

# SRI International

MACHINE LEARNING FOR INFORMATION MANAGEMENT

Technical Note 252

July 28, 1981

By: Norman Haas, Computer Scientist  
Gary G. Hendrix, Program Director/Natural Language

Artificial Intelligence Center  
Computer Science and Technology Division

This research was supported by the Defense Advanced Research Projects Agency with the Naval Electronic Systems Command under Contracts N00039-79-C-0118 and N00039-80-C-0575. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency of the United States Government.

This paper expands upon material previously presented in SRI AIC Technical Note 227.



333 Ravenswood Ave. • Menlo Park, CA 94025  
(415) 859-6200 • TWX: 910-373-2046 • Telex: 334 486

## ABSTRACT

This paper discusses machine learning in the context of information management. The core idea is that of a compiler system that can hold a conversation with a user in English about his specific domain of interest, subsequently retrieve and display information conveyed by the user, and apply various types of external software systems to solve user problems.

The specific learning problem discussed is how to enable computer systems to acquire information about domains with which they are unfamiliar from people who are expert in those domains, but have little or no training in computer science. The information to be acquired is that needed to support question-answering or fact retrieval tasks, and the type of learning to be employed is learning by being told. Reflecting the intimate connection between language and reasoning, this paper is largely concerned with the problems of learning concepts and language simultaneously.

## I OVERVIEW

### A. Learning for Information Management

This paper discusses machine-learning aspects of a project whose broad goal is to create computer systems that can aid users in managing information. The specific learning problem discussed is how to enable computer systems to acquire information about domains with which they are unfamiliar from people who are experts in those domains, but have little or no training in computer science. The information to be acquired is that needed to support question-answering or fact retrieval tasks, and the type of learning to be employed is learning by being told.

### B. The KLAUS Concept

Our interest in knowledge acquisition is motivated by the desire to create computer systems that can aid users in managing information. The core idea of what we call a KLAUS\* system is that of a machine that can hold a conversation with a user in English about his specific domain of interest, subsequently retrieve and display information conveyed by the user, and apply various types of external software systems to solve user problems. Such software would include data base management systems, report generators, planners, simulators, and statistical packages.

Interactive dialogues in natural language appear to be a convenient means for obtaining most of the application-specific knowledge needed by intelligent systems for information management. But systems that acquire knowledge about new domains through natural-language dialogues must possess some very special capabilities.

KLAUS systems must support interactive, mixed-initiative dialogues. Because a user may provide new knowledge in an incremental and incomplete manner, the system must keep track of what it has already been told, so that it can deduce the existence of missing information

-----  
\* "KLAUS" stands for Knowledge-Learning and -Using Systems.

and explicitly ask the user to supply it. Moreover, it must carefully distinguish what it does not know from what it knows to be false.

A primary requirement of a KLAUS system is that it be capable of simultaneously learning both new concepts and the linguistic constructions used to express them. KLAUS systems must acquire domain-specific language expertise, not only to understand natural-language statements formulated by the user about his domain, but also for generating natural-language responses to user requests.

The intimate connection between language and reasoning is reflected in the need to acquire concepts and language simultaneously. This poses a great challenge in the task of creating KLAUS systems. Thus, this paper is largely concerned with the problems of learning concepts and language simultaneously.

### C. Research Problems for KLAUS Systems

Before systems can be created that are capable of learning about new domains through interactive dialogues in English, several fundamental research problems must be resolved:

- \* A powerful natural-language processing capability is required. Although much progress has been made in recent years, previous work has presupposed a complete knowledge base. Knowledge acquisition dialogues require numerous adaptations and extensions to the technology.
- \* A structure for lexical entries must be specified so that the system can acquire new lexical information. Because such information constitutes a key link between surface linguistic form and underlying meaning, structural specification is a challenging task for certain categories of words, particularly verbs.
- \* The linguistic constructions that people use in introducing new concepts must be identified and analyzed, so they can be interpreted correctly by the natural-language processing system. Such constructions range from simple syntactic patterns to complex analogies.
- \* Seed concepts and seed vocabulary must be identified for inclusion in a core system. It is not obvious which words and concepts would be most useful in helping users describe new domains.

- \* A flexible scheme of knowledge representation is necessary. Such a representation must have general expressive power, since it may be applied to diverse domains and must support the addition of new information. It should include inherent features that can aid in organizing knowledge and supporting incremental acquisition.
- \* An efficient problem-solving capability is needed to answer questions and draw inferences for integrating newly acquired information. This capability must be based on general principles, because no application-specific problem-solving procedures will be included in the system. (How to learn application-specific problem-solving procedures is a separate and interesting research question.)
- \* A methodology is needed for integrating new concepts into the system's knowledge base. Because users will often provide only partial descriptions of new concepts, methods must be devised for ascertaining what additional facts should be sought from the user to ensure proper linkage between the new concepts and those previously acquired.
- \* A set of readily understandable questions is needed for eliciting information from the user. The length and number of questions should be kept to a minimum, so as not to impose an excessive burden on users.
- \* Facilities must be provided for allowing a user to change his mind about what he has told the system. That is, users should be able to instruct the system to modify, revise or refute information it has been told previously.
- \* Means are required for detecting and dealing with inconsistent data.

These problems must be dealt with in an integrated manner, balancing the requirements of one facet of the system against those of other facets. Our initial attempts to cope with this complex web of issues are described below.

#### D. Other Learning Systems

Our application of the technique called "learning-by-being-told" to collect and organize relatively large aggregations of individual facts for use in question-answering tasks stands apart from two other major approaches in knowledge acquisition research. One of these involves the acquisition of rules for judgmental reasoning, as exemplified by the

work of Davis [1] and Dietterich and Michalski [2]. In this paradigm, knowledge is viewed not so much as a collection of facts, but as a set of rules that in their aggregate comprise an algorithm for making some type of decision. The other approach explores learning by example and learning by analogy. This latter notion is the basis of much other research on learning, such as that of Winston [19] and Mitchell [12].

## II TECHNICAL APPROACH: EXPERIMENTS WITH THE KLAUS CONCEPT

We have recently developed and tested a pilot KLAUS, called NANOKLAUS. A sample transcript of interactions with this system is contained in the appendix. Readers are encouraged to glance through this transcript before proceeding with the reading of this text.

The principal components of NANOKLAUS are a natural-language processing module, a formal-deduction module that operates on a data base of well-formed formulas (wffs) in first-order logic, and a number of support procedures that aid in assimilating knowledge about new subject domains and in maintaining the data base.

### A. Seed Concepts

NANOKLAUS comes preprogrammed with a fixed set of syntactic and semantic rules covering a small subset of English. It also comes with seed concepts and a seed vocabulary, which are to be extended as the system learns about a new domain. For example, the system comes with a preliminary taxonomy of concepts already defined. This basic set includes such things as PHYSICAL OBJECTS, PERSONS, MEASURES, and the like. NANOKLAUS also has predefined lexical entries for the basic function words of English, as well as of such words as "unit," "kind," and "plural" that are used frequently in articulating definitions of new words and concepts. These seed concepts allow the untrained NANOKLAUS to "understand" inputs such as those of Interactions 3 and 4 of the transcript.

The choice of seed concepts for a system that must bootstrap its entry into new domains is problematical. Most of the concepts we selected for NANOKLAUS are classes of THINGS and RELATIONS. They have been included in the system either simply to avoid forcing users to relate everything to the most general concept (THING), or because they have a special status in English. For example, because pronominal usage depends partially on gender, the class MALE is defined and associated with the pronoun "he."

It is important not to think of the seed concepts as a set of primitives, in terms of which all other concepts must be defined. Concept acquisition in NANOKLAUS is not based on definitions. Rather, new concepts are introduced by the user and progressively refined by adding more and more facts connecting the new concept to other concepts. Each new fact acts as a constraint that the concepts it mentions must satisfy. Thus, concept acquisition is a process of pruning away possibilities, rather than building up from primitives. For arguments as to the general unworkability of the latter approach, see [3].

NANOKLAUS uses seven principles of knowledge organization to integrate new knowledge:

- (1) There are things.
- (2) There are subclasses of things (i.e., things can be classified taxonomically).
- (3) There are relations among things.
- (4) There are subclasses of relations.
- (5) Some relations are functions (i.e., n to 1 maps).
- (6) Sometimes a given set of constraints is sufficient to distinguish a unique individual.
- (7) Equals are interchangeable.

NANOKLAUS is not programmed to hold explicit conversations about these principles, but rather to utilize them in its internal operations.

## B. NANOKLAUS's Natural-Language Component

The natural-language component of NANOKLAUS is based on LIFER [6] and uses a pragmatic grammar in the style of LADDER [8]. In particular, its grammar consists of a number of highly specific, special-purpose rules for processing various types of sentences.\* For example, the grammar may be thought of as including a rule of the form

```
<SENTENCE> > <PRESENT> THE <KNOWN-COUNT-NOUN>  
  
// (DISPLAY <KNOWN-COUNT-NOUN>),
```

which is used to match such inputs as

```
What are the ships?  
Show me the officers.  
List the carriers.
```

The metasympol <PRESENT> matches the underlined portion of these inputs, "THE" matches "the," and <KNOWN-COUNT-NOUN> matches the last word in each of the examples. (Count nouns refer to discrete objects that can be counted, such as ships and ports. NANOKLAUS does not deal with mass nouns, e.g., "steel" and "water.")

Whenever a sentence is found that matches this pattern, the function DISPLAY is called with the value of <KNOWN-COUNT-NOUN>. This function thereupon retrieves from the data base and displays to the user all currently known instances of objects that might be referred to by the <KNOWN-COUNT-NOUN>.

Although most of the linguistic processing performed by the system follows fairly standard practice, the pragmatic grammar is distinguished by its explicit identification of a number of syntactic structures used principally to define new concepts. As an oversimplified example, NANOKLAUS might be thought of as looking for the syntactic pattern

-----  
\* The rules used by NANOKLAUS are much more linguistically motivated than those used in LADDER. In our discussion, we have suppressed the complexity of the rules and response functions actually used, so as to characterize the essence of the methodology more succinctly.



<SENTENCE> > <A> <NEW-WORD> <BE> <A> <KNOWN-COUNT-NOUN>

to account for such inputs as

A CARRIER IS A SHIP.

The system's definition of the category <NEW-WORD> allows <NEW-WORD> to match any LISP atom (or atom sequence). The syntactic category <KNOWN-COUNT-NOUN> originally contains only count nouns associated with seed concepts, such as "thing," "person," "physical object," and the like.

When one of NANOKLAUS's concept-defining patterns is recognized, an assimilation procedure associated with the pattern is called. This procedure usually adds new facts to the system's set of wffs and generates new entries in its lexicon. The various assimilation procedures also have provisions for interacting with the user/teacher. Response generation is accomplished by means of preprogrammed phrases and templates.

For example, when the routine associated with the last pattern shown above is called, it first makes a new lexical entry in category <KNOWN-COUNT-NOUN> for the atom matched by the <NEW-WORD>. In this case, "CARRIER" becomes a new <KNOWN-COUNT-NOUN>. Then the routine creates a new sort predicate\* for CARRIER in the system's knowledge base and enters the assertion that "for every x, if x is a CARRIER then x is a SHIP." Finally the routine asks questions of the user to determine relationships between the sorts of objects that are CARRIERS and other sorts of objects that are SHIPS. Interactions 7 and 23 of the transcript illustrate this interaction.

-----  
\* A sort predicate is a one-argument predicate that indicates what kind, class, or sort of thing an object is. For example, CARRIER and SHIP are sort predicates in the formula (ALL X) ((CARRIER X) > (SHIP X)). We use the word "sort" rather than "class" to avoid the connotation that a sort predicate is associated with a set of objects, which could be extensionally defined.

C. NANOKLAUS's Knowledge Base and Deduction Component

First-order logic was chosen as the basis for NANOKLAUS's knowledge representation scheme because of its generality and because of the computational soundness and power of problem-solving systems that use it.

1. Typical Wffs Used by NANOKLAUS

The introduction of sort predicates mentioned above provides an example of how NANOKLAUS makes use of constructs from first-order logic. Whenever NANOKLAUS learns a new count noun, it creates a new single-place predicate to characterize objects of the associated sort. For example, we have seen that, upon learning the concept of a carrier, NANOKLAUS creates a predicate called "CARRIER" and asserts the fact

(ALL X) ((CARRIER X) > (SHIP X)).

As another example of NANOKLAUS's use of formulas in logic, when NANOKLAUS learns that carriers and submarines are distinct sorts of objects, it effectively asserts the fact

(ALL X) (NOT (AND (CARRIER X) (SUBMARINE X))).

When NANOKLAUS learns of a new individual, such as the JFK (see Interaction 26), it creates a new constant term in the logic system and relates it to one of the sorts, namely

(KITTYHAWK JFK).

Upon learning a new verb, such as "command" (see Interaction 18), NANOKLAUS creates a new predicate with the proper number of argument positions and constrains the domains of those arguments by assertions such as

(ALL X Y) ((COMMAND X Y) > (AND (OFFICER X) (SHIP Y))).

Most of the assertions made by a user are translated into propositions in a straightforward manner. Eg., "Brown commands the Saratoga" (see Interaction 43) produces

(COMMAND BROWN SARATOGA).

## 2. Consistency

NANOKLAUS checks each new fact as it is asserted to determine whether it is consistent with its previous knowledge. This gives rise to the behavior shown in Interactions 27, 44 and 45 of the transcript. NANOKLAUS currently has no provision for unlearning. Therefore, if a new assertion causes an inconsistency because a previous assertion was not correct, there is no provision for withdrawing the incorrect assertion.

## 3. More Reasons for Using First-Order Logic

The notion of using first-order logic in combination with automatic deduction as the basis of an intelligent system dates back to the very beginning of AI research. Newell and Simon [15] published a paper on The Logical Theorist in 1956, and McCarthy, in his 1959 "Advice Taker" proposal (republished in 1968 in [11]), suggested using such a combination as the basis of a system capable of commonsense reasoning.

Following a vigorous start, work on the use of logic as a basis for AI systems fell on hard times during the 1960s and early 70s after experimentation by Green [4] and by others showed that the computational effort required to solve problems in first-order logic using Robinson's [16] resolution principle grows exponentially with the number of wffs used in the axiomatization of a domain.

However, more recent work, such as that of Hayes [5], Kowalski [9], Moore [13], and Weyhrauch [18], has suggested how control information may be used to increase the efficiency of the deduction process.

But our main motivation for using first-order logic is that KLAUS systems are incremental learning systems and therefore must be capable of dealing with incomplete knowledge. As pointed out by Moore [14],

Any knowledge representation formalism that is capable of handling the kinds of incomplete information people can understand must at least be able to

Say that something has a certain property without saying which thing has that property:

SOME (X) P(X)

Say that everything in a certain class has a certain property without saying what everything in that class is:

ALL (X) (P(X)  $\rightarrow$  Q(X))

Say that at least one of two statements is true without saying which statement is true:

P OR Q

Explicitly say that a statement is false, as distinguished from simply not saying that it is true:

NOT(P)

Any representation formalism that has these capabilities will be, at the very least, an extension of classical first-order logic, and any inference system that can deal adequately with these kinds of generalizations will have to have at least the capabilities of an automatic-deduction system.

#### D. Acquisition Procedures: Using Dialogue to Aid Assimilation

By and large, it is unreasonable to expect users to volunteer all the information NANOKLAUS needs to assimilate a new concept. In particular, users cannot be expected to know what conclusions NANOKLAUS will draw about a newly taught concept from its previous knowledge, since they know neither the details of its current state of knowledge nor the even more esoteric details of its assimilation procedures. NANOKLAUS must ask the user explicitly for the information it needs. Therefore, whenever a new concept (or word) is presented to NANOKLAUS, a special procedure is called that temporarily assumes control of the dialogue, prompting the user for whatever additional information it may require to assimilate the new concept.

NANOKLAUS must phrase its questions so as to make them readily understandable by people unfamiliar with computers or linguistics. This introduces a number of human-engineering considerations. The acquisition of new verbs offers a cogent illustration of the problem.

English verbs are highly idiosyncratic. Consequently, making proper entries for them in a lexicon is a formidable task. Among other

criteria, one must ascertain whether a verb is transitive, whether it can be used in the passive voice, whether its indirect object can become the object of a FOR or TO prepositional phrase, whether it is reflexive or nonreflexive, and how the syntactic cases of its arguments may be "normalized" when the verb appears in different syntactic constructions. NANOKLAUS's users cannot be expected to describe verbs in linguistic terms; therefore, to elicit the same information, the system must ask a series of questions that users can understand. Interactions 18 and 19 in the transcript are typical verb acquisition exchanges. While the dialogue is moderately natural and can be used by a nonlinguist, there is obviously considerable room for improvement in its design.

#### E. Some Major Limitations of NANOKLAUS Technology

Many of the major limitations of NANOKLAUS can be seen simply by reading through the transcript and noting that, although English is being used, the conversation is nevertheless highly stylized and artificial. For the most part, NANOKLAUS is limited to learning about very concrete types of objects and their interrelations. It has no capacity to deal with time, process, causality, intent, want, belief, or judgment. This, of course, severely limits its range of application.

Even when considering concrete objects and their interrelations, NANOKLAUS can deal with only highly specific statements. For example, NANOKLAUS has no capacity to deal with analogy, as in

A SOFTBALL IS LIKE A BASEBALL, BUT BIGGER AND SOFTER.

In general, the interpretation of information volunteered by people about new domains may necessitate deep reasoning and require information from other domains. Much of the volunteered information may contain inconsistencies that the user himself has no way (or particular reason) to resolve. The NANOKLAUS system represents a starting point for work on learning-by-being-told; still, it barely scratches the surface of a vast body of difficult problems.

### III MORE TECHNICAL DETAILS

In this section we present additional details about some of the more interesting aspects of the NANOKLAUS system.

#### A. NANOKLAUS's Sort Hierarchy

NANOKLAUS's knowledge representation system uses a many-sorted, first-order logic that combines features from [13] and [7]. The backbone of the system is a treelike data structure reflecting the hierarchy of sorts used by the system (see Figure 1). The data structure maintains information about the immediate ancestors and descendants of each sort, including whether an ancestor is exhausted (spanned) by some or all of its descendants, and whether two or more sibling sorts are mutually exclusive. The sort hierarchy is a "tangled" tree, i.e., any given sort may have multiple ancestors. It is no accident that a sort hierarchy should serve as the primary data structure for an English-based acquisition system. "ISA" hierarchies are used by many natural-language processing systems; it appears that something very similar to a sort hierarchy plays a central role in the way humans organize their knowledge [10].



Straightforward utilization of sort information is illustrated throughout the transcript in the appendix, but especially in Interactions 7 to 14 and 25 to 29.

The sort hierarchy is implemented by means of a special-purpose network representation indicating entailments of systems of unary predicates. Facts stored in this representation can be used exactly like the other facts in the fact store, but are organized to support rapid access to sort information--which is used in the language subsystem as well as in the deduction subsystem. Whenever syntactic combinations are proposed, the parser uses sort information to determine if noun phrases are valid arguments of verbs, adjectives, and relational nouns--thus reducing ambiguity. The response generator consults the sort hierarchy when selecting pronouns for anaphoric reference. If a referent is described as a MALE or a FEMALE, the nominative singular pronoun chosen will be "he" or "she," respectively; if not described as either, but known to be a member of the set of PERSONS, the phrase "he or she" will be used, otherwise "it" will be selected. This can be seen in Interactions 71 and 72 in the transcript.

Sort information is also used to assign default sorts to omitted arguments of verbs. For instance, if an assertion is made that a particular officer commands, NANOKLAUS knows from the acquisition dialogue it had with its teacher that the officer commands something, and that that thing must be of sort SHIP (see Interaction 45).

Measures, or dimensioned quantities, occupy a distinguished place in NANOKLAUS's sort hierarchy; MEASUREs are a distinct sort of THINGs, comprised of LENGTHs, WEIGHTs, PRICEs, etc. The user can define additional sorts of measures. He can also define UNITs in which MEASUREs are measured, as well as the conversion factors between different units of the same measure (see Interaction 5). NANOKLAUS can perform conversion when answering questions (compare Assertion 47 and Questions 58 and 59), although its arithmetic capabilities are quite limited.



## B. NANOKLAUS's Verb System

One of NANOKLAUS's strengths is its ability to deal with a large number of syntactic variations in verb usage. For example, facts asserted in active voice may be queried in both active and passive voice. In general, NANOKLAUS translates clauses into internal structures of the form

(VERB-PREDICATE Arg1 Arg2 Arg3),

using information about permissible syntactic patterns in which the clause's verb can occur.

The basic verb patterns handled by NANOKLAUS are summarized in Figure 2. [NANOKLAUS does not handle modal verbs (e.g., "want" and "know"), or verbs with adverbial particles (e.g., "pick up," "preside over"), or sentential objects (e.g., "The captain requested that the ship change course").] The objective of NANOKLAUS's verb acquisition dialogues (Interactions 18 and 19) is primarily to determine which patterns may be used with a new verb. The system does not need to ask about each pattern. For example, if pattern A3 is not used with a given verb, patterns A3D, A3W, P3, P3', and P3D are automatically ruled out.

---

Pattern	Arg1 (A)	Arg2 (B)	Arg3 (C)	Sample Sentence of this Pattern
A3	Subj	NP1	NP2	A Joe gave B Sue C a-ball.
A3D	Subj	NPd	NP	A Joe gave C a-ball to B Sue.
A3W	Subj	NP	NPw	A Joe supplied B Sue with C a-ball.
P3	(NPb)	NP	Subj	C a-ball was given B Sue (by A Joe).
P3'	(NPb)	Subj	NP	B Sue was given C a-ball (by A Joe).
P3D	(NPb)	NPd	Subj	C a-ball was given to B Sue (by A Joe).
A2	Subj	-	NP	A Joe wrote C a-letter.
A2'	Subj	NP	-	A Joe wrote B Sue.
A2D	Subj	NPd	-	A Joe wrote to B Sue.
P2	(NPb)	-	Subj	C a-letter was written (by A Joe).
P2'	(NPb)	Subj	-	B Sue was written (by A Joe).
A1	Subj	-	-	A Joe wrote.
A1'	-	-	Subj	C a-vase broke. (from A2 of "A Joe broke C a-vase.)

---

Pattern names indicate active (A) or passive (P) voice, the number of top-level noun phrases occurring in the sentence, and (in some cases) an indication of a dative noun phrase (D) moved into a "to" or "for" prepositional phrase, or an indication of a "with" (W) prepositional phrase.

**Notation:**

- Subj the surface subject of the sentence.
- NP1 the first unmarked <NP> in the verb phrase.
- NP2 the second unmarked <NP> in the verb phrase.
- NP the only unmarked <NP> in the verb phrase.
- NPb <NP> marked by the preposition "by." May be omitted.
- NPw <NP> marked by the preposition "with." (not instrumental)
- NPd <NP> marked by either "to" or "for." (dative)

Figure 2: Verb Usage Patterns

---

To appreciate the range of variation in English verbs, consider the following sentences, each of which describes a situation using a ditransitive verb construction (the A3 pattern of Figure 2). The symbol '\*' marks ungrammatical sentences; '?' marks sentences whose grammaticality is questionable.

A3            John cooked Mary the fish.  
 ---           ? John supplied the school the books.  
               \* John ran Mary the machine.  
                       (I.e., John ran it for Mary.)  
               John served Mary the fish.  
               John caught Mary the fish.

The following sentence sets are variations of the above, using the same verbs but in different syntactic patterns. In each set the same pattern is used throughout. Notice that not all verbs can be used grammatically in each pattern and that, moreover, some patterns act to describe a different situation from the ones described above (marked by '-').

A3w         - John cooked Mary with the fish.  
 ---         John supplied the school with the books.  
              - John ran Mary with the machine.  
              - John served Mary with the fish.  
              - John caught Mary with the fish.

A3d         John cooked the fish for Mary.  
 (for)       ? John supplied the books for the school.  
 -----     John ran the machine for Mary.  
              - John served the fish for Mary.  
              John caught the fish for Mary.

A3d         \* John cooked the fish to Mary.  
 (to)         John supplied the books to the school.  
 -----     \* John ran the machine to Mary.  
              John served the fish to Mary.  
              \* John caught the fish to Mary.

A2'         - John cooked Mary.  
 ---         John supplied the school.  
              - John ran Mary.  
              John served Mary.  
              - John caught Mary.

A1           John cooked.  
 --           ? John supplied.  
              - John ran.

John served.  
? John caught.

A1'        The fish cooked.  
---        \* The books supplied.  
           The machine ran.  
           \* The fish served.  
           \* The fish caught.

There are two principal steps in the translation of a clause expressed in English into a proposition in first-order logic. First, syntactic analysis recognizes which of the various verb patterns is being used. Then syntactic cases (such as SUBJ and NP1) are mapped into argument positions for the predicate associated with the verb sense. For example,

JOHN GAVE SAM FIDO

is in the A3 pattern. According to the table of Figure 1, its SUBJECT JOHN is therefore mapped to Arg1, its indirect object (or NP1) to Arg2, and its direct object (or NP2) to Arg3. The end result is the proposition

(GAVE JOHN SAM FIDO)

The related sentence

FIDO WAS GIVEN TO SAM

is in the P3D pattern, so the subject is mapped to Arg3 and the dative SAM to Arg2, resulting in (GAVE -- SAM FIDO), which in turn is converted into (EXIST X)(GAVE X SAM FIDO).

### C. Relating KLAUS Systems to Conventional DBMSs

In several ways, basic KLAUS systems such as NANOKLAUS are similar to conventional DBMSs (data-base-management systems) in that they are intended to file, sort, selectively recall, and display data in various formats. However, DBMSs are systems for dealing with data structures through a formal command/query language, whereas a KLAUS is a system for learning and manipulating concepts through interactions in English.

Furthermore, conventional DBMSs store only single-fact sentences, such as

THE KENNEDY IS OWNED BY THE U.S.

Facts involving logical connectives, such as "OR" in

THE KENNEDY IS EITHER IN PORT OR AT SEA,

or involving quantification, such as

ALL CARRIERS CARRY DOCTORS,

are not ground literals and cannot be represented explicitly in a conventional DBMS. They can, however, be represented, discussed and reasoned about in the first-order logic supporting a KLAUS.\*

#### IV CONCLUSIONS AND DIRECTIONS FOR FUTURE WORK

NANOKLAUS is best described as a fragile, proof-of-concept system that was built to establish the feasibility of achieving the broader KLAUS goals. The natural-language research group at SRI International is now implementing a more sophisticated KLAUS, called MICROKLAUS. MICROKLAUS will cover a broader range of English constructions by replacing NANOKLAUS's LIFER-based pragmatic grammar with Jane Robinson's [17] DIAGRAM grammar. MICROKLAUS will also feature a more efficient deduction system than the one used by its predecessor.

Our work on the subject of knowledge acquisition has only recently begun and there are many capabilities that we have yet to address. For example, NANOKLAUS has no provision for learning by analogy, acquiring or reasoning about the internal structures of processes, dealing with causality, handling mass terms, allowing users to change their minds

---

\* As of July, 1981, universal quantification was operational in the MICROKLAUS system, and both universal quantification and disjunction were operational in its deduction and representation components.

about previously asserted "facts," or dealing with multiple senses of words.

The KLAUS project is not limited to the study of knowledge acquisition, but is intended to provide technology for a system that combines a knowledge of how to use various software packages with an ability to learn facts about new domains. It is intended that an advanced KLAUS system act as a kind of broker between multiple external agents, one of whom happens to be the user.

The following deceptively simple hypothetical dialogue illustrates the value and the basic characteristics of such an advanced KLAUS:

1. User: > DO I HAVE NEW MAIL?
2. KLAUS: You have one message  
FROM: HENDRIX  
SUBJECT: NSF funds have been cut by Congress  
Do you want to read it now?
3. User: > YES
4. KLAUS: (Presents message about funding cuts at NSF.)
5. User: > WHO ARE THE PROJECT LEADERS ON OUR NSF GRANTS?
6. KLAUS: The following table presents the PIs of NSF projects in the AI Center.

PI	PROJECT
Fischler	1202
Grosz	6194
Robinson	7483
Waldinger	7409
Witkin	8682

7. User: > FORWARD THE MESSAGE TO THEM.

In this dialogue, the user first asks about his mail (Utterance 1).<sup>\*</sup> Then the user says he wants to read his message, whereupon KLAUS

<sup>\*</sup> This dialogue does not require KLAUS to understand the title or body of the message; however, KLAUS must understand that there was a message and that it was from Hendrix.

displays it. It is about funding cuts at NSF. The user decides to find out who the project leaders of the relevant projects are, so he asks Question 5. Subsequently, in Request 7, he asks that the message be forwarded to these leaders.

Most of the capabilities required for this dialogue can be found in existing systems. However, no system yet built is able to support the entire interaction as shown, because all systems to date lack a controlling and coordinating top level that would enable a single system to

- \* Interact with the user in natural language.
- \* Decide which of several subsystems should be invoked to meet the user's needs.
- \* Automatically create and execute the calling sequences to invoke those subsystems.
- \* Maintain a dialogue context with the user that reflects and integrates the various local contexts that have been established by and for the use of subsystems.

In other words, present systems lack a crucial level of awareness of the interrelationships and potential interactions of the various software packages, the user, and the domain of application. In short, processing the sample dialogue requires knowledge of the domain of application, and of computer programs, as well as knowledge of the way these pieces fit together in a larger framework. How to acquire, represent, and apply such knowledge in a computer-based system are major problems for future research.

#### ACKNOWLEDGMENTS

The deduction system supporting NANOKLAUS was developed in large part by Mabry Tyson, with Robert Moore, Nils Nilsson, and Richard Waldinger as advisors. Beth Levin made major contributions to NANOKLAUS's verb acquisition algorithm. Barbara Grosz, Earl Sacerdoti, and Daniel Sagalowicz provided very useful critiques of early drafts of this chapter.

## APPENDIX

### Training NANOKLAUS About Aircraft Carriers

What follows is a transcript of a dialogue with the NANOKLAUS system. The consecutive numbers at the left of some lines are NANOKLAUS prompts; the sentences on those lines are typed by the user. Because subsequent utterances by NANOKLAUS and the user are clearly distinguishable, they are not specially marked. Some of the interaction has been deleted to shorten the transcript. Sentences in braces are descriptive annotations.

@nanoklaus

Welcome to Nanoklaus !  
Just say 'Help' if you need help.

1\_what do you know about?

{The user asks NANOKLAUS what classes of things have been described to it.}

I know about PHYSICAL OBJECTS, PEOPLE, MEN, WOMEN, MEASURES, and THINGS (of course).

2\_A length is a measure

{The user defines LENGTH.}

You're saying that anything that is a LENGTH is also a MEASURE.

Ok, now I understand LENGTH.

3\_A foot is a unit of length

{The user defines FOOT.}

FOOTS are the first units of LENGTH you've defined.

Now I understand FOOT.

4\_The plural of foot is feet

FEET - got it. Thanks.

5\_A meter is a unit of length

How is it related to FOOT? There are 3.3 feet in a meter

{Whenever an additional unit of a measure is defined, NANOKLAUS requests the factor for conversion to one of the previously defined units.}

Now I understand METER.

6\_A physical object has a length

{The user gives the domain over which an attribute named LENGTH is defined, and implicitly indicates that the value of this attribute must be an element of the set of lengths, defined in (2).}

So PHYSICAL OBJECTS have LENGTHS.

Must a PHYSICAL OBJECT have exactly one LENGTH? yes

{NANOKLAUS determines that the relationship between physical objects



and lengths is functional.}

7 A ship is a physical object

You're saying that anything that is a SHIP is also a PHYSICAL OBJECT.

Is PERSON a proper subclass of SHIP? no {When a new subset of an existing set is created, NANOKLAUS queries the user to determine how it relates to that set, as well as to other previously defined subsets of that set.}

Is PERSON necessarily composed of entirely different members from SHIP? yes

Do PERSON and SHIP span the set of all PHYSICAL OBJECTS? no

Ok, now I understand SHIP.

8 A ship is a unit {Sorts may be subsets of more than one set.}

You're saying that anything that is a SHIP is also an UNIT.

What is an UNIT? a thing {NANOKLAUS always wants to know how the set and superset relate to the existing taxonomy (the taxonomy is always a connected graph). Note that it generated the incorrect indefinite article in the question, but see Interaction 9 below.}

You're saying that anything that is an UNIT is also a THING.

Which of the following, if any, are proper subclasses of UNIT:

PHYSICAL OBJECT or MEASURE? neither of them

Which of the following classes, if any, could never have any members in common with UNIT:

PHYSICAL OBJECT or MEASURE ? neither of them

Ok, now I understand UNIT.

Ok, now I understand SHIP.

9 A unit {The user corrects NANOKLAUS's English.}

OK, A UNIT

10 A submarine is a ship {Feedback from NANOKLAUS, such as

11 A destroyer is a ship seen in Interaction 8, is omitted

12 A carrier is a ship for the sake of brevity.}

13 A Kittyhawk is a carrier

14 A Forrestal is a carrier

You're saying that anything that is a KITTYHAWK is also a CARRIER.

Is KITTYHAWK a proper subclass of FORRESTAL? no

Is KITTYHAWK necessarily composed of entirely different members from FORRESTAL? yes

Do KITTYHAWK and FORRESTAL span the set of all CARRIERS? no

Ok, now I understand KITTYHAWK.

15 A Hermes is a carrier

16 A task group is a unit

17 An officer is a person

18 An officer can command a ship {Here the verb COMMAND is defined as mapping into a predicate over

Give the -ed and -en forms of COMMAND (for example, for GO: WENT GONE):

commanded commanded (officer, ship) pairs.)  
If an OFFICER can COMMAND a SHIP something, type in the most general description of that thing, otherwise type <carriage return>

{NANOKLAUS tried to determine if COMMAND doesn't in fact take a third argument.}

If an OFFICER can COMMAND something a SHIP, then please type in the most general description of that thing, otherwise type <carriage return>

If an OFFICER can COMMAND something to a SHIP, then please type in the most general description of that thing, otherwise <carriage return>

{In this context, "to" marks an indirect object, not the destination of a verb of motion.}

If an OFFICER can COMMAND a SHIP to something, then please type in the most general description of that thing, otherwise <carriage return>

Can a SHIP be COMMANDED by an OFFICER? yes

{NANOKLAUS acquires information enabling it to map into a canonical predicate calculus form from passive occurrences of the verb and occurrences in which some of the noun phrases are omitted.}

Can a SHIP COMMAND? no

Can an OFFICER COMMAND? yes

19 An officer can assign a task group a unit

{Another example, this time a ditransitive verb.}

Give the -ed and -en forms of ASSIGN

(for example, for GO: WENT GONE):

assigned assigned

Can an OFFICER ASSIGN a UNIT TO a TASK GROUP? yes

Could a UNIT be ASSIGNED a TASK GROUP by an OFFICER? yes

Can an OFFICER ASSIGN a UNIT with a TASK GROUP? no

Can an OFFICER ASSIGN a TASK GROUP? no

Can an OFFICER ASSIGN? yes

20 Some ships can be nuclear

{NANOKLAUS is taught an adjective, which it maps into a unary predicate whose domain is ships.}

You're saying that at least one SHIP could be nuclear.

21 A nationality is a thing

22 A ship has a nationality

So SHIPS have NATIONALITIES.

Must a SHIP have exactly one NATIONALITY? yes

23 A rank is a thing

You're saying that anything that is a RANK is also a THING.

Which of the following, if any, are proper subclasses of RANK:

PHYSICAL OBJECT, LEGAL PERSON, MEASURE, UNIT, or NATIONALITY? none of them

Which of the following classes, if any, could never have any members in common with RANK:

PHYSICAL OBJECT, LEGAL PERSON, MEASURE, UNIT, or NATIONALITY ? all of them  
Ok, now I understand RANK.

24 An officer has a rank

25 A carrier has a doctor who is a person {Here the attribute named  
So CARRIERS have PEOPLE for DOCTORS. DOCTOR takes values that  
Must a CARRIER have exactly one DOCTOR? no must be members of the set  
of persons.}

26 The JFK is a Kittyhawk {First introduction of an  
You're saying that JFK is one instance of a KITTYHAWK. individual.}  
Ok, now I know about JFK.

27 The JFK is a Forrestal

You're saying that JFK is one instance of a FORRESTAL.  
Sorry, but JFK is an individual of the following types: KITTYHAWK,  
and one of them conflicts with FORRESTAL.

{NANOKLAUS used its knowledge of  
subset relationships to discover  
and then report an inconsistency  
in its user's assertions.}

28 The Saratoga is a Forrestal

29 The Enterprise is a carrier

30 The Bulwark is a Hermes

31 US is a nationality

32 UK is a nationality

33 US is the nationality of the JFK

Proving (US is the nationality of JFK)

Asserting that (US is the nationality of JFK)

{Asserts the value of  
an attribute. Note that,  
instead of blindly entering  
the fact in the fact store,  
NANOKLAUS first checks as to  
whether it already knows  
the fact.}

34 The nationality of the Saratoga is US

35 The nationality of the Enterprise is US

36 The nationality of the Bulwark is UK

37 Moffet is an officer

38 Brown is an officer

39 Captain is a rank

40 The rank of Brown is captain

41 Captain is the rank of Moffet

42 The JFK is commanded by Moffet

Proving (MOFFET commanded JFK)

Asserting that (MOFFET commanded JFK)

{Uses verbs to describe an  
event or situation. Note trans-  
formation from passive to active  
voice by NANOKLAUS.}

43 Brown commands the Saratoga

Proving (BROWN commanded SARATOGA)

Asserting that (BROWN commanded SARATOGA)

{NANOKLAUS employs an untensed logic;  
its canonical expression of its  
knowledge is always past tense.}

44 Moffet commands the JFK.

Proving (MOFFET commanded JFK)

I already knew that.

{NANOKLAUS detected that it had  
previously been told this,  
although in a different voice.}

45 Brown commands.

Proving (BROWN commanded Some ship) {NANOKLAUS filled in the omitted argument and then determined that it had been told of an event or situation that could conceivably have been a description of the same event or situation.}

Is any one of the following statements true concerning this event:  
the SHIP is SARATOGA ? Yes

46 Brown commands a Forrestal.

Proving (BROWN commanded Some forrestal)

Is any one of the following statements true concerning this event:  
the FORRESTAL is SARATOGA ? Yes.

47 1072 feet is the length of the JFK

Asserting that (1072 FEET is [equal to] the LENGTH of JFK)

48 The length of the Bulwark is 650 feet

49 ctg18 is a task group

50 ctg36 is a task group.

51 Moffet assigned the Saratoga to ctg36

Proving (MOFFET assigned CTG36 SARATOGA)

Asserting that (MOFFET assigned CTG36 SARATOGA)

52 The Saratoga was assigned to ctg18 by Brown

Proving (BROWN assigned CTG18 SARATOGA)

Asserting that (BROWN assigned CTG18 SARATOGA)

53 The Enterprise is a nuclear ship

{Uses a predicate adjective.}

54 What are the ships?

{The user interrogates

ENTERPRISE, JFK, SARATOGA, AND BULWARK.

NANOKLAUS to verify that it has assimilated what it

has been told. Note that the user has not explicitly described any of the answers as ships.}

55 Who commands the Kittyhawks?

(MOFFET JFK).

56 Who commands the JFK?

MOFFET.

57 Brown commands what?

SARATOGA.

58 What is the length of the JFK?

1072 FEET.

59 How many meters is the length of the JFK?

Proving (Some length is [equal to] the LENGTH of JFK)

324.8485.

{Using the unit conversion factor.}

60 What is the nationality of the Saratoga?

US.

62 Brown assigned what ships?

Proving (BROWN assigned Some task group Some ship)

(CTG18 SARATOGA).

{I.e., Brown assigned CTG18 Saratoga.}

63 Who assigned ships to ctg36?

Proving (Some officer assigned CTG36 Some ship)

(MOFFET SARATOGA).

64 Who assigned ctg18 the Saratoga?

Proving (Some officer assigned CTG18 SARATOGA)  
BROWN.

65 Who assigned the Saratoga?

Proving (Some officer assigned Some task group SARATOGA)  
(MOFFET CTG36) and (BROWN CTG18).

66 Which ships are nuclear?

Proving (Some ship is a nuclear ship)  
ENTERPRISE.

67 What do you know about carriers?

US is the nationality of JFK.

US is the nationality of SARATOGA.

US is the nationality of ENTERPRISE.

UK is the nationality of BULWARK.

MOFFET commanded JFK.

{NANOKLAUS has access to its  
fact store indexed by sort,  
as well as by predicate.  
Here it reports all the  
atomic facts it has about  
carriers.}

BROWN commanded SARATOGA.

1072 FEET is [equal to] the LENGTH of JFK.

the LENGTH of BULWARK is [equal to] 650 FEET.

MOFFET assigned CTG36 SARATOGA.

BROWN assigned CTG18 SARATOGA.

ENTERPRISE is a nuclear ship.

69 Now describe ship

SHIP:

A SHIP is a PHYSICAL OBJECT and UNIT.

Therefore it is also a THING.

SUBMARINES and CARRIERS are SHIPS.

SUBMARINES and CARRIERS cannot have any members in common.

SHIP has the following attributes and types:

NATIONALITY.

SHIPS can be NUCLEAR.

71 Please describe Moffet then describe Brown.

{Here it describes all the  
generic and sort information  
it has about a given sort.}

{In reponse to this  
request, NANOKLAUS  
displays all the facts it  
knows about a particular  
individual.}

Moffet is an officer. His or her rank is CAPTAIN. MOFFET  
commanded JFK. MOFFET assigned CTG36 SARATOGA.

Brown is an officer. His or her rank is CAPTAIN. BROWN commanded  
SARATOGA. BROWN assigned CTG18 SARATOGA.

72 Now tell me about Saratoga.

Saratoga is a forrestal. Its nationality is US. BROWN

commanded SARATOGA. MOFFET assigned CTG36 SARATOGA. BROWN assigned  
CTG18 SARATOGA.

76 Who commands what?

Proving (Some officer commanded Some ship)  
(MOFFET JFK) and (BROWN SARATOGA).

77 good bye

See you later

## REFERENCES

1. R. Davis, "Interactive Transfer of Expertise: Acquisition of New Inference Rules," Proc. 5th International Joint Conference on Artificial Intelligence, Cambridge, Massachusetts, pp. 321-328 (August 1977).
2. T. G. Dietterich and R. S. Michalski, "Learning and Generalization of Characteristic Descriptions: Evaluation Criteria and Comparative Review of Selected Methods," Proc. 6th International Joint Conference on Artificial Intelligence, Tokyo, Japan, pp. 223-231 (August 1979).
3. J. A. Fodor, The Language Of Thought, pp. 124-156, (Thomas Y. Crowell Company, New York, New York, 1975).
4. C. Green, "Theorem-Proving by Resolution as a Basis for Question-Answering Systems," in Machine Intelligence 4, B. Meltzer and D. Michie, eds., pp. 183-205 (American Elsevier Publishing Company, Inc., New York, New York, 1969).
5. P. J. Hayes, "Computation and Deduction," Proc. 2nd Symposium on Mathematical Foundations of Computer Science, Czechoslovak Academy of Sciences, pp. 105-116 (1973).
6. G. G. Hendrix, "The LIFER Manual: A Guide to Building Practical Natural Language Interfaces," Technical Note 138, Artificial Intelligence Center, Stanford Research Institute, Menlo Park, California (February 1977).
7. G. G. Hendrix, "Encoding Knowledge in Partitioned Networks," in Associative Networks-The Representation and Use of Knowledge in Computers, N. V. Findler, ed. (Academic Press, New York, New York, 1979).
8. G. G. Hendrix, E. D. Sacerdoti, D. S. Sagalowicz and J. Slocum, "Developing a Natural Language Interface to Complex Data," ACM Transactions on Database Systems, Vol. 3, No. 2, pp. 105-147 (June 1978).

9. R. Kowalski, "Predicate Logic as a Programming Language," in Information Processing 74, pp. 569-574 (North-Holland Publishing Company, Amsterdam, The Netherlands, 1974).
10. P. H. Lindsay and D. A. Norman, Human Information Processing. (Academic Press, New York, New York, 1972.)
11. J. McCarthy, "Programs with Common Sense," in Semantic Information Processing, M. Minsky, ed., pp. 403-417 (The MIT Press, Cambridge, Massachusetts, 1968).
12. T. M. Mitchell, "Version Spaces: a Candidate Elimination Approach to Rule Learning," Proc. 5th International Joint Conference on Artificial Intelligence, Cambridge, Massachusetts, pp. 305-310 (August 1977).
13. R. Moore, "Reasoning from Incomplete Knowledge in a Procedural Deduction System," MIT Artificial Intelligence Laboratory, AI-TR-347, Massachusetts Institute of Technology, Cambridge, Massachusetts (1975).
14. R. Moore, "Automatic Deduction for Commonsense Reasoning: An Overview." To appear in the Handbook of Artificial Intelligence, A. Barr and E.A. Feigenbaum, eds. (William Kaufman, Inc., Los Altos, California, 1982).
15. A. Newell and H. A. Simon, "The Logic Theory Machine," IRE Transactions on Information Theory, Vol. IT-2, No. 3, pp. 61-79 (September 1956).
16. J. A. Robinson, "A Machine-Oriented Logic Based on the Resolution Principle," Journal of the Association for Computing Machinery, Vol. 12, No. 1, pp. 23-41 (January 1965).
17. J. J. Robinson, "DIAGRAM: an Extendable Grammar for Natural Language Dialogue," Technical Note 205, Artificial Intelligence Center, SRI International, Menlo Park, California (February 1980).
18. R. W. Weyhrauch, "Prolegomena to a Theory of Mechanized Formal Reasoning," Artificial Intelligence, Vol. 13, Nos. 1, 2, pp. 133-170 (April 1980).

19. P. H. Winston, "Learning Structural Descriptions from Examples," Chapter 5 in The Psychology of Computer Vision, P. H. Winston, ed. (McGraw-Hill Book Company, New York, New York (1975)).