, the **subject** of a SHOW-PHRASE might map into the **agent** of a DISPLAY action.

The mapping rules are triggered automatically by the KLONE interpreter. This is facilitated by the interpreter's "pushing down" a Concept to the most specific place it can be considered to belong in the taxonomy (using only "analytic", definitional constraints). Figure 3 illustrates schematically the way a Concept can descend to the most specific level implied by its internal description. The Concept being added to the network is an NP whose **head** is "ARC" and whose **modifier** is "PUSH" (NP00023). It is initially considered a direct (Generic) subConcept of the Concept

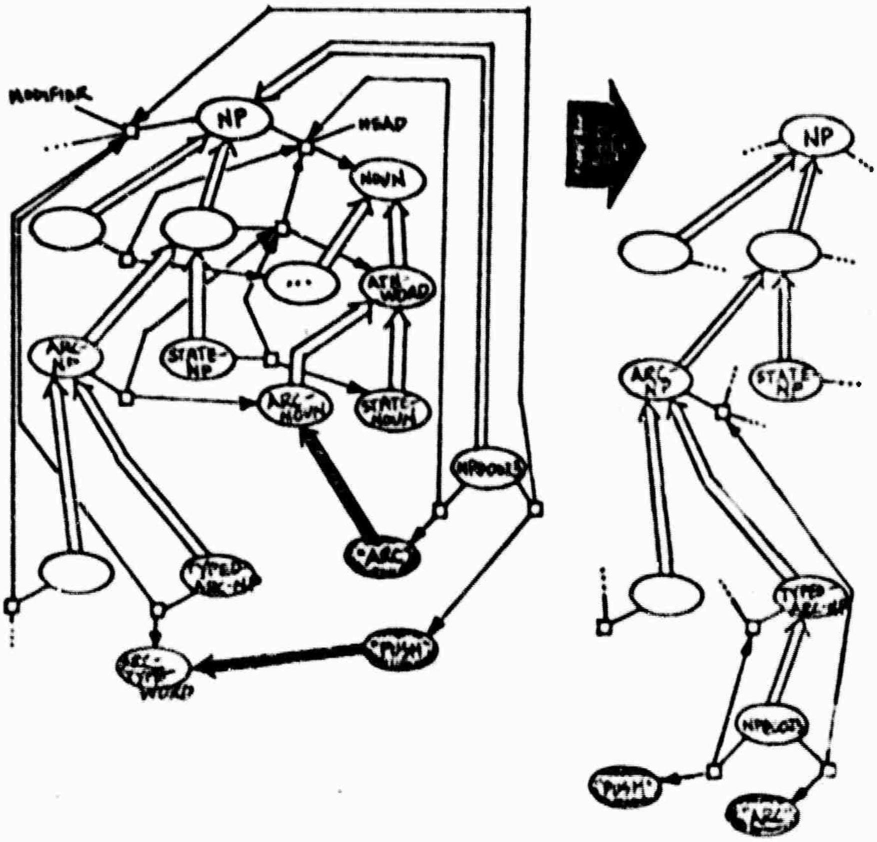

Fig. 3. Automatic concept descent.

for its basic syntactic type (NP). Its Role structure, however, implies that it in fact belongs in a more restricted subclass of NP's, that is, TYPED-ARC-NP (an NP whose **head** is an ARC-NOUN and whose **modifier** is an ARC-TYPE-WORD). The interpreter, on the basis of only definitional constraints expressed in the network, places the new Concept below its "most specific subsumer" -- the proper place for it in the taxonomy. The process proceeds incrementally, with each new piece of the constituent possibly causing further descent. In this case, NP00023 would initially only have its **head** Role specified, and on that basis, it would be placed under ARC-NP (which is "an NP whose **head** is an ARC-NOUN"). Then the parser would add the **modifier** specification, causing the Concept's descent to the resting place shown in the right half of Figure 3. When the constituent whose description is being added to the network is "popped" in the parser, its KLONE description is **individuated** -- causing the invocation of all "WHEN-Individuated" attached procedures inherited through superConcept Cables. These procedures cause an interpretation for the constituent to be built on the basis of the interpretations of component parts of the syntactic description.

The literal semantic interpretation of a phrase produced by semantic interpretation - also a KLONE structure - is the "input" to the discourse component. An important element of this interface between the syntactic processor and the discourse component is that the parser/interpreter commits itself only to information explicitly present in the input phrase, and leaves all inference about quantifier scope, etc. to the discourse expert. Two kinds of representational structures support this. The Concept DSET (for "determined set") is used extensively to capture sets implicit in noun phrases and clauses. DSETs use the inherent multiplicity of RoleSets to group together several entities under a single Concept, and to associate determiners (definite/indefinite, quantifiers, etc.) with such a set of entities. The former is accomplished using a single **member** RoleSet whose multiplicity is open-ended (between 0 and infinity); the latter is achieved by simply having a **determiner** RoleSet whose number is restricted to be 1. A DSET can express the characteristics of a set of entities without enumerating them explicitly, or even indicating how many members the set is expected to have. RoleValueMaps allow constraints between DSETs to be expressed in a general way - a RoleValueMap expresses a subset or equality relation between two RoleSets. Such relations can be constructed without knowing in advance the cardinality of the sets or any of their members.

Figure 4 illustrates the use of these structures to express the intent of the sentence, "Show me states S/NP, S/AUX, and S/DCL."* DSET#00035 represents the interpretation of the noun phrase, "the states S/NP, S/AUX, and S/DCL". The generic DSET Concept has two Roles, **member** and **determiner**. The **member** Role can be filled multiply, and therein lies the "settedness" of the DSET. DSET#00035 has a particularized version of the **member** Role: Role R1 represents the set of three states mentioned in the noun phrase, as

* RoleSets in this figure are drawn as squares with circles around them. RoleSets with filled-in circles are a special kind of particularized RoleSet that can occur only in Individual Concepts. The RoleValueMap is pictured as a diamond.

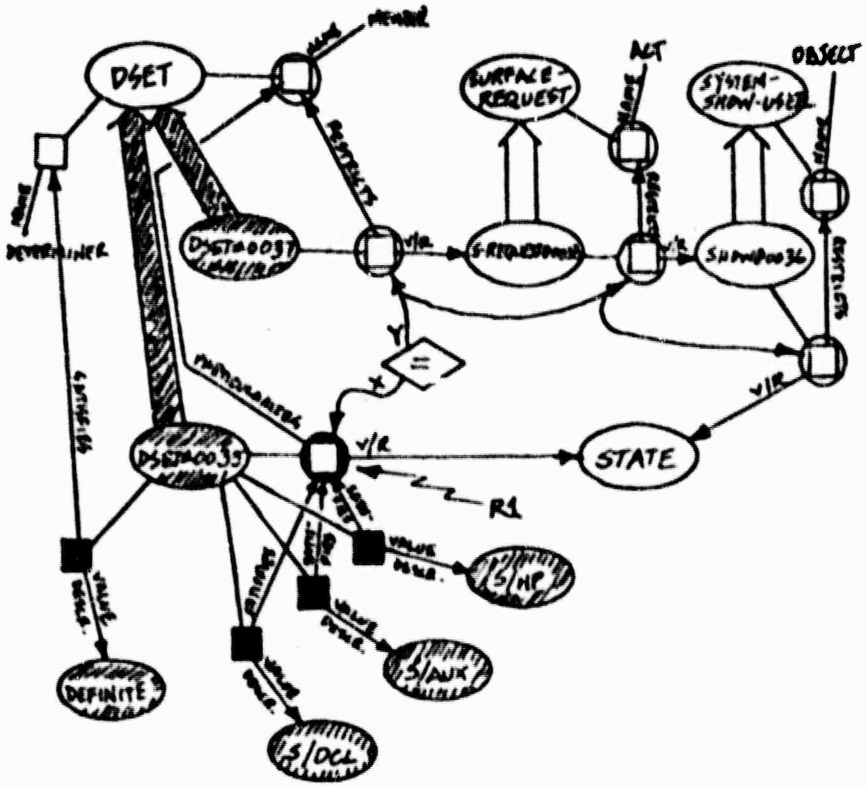

Fig. 4. KLONE description of
"Show me states S/NP, S/AUX, and S/DCL"

a group. Thus, the Value Restriction of R1, STATE, applies to each member. The three IRoles of DSET#00035, connected by "Satisfies" links to the particularized **member** RoleSet, indicate that the particular states are the members of the set.*

* The Value Restriction, STATE, is redundant here, since the members of this particular set were explicitly specified (and are known to be states). In other cases, the information is more useful. For example, no IRoles would be constructed by the parser if the sentence were "Are there three states?"; only one would be constructed in "Show me state S/NP and its two nearest neighbors". On the other hand, no Value Restriction would be directly present on Role R1 if the noun phrase were just "S/NP, S/AUX, and S/DCL".

The other DSET in the figure, DSET#00037, represents the clause-level structure of the sentence. The clause has been interpreted into something like "the user has performed what looks on the surface to be a request for the system to show the user some set of states".

This captures several kinds of indeterminacy: (1) that the sentence may only be a request at the surface level ("Don't you know that pigs can't fly?" looks like a request to inform), (2) that there is more than one way to effect a "show" ("show" could mean redraw the entire display, change it slightly to include a new object, or simply highlight an existing one), (3) that it is not clear how many operations are actually being requested (showing three objects could take one, two, or three actions). Therefore, the interpretation uses Generic Concepts to describe the kind of events appearing in the surface form of the sentence and makes no commitment to the number of them requested. The only commitment to "quantificational" information is expressed by the RoleValueMap. Its two pointers, X (pointing to the **member** Role of DSET#00035) and Y* (pointing to the **object** of the requested **act**), indicate that the

* Y is a chained pointer going first through the **member** Role of DSET#00037, then through the **act** Role of S-REQUEST#00038, and finally to the **object** Role of SHOW#00036. It is considered to refer to the set of IRoles expressing the objects of all SHOW events ultimately S-REQUESTed, when it is determined exactly how many there are to be (i.e., when the IRoles of DSET#00037 are finally specified). Thus, if there are ultimately two SHOWs, one

ultimate set of things to be shown, no matter how many particular SHOW events take place, must be the same as the set of members in the noun phrase DSET (namely, the three states).

Given the input from semantic interpretation, the discourse expert looks for a plan that it can hypothesize its user to be following, in order to interpret indirect speech acts. Following [Allen, 1979], the speech acts REQUEST, INFORM, INFORMREF, and INFORMIF are defined as producing certain effects by means of the hearer's recognition of the speaker's intention to produce these effects. Indirect speech act recognition proceeds by inferring what the user wants the system to think is his/her plan. Plan-recognition involves making inferences of the form, "the user did this action in order to produce that effect, which s/he wanted in order to do this (next) action".

Making inferences at the level of "intended plan recognition" is begun by analyzing the user's utterance as a "surface" speech act (SURFACE-REQUEST or SURFACE-INFORM) indicating what the utterance "looks like". By performing plan-recognition inferences whose plausibility is ascertained by using mutual beliefs, the system can, for instance, reason that what looked to be an INFORM of the user's goal is actually a REQUEST to include some portion of

of one state and the other of two, the Y pointer implicitly refers to the set of all three states shown.

the ATN into the display. Thus, the second clause of the utterance, "No; I want to be able to see S/AUX," is analyzed as a REQUEST to INCLUDE S/AUX by the following chain of plan-recognition inferences:

The system believes

1) The user has performed a SURFACE-INFORM of his/her goal; thus

2) The user intends for the system to believe that the user wants to be able to see S/AUX. Since this requires that S/AUX be visible,

3) The user intends for the system to believe that the user wants the system to plan an action to make S/AUX visible. Because the "No" leads to an expectation that the user might want to modify the display, the system plans to INCLUDE S/AUX in the existing display, rather than DISPLAY S/AUX alone.

4) Hence, the user intends for the system to believe that user wants the system to INCLUDE S/AUX.

5) The user has performed a REQUEST to INCLUDE.

The system responds by planning that action.

In addition to using Contexts to hold descriptions of beliefs and wants, the plan-recognition process makes extensive use of RoleValueMaps and DSETs (see Figure 4). Plan-recognition inferences proceed using just the clause-level structure and pay no attention to the particulars of the noun phrase interpretations. The system creates new DSETs for intermediate sets and equates them to previous ones by RoleValueMaps, as, for example, when it decides to do a SHOW whose **object** is to be the same as whatever was to be visible. At the end of plan-recognition the system may need to trace through the constructed RoleValueMaps to find all sets equivalent to a given one. For instance, when it determines that it needs to know which set of things to display, highlight, or include, it treats the equated RoleValueMaps as a set of rewrite rules, traces back to the original noun phrase DSET, and then tries to find the referent of that DSET.*

Finally, not only are parse structures and semantic interpretations represented in KLONE, but the data base - the ATN being discussed - is also represented as KLONE structure (see Figure 2, above). Further, descriptions of how to display the ATN, and general descriptions of coordinate mappings and other display information are represented in KLONE too. Commands to the display expert are expressed as Concepts involving actions like SHOW, CENTER, etc. whose "arguments" are descriptions of desired shapes, etc. Derivations of particular display forms from generic descriptions, or from mapping changes, are carried out by the attached procedure mechanism. Finally, once the particular shapes are decided upon, drawing is achieved by invoking "how to draw" procedures attached to display form Concepts. Once again, the

* The system only finds referents when necessary. This depends on the user's speech acts and the system's needs in understanding and complying with them. Thus, it is intended that a naming speech act like "Call that the complement network" will not cause a search for the referent of "the complement network".

taxonomic nature of the structured inheritance net allows domain structure to be expressed in a natural and useful way.

4. References

Allen, James F. A Plan-based Approach to Speech Act Recognition. Technical Report No. 131/79. Toronto, Ontario: Dept. of Computer Science, University of Toronto, February 1979.

Bobrow, R.J. The RUS System. In Research in Natural Language Understanding: Quarterly Progress Report No. 3 (1 March 1978 to 31 May 1978). BBN Report No. 3878. Cambridge, MA: Bolt Beranek and Newman Inc., July 1978.

Brachman, R.J. A Structural Paradigm for Representing Knowledge. Ph.D. Dissertation, Harvard University, Cambridge, MA, May 1977. Also BBN Report No. 3605. Cambridge, MA: Bolt Beranek and Newman Inc., May 1978.

Brachman, R.J. On the Epistemological Status of Semantic Networks. In Findler, N.V. (ed.). **Associative Networks: Representation and Use of Knowledge by Computers.** New York: Academic Press, 1979, pp. 3-50.

Burton, R.R. Semantic Grammar: An Engineering Technique for Constructing Natural Language Understanding Systems. BBN Report No. 3453. Cambridge, MA: Bolt Beranek and Newman Inc., December, 1976.

Woods, W.A., Kaplan, R. M., and Nash-Webber, B. The Lunar Sciences Natural Language Information System: Final Report. BBN Report No. 2378. Cambridge, MA: Bolt Beranek and Newman Inc., 1972.

Official Distribution List
Contract N00014-77-C-0378

	Copies
Defense Documentation Center Cameron Station Alexandria, VA 22314	12
Office of Naval Research Information Systems Program Code 437 Arlington, VA 22217	2
Office of Naval Research Code 200 Arlington, VA 22217	1
Office of Naval Research Code 455 Arlington, VA 22217	1
Office of Naval Research Code 458 Arlington, VA 22217	1
Office of Naval Research Branch Office, Boston 495 Summer Street Boston, MA 02210	1
Office of Naval Research Branch Office, Chicago 536 South Clark Street Chicago, IL 60605	1
Office of Naval Research Branch Office, Pasadena 1030 East Green Street Pasadena, CA 91106	1
Office of Naval Research New York Area Office 715 Broadway - 5th Floor New York, NY 10003	1

Naval Research Laboratory 6
Technical Information Division
Code 2627
Washington, D.C. 20380

Naval Ocean Systems Center 1
Advanced Software Technology Division
Code 5200
San Diego, CA 92152

Dr. A.L. Slafkosky 1
Scientific Advisor
Commandant of the Marine Corps (Code RD-1)
Washington, D.C. 20380

Mr. E.H. Gleissner 1
Naval Ship Research & Development Center.
Computation & Mathematics Dept.
Bethesda, MD 20084

Capt. Grace M. Hopper 1
NAICOM/MIS Planning Branch (OP-916D)
Office of Chief of Naval Operations
Washington, D.C. 20350

Mr. Kin B. Thompson 1
NAVDAC 33
Washington Navy Yard
Washington, D.C. 20374

Advanced Research Projects Agency 1
Information Processing Techniques
1400 Wilson Boulevard
Arlington, VA 22209

Capt. Richard L. Martin, USN 1
Commanding Officer
USS Francis Marion (LPA-249)
FPO New York 09501

Director, National Security Agency 1
Attn: R53, Mr. Glick
Fort G. G. Meade, MD 20755

Ms. Robin Dillard 1
Naval Ocean Systems Center
C2 Information Processing Branch (Code 8242)
271 Catalina Blvd.
San Diego, CA 92152