The specific aim of the TACITUS project is to develop breakdowns of machinery. These interpretation processes for handling casualty reports from messages for entry into a database or an routing and systems for the automatic extraction of information. Thus, we first proceeded from the outside in by examining the concept of wear, as in worn bearings, seeking to define wear, and then to define the concepts we defined wear in terms of, pushing the process back to basic concepts in the domains of space, materials, and force, among others. We then proceeded from the inside out, trying to flesh out the core theories of these domains, as well as the domains of scalar notions, time, measure, orientation, shape, and functionality. Then to test the adequacy of these theories, we began working from the outside in again, spending some time defining, or characterizing, the words related to these domains that occurred in our target set of casreps. We are now working from the inside out again, going over the core theories and the definitions with a fine-tooth comb, checking manually for consistency and adequacy, and proving simple consequences of the axioms on the KADS theorem-prover. This work is described in Hobbs et al.

Domain Knowledge

In all of our work we are seeking general solutions that can be used in a wide variety of applications. This may seem impossible for domain knowledge. In our particular case, we must express facts about the starting air compressor of a ship. It would appear difficult to employ this knowledge in any other application. However, our approach makes most of our work, even in this area, relevant to many other domains. We are specifying a number of "abstract machines" or "abstract systems", in levels, of which the particular device we must model is an instantiation. We define, for example, a closed producer-consumer system. We then define a closed clean fluid producer-consumer system as a closed producer-consumer system with certain additional properties, and at one more level of specificity, we define a pressurized lube-oil system. The specific lube-oil system of the starting air compressor, with all its idiosyncratic features, is then an instantiation of the last of these. In this way, when we have to model other devices, we can do so by defining...

Syntactic and Semantic Translation

Syntactic analysis and semantic translation in the TACITUS project are being done by the DIALOGIC system. DIALOGIC has perhaps as extensive a coverage of English syntax as any system in existence, it produces a logical form in first-order predicate calculus, and it was used as the syntactic component of the TEAM system. The principal addition we have made to the system during the TACITUS project has been a menu-based component for rapid vocabulary acquisition that allows us to acquire several hundred lexical items in an afternoon's work. We are now modifying DIALOGIC to produce neutral representations instead of multiple readings for the most common types of syntactic ambiguities, including prepositional phrase attachment ambiguities and compound noun ambiguities.
them to be the most specific applicable abstract machine that has been defined previously, thereby obviating much of the work of specification. An electrical circuit, for example, is also a closed producer-consumer system.

DEDUCTION

The deduction component of the TACITUS system is the KLAUS Automated Deduction System (KADS), developed as part of the KLAUS project for research on the interactive acquisition and use of knowledge through natural language. Its principal inference operation is nonclausal resolution, with possible resolution operations encoded in a connection graph. The nonclausal representation eliminates redundancy introduced by translating formulas to clause form, and improves readability as well. Special control connectives can be used to restrict use of the formulas to either forward chaining or backward chaining. Evaluation functions determine the sequence of inference operations in KADS. At each step, KADS resolves on the highest-rated link. The resolvent is then evaluated for retention and links to the new formula are evaluated for retention and priority. KADS supports the incorporation of theories for more efficient deduction, including deduction by demodulation, associative and commutative unification, many-sorted unification, and theory resolution. The last of these has been used for efficient deduction using a sort hierarchy. Its efficient methods for performing some reasoning about sorts and equality, and the facility for ordering searches by means of an evaluation function, make it particularly well suited for the kinds of deductive processing required in a knowledge-based natural language system.

LOCAL PRAGMATICS

We have begun to formulate a general approach to several problems that lie at the boundary between semantics and pragmatics. These are problems that arise in single sentences, even though one may have to look beyond the single sentence to solve them. The problems are metonymy, reference, the interpretation of compound nominals, and lexical and syntactic ambiguity. All of these may be called problems in “local pragmatics”. Solving them constitutes at least part of what the interpretation of a text is. We take it that interpretation is a matter of reasoning about what is possible, and therefore rests fundamentally on deductive operations. We have formulated very abstract characterizations of the solutions to the local pragmatics problems in terms of what can be deduced from a knowledge base of commonsense and domain knowledge. In particular, we have devised a general algorithm for building an expression from the logical form of a sentence, such that a constructive proof of the expression from the knowledge base will constitute an interpretation of the sentence. This can be illustrated with the sentence from the casreps

Disengaged compressor after lube oil alarm.

To resolve the reference of *alarm*, one must prove constructively the expression

\[(\exists x) \text{alarm}(x)\]

To resolve the implicit relation between the two nouns in the compound nominal *lube oil alarm* (where *lube oil* is taken as a multiword), one must prove constructively from the knowledge base the existence of some possible relation, which we may call *nn*, between the entities referred to by the nouns:

\[(\exists x,y) \text{alarm}(x) \land \text{lube-oil}(y) \land \text{nn}(y,x)\]

A metonymy occurs in the sentence in that *after* requires its object to be an event, whereas the explicit object is a device. To resolve a metonymy that occurs when a predicate is applied to an explicit argument that fails to satisfy the constraints imposed by the predicate on its argument, one must prove constructively the possible existence of an entity that is related to the explicit argument and satisfies the constraints imposed by the predicate. Thus, the logical form of the sentence is modified to

\[
\ldots \land \text{after}(d,e) \land q(e,x) \land \text{alarm}(x) \land \ldots 
\]

and the expression to be proved constructively is

\[(\exists e) \text{event}(e) \land q(e,x) \land \text{alarm}(x) \land \ldots\]

In the most general approach, *nn* and *q* are predicate variables. In less ambitious approaches, they can be predicate constants, as illustrated below.

These are very abstract and insufficiently constrained formulations of solutions to the local pragmatics problems. Our further research in this area has probed in four directions.

1. We have been examining various previous approaches to these problems in linguistics and computational linguistics, in order to reinterpret them into our framework. For example, an approach that says the implicit relation in a compound nominal must be one of a specified set of relations, such as “part-of”, can be captured by treating *nn* as a predicate constant and by including in the knowledge base axioms like

\[(\forall x,y) \text{part-of}(y,x) \Rightarrow \text{nn}(x,y)\]

In this fashion, we have been able to characterize succinctly the most common methods used for solving these problems in previous natural language systems, such as the methods used in the TEAM system.

2. We have been investigating constraints on the most general formulations of the problems. There are general constraints, such as the Minimality Principle, which states that one should favor the minimal solution in the sense that the fewest new entities and relations must be hypothesized. For example, the argument-relation pattern in compound nominals, as in *lube oil pressure*, can be seen as satisfying the Minimality Principle, since the implicit relation is simply the one already given by the head noun. In addition, we are looking for constraints that are specific to given problems. For example, whereas whole-part compound nominals, like *regulator valve*, are quite common, part-whole compound
In our implementation of the TACITUS system, we are

(3) A knowledge base contains two kinds of know-
ledge, "type" knowledge about what kinds of situations
are possible, and "token" knowledge about what the
actual situation is. We are trying to determine which of
these kinds of knowledge are required for each of the
pragmatics problems. For example, reference requires
both type and token knowledge, whereas most if not all
instances of metonymy seem to require only type know-
ledge.

(4) At the most abstract level, interpretation requires
the constructive proof of a single logical expression
consisting of many conjuncts. The deduction component
can attempt to prove these conjuncts in a variety of
orders. We have been investigating some of these possi-
able orders. For example, one plausible candidate is that
one should work from the inside out, trying first to solve
the reference problems of arguments of predications
before attempting to solve the compound nominal and
metonymy problems presented by those predications. In
our framework, this is an issue of where subgoals for the
deduction component should be placed on an agenda.

IMPLEMENTATION

In our implementation of the TACITUS system, we are
beginning with the minimal approach and building up
slowly. As we implement the local pragmatics oper-
ations, we are using a knowledge base containing only
the axioms that are needed for the test examples. Thus,
it grows slowly as we try out more and more tests. As
we gain greater confidence in the pragmatics operations,
we will move more and more of the axioms from our
commonsense and domain knowledge bases into the
system's knowledge base. Our initial versions of the
pragmatics operations are, for the most part, fairly stand-
ard techniques recast into our abstract framework. When
the knowledge base has reached a significant size, we will
begin experimenting with more general solutions and
with various constraints on those general solutions.

FUTURE PLANS

In addition to pursuing our research in each of the areas
described above, we will institute two new efforts next
year. First of all, we will begin to extend our work in
pragmatics to the recognition of discourse structure. This
problem is illustrated by the following text:

Air regulating valve failed.
Gas regulating valve wouldn't turn over.
Valve parts corroded.

The temporal structure of this text is 3-1-2; first the
valve parts corroded, and this caused the valve to fail,
which caused the engine to not turn over. To recognize
this structure, one must reason about causal relationships
in the model of the device, and in addition one must
recognize patterns of explanation and consequence in the
text.

The second new effort will be to build tools for
domain knowledge acquisition. These will be based on
the abstract machines in terms of which we are presently
encoding our domain knowledge. Thus, the system
should be able to allow the user to choose one of a set of
abstract machines and then to augment it with various
parts, properties and relations.

ACKNOWLEDGMENT

The TACITUS project is funded by the Defense
Advanced Research Projects Agency under Office of
Naval Research contract N00014-85-C-0013, as part of the
Strategic Computing program.

REFERENCE

Hobbs, Jerry R.; Croft, William; Davies, Todd; Edwards, Douglas; and
Laws, Kenneth 1986 Commonsense Metaphysics and Lexical
Semantics. In Proceedings, 24th Annual Meeting of the Association
for Computational Linguistics. New York (June): 231-240.

CALLS

FOR PAPERS, PROPOSALS, AND PICTURES
FOR AWARD NOMINATIONS

FOURTH SYMPOSIUM ON EMPIRICAL FOUNDATION OF
INFORMATION AND SOFTWARE SCIENCES (EFISS)

22-24 October 1986, Georgia Institute of Technology,
Atlanta, Georgia

The purpose of the meeting is to explore subjects and
methods of scientific inquiry of common interest to infor-
mation and software science, and to identify directions of
research that will benefit from the mutual interaction of
the two fields. The main theme of this symposium is
empirical methods of evaluation of man-machine interfaces.

Specific examples of relevant focal topics are: friendli-
ness, portability, sensitivity, fidelity, integrity, fault-toler-
ance, compatibility, modularity, and evolution of
man-machine interfaces; efficiency of interfaces as
communication channels; evaluation of effects of error
propagation through interfaces; modeling man-machine
interfaces.

Contributed papers will be considered also on other
aspects of empiric foundations of information and soft-
ware sciences such as methods of experimental design;
measurement theory and techniques; empirical laws and
theories of information and software sciences; their vali-
dation and verification; experimental data bases; and
software properties and their evaluation and measure-
ment.

All submitted papers will be refereed. Those selected
will be scheduled for presentation and published in the
proceedings of the symposium.

Abstracts of papers (at least 150 words long) are due
by 15 March 1986. Authors will be notified of their