THE FINITE STRING NEWSLETTER

SITE REPORT

THE ESPRIT PROJECT LOKI

Since the beginning of 1985, the Research Unit for Information Science and Artificial Intelligence at the University of Hamburg has been participating in a new project: LOKI – A Logic Oriented Approach to Knowledge and Data Bases. Supporting a Natural User Interface, LOKI derives its funds from the ESPRIT program of the CEC. The main contractors are the Belgian Institute of Management SA (BIM, Bruxelles), the Fraunhofer Institute IAO (Stuttgart), the Cretan Computer Institute (CCI), and Scicon Limited (London). The research unit is only one of several groups participating in the LOKI project.

The following institutions are involved with other parts of the project: SCS Hamburg, the Technical Institute of Munich (TUM), the Fraunhofer Institute IAO (Stuttgart), the Cranfield Institute of Technology (Cranfield), Scicon Limited (London), BIM (Bruxelles), and the Cretan Computer Institute (CCI).

The goal of our work here in Hamburg is the design and implementation of a natural user interface for knowledge and data bases. This interface currently bears the working title "LOQUI". The staff in Hamburg consists of: Walther von Hahn (project leader), Helmut Horacek, Claudius Pyka, Martin Schroeder, and Tom Wachtel. The duration of the first phase is 1 August 1984 – 31 January 1987. Preparations are now being made to apply for a second phase, which is planned for the period from 1 February 1987 through 31 July 1988.

The framework of our work may be sketched as follows:

- Programming in Prolog (BIMProlog).
- Programming is taking place on a SUN Workstation (SUN 2/120) by using the operating system UNIX bsd 4.2 version 2.0.
- The natural language interface (NLI) will be dialogue-oriented, and will have a **kernel** that is independent of application.
- There are plans for a project management system as a pilot application.
- We are developing a semantic representation language LOLA (LOqui LAnguage) for use in analysis and generation.
- As a support for global dialogue strategies, we are planning an explicit dialogue structure with speech act recognition, taking focus into account.
- We are using unification grammar for analysis and generation (in particular, Lexical Functional Grammar, or a version of LFG modified for our purposes).

Presently, we are working on the implementation of the first version of the NLI, which will be completed in early 1986.

The LOQUI group publishes reports and memos, giving information about the state of our work and the research activities of our staff.

More information, including our published material, may be obtained from:

Research Unit for Information Science and Artificial Intelligence
University of Hamburg
Mittelweg 179
D-2000 Hamburg 13
West Germany
Tel. (040) 4123 - 4529/2573/2574/3315

SITE REPORTS

FROM SEVERAL NATURAL LANGUAGE TECHNOLOGY BASE CONTRACTS WITH DARPA'S STRATEGIC COMPUTING PROGRAM

OVERVIEW

Lt. Col. Robert L. Simpson
Information Science Technology Office
Defense Advanced Research Projects Agency

The overall objective of the Strategic Computing (SC) Program of the Defense Advanced Research Projects Agency (DARPA) is to develop and demonstrate a new generation of machine intelligence technology *that* can form the basis for more capable military systems in the future and also maintain a position of world leadership for the US in computer technology. Begun in 1983, SC represents a focused research strategy for accelerating the evolution of new technology and its rapid prototyping in realistic military contexts. The more specific top level goals supporting this single broad objective are to produce technology that will:

1. enable the operation of military systems under critical constraints such as time, information overload, etc.,
2. enable the management of forces/resources under constraints of information overload, geographic distribution, cost of operation, etc., and
3. facilitate the design, manufacture, and maintenance of defense systems within time, performance, quality, reliability, and cost constraints.

Even though capabilities for man-machine interaction will ultimately form an important component of systems in all of these areas, the second of those goals has been selected as the initial area to include emphasis on decision-making aids, including natural language processing.

Subgoals of these top level goals include:

1. To strengthen/develop areas of science and technology that enable the building of computer systems needed to attain the top level goals. The technologies identified are:
   - Artificial Intelligence,

  - Software development and machine architectures,
  - Micro-electronics, and related infrastructure.
2. To build demonstration systems in specific military areas that:
   - Provide focus for technology development,
   - Provide means for exercising technology in real environments,
   - Facilitate manpower training,
   - Facilitate development of industrial capability, and
   - Facilitate technology transfer to the military.

There are four very ambitious demonstration prototypes being developed within the SC Program. They are:

1. the Pilot's Associate, which will aid the pilot in route planning, aerial target prioritization, evasion of missile threats, and aircraft emergency safety procedures during flight;
2. the Autonomous Land Vehicle (ALV), which integrates in a major robotic test bed the technologies for dynamic image understanding, knowledge-based route planning with replanning during execution, hosted on new advanced parallel architectures;
3. two battle management projects: one for the Army, which is just getting started, called the AirLand Battle Management program (ALBM), which will use knowledge-based systems technology to assist in the generation and evaluation of tactical options and plans at the Corps level;
4. and the other more established program for the Navy, the Fleet Command Center Battle Management Program (FCCBMP) at Pearl Harbor. The FCCBMP is employing knowledge-based systems and natural language technology in an evolutionary test bed situated in an operational command center to demonstrate and evaluate intelligent decision aids *that* can assist in the evaluation of fleet readiness and explore alternatives during contingencies. It is within this context that the natural language contractors are currently demonstrating the potential of natural language technology.

Competitive awards were made to seven contractors in 1984. Four (BBN Laboratories, USC/Information Sciences Institute, the University of Pennsylvania, and the University of Massachusetts) are involved in research and development in natural language interfaces; three (New York University, Systems Development Corporation, and SRI International) are involved in research and development in text processing.

The work focuses on producing and demonstrating two "new generation systems": one for natural language interfaces and another for processing text in free form from military messages. BBN Laboratories serves as the integration contractor in natural language interfaces; New York University serves as the integration contractor in message processing. The remaining contractors are working on various component technologies, directly or indirectly contributing to the two new generation systems.

BBN LABORATORIES

Research and Development in Natural Language Processing in the Strategic Computing Program

BBN Laboratories, Inc.
Cambridge, MA 02238

Staff: Ralph Weischedel (Principal Investigator), Remko Scha, Edward Walker, Damaris Ayuso, Andrew Haas, Erhard Hinrichs, Robert Ingria, Lance Ramshaw, Varda Shaked, David Stallard

1 BACKGROUND

BBN's responsibility is to conduct research and development in natural language interface technology. This responsibility has three aspects:

- to demonstrate state-of-the-art technology in a Strategic Computing application, collecting data regarding the effectiveness of the demonstrated heuristics,
- to conduct research in natural language interface technology, as itemized in the description of JANUS later in this note, and
- to integrate technology from other natural language interface contractors, including USC/Information Sciences Institute, the University of Pennsylvania, and the University of Massachusetts.

Of the three initial applications described in the overview, the Fleet Command Center Battle Management Program (FCCBMP) has been the application providing the domain in which our work is being carried out. The FCCBMP encompasses the development of expert system capabilities at the Pacific Fleet Command Center in Hawaii, and the development of an integrated natural language interface to these new capabilities as well as to the existing data bases and graphic display facilities. BBN is developing a series of increasingly sophisticated natural language understanding systems which will serve as an integrated interface to several facilities at the Pacific Fleet Command Center: the Integrated Data Base (IDB), which contains information about ships, their readiness states, their capabilities, etc.; the Operations Support Group Prototype (OSGP), a graphics system, which can display locations and itineraries of ships on maps; and the Force Requirements Expert System (FRESH), which is being built by Texas Instruments.

The target users for this application are naval officers involved in decision making at the Pacific Fleet Command Center; these are executives whose effort is better spent on navy problems and decision making than on the details of which software system offers a given information capability, how a problem should be divided to make use of the various systems, or how to synthesize the results from several sources into the desired answer. Currently they do not access the data base or OSGP application programs themselves; instead, on a round-the-clock basis, two operators act as intermediaries between the Navy staff and the computers. The utility of a natural language interface in such an environment is clear.

The starting point for development of the natural language interface system at the Pacific Fleet Command Center was the IRUS system, which has been under development at BBN for a number of years. A new version of this system, IRUS-86, has been installed in the FCCBMP testbed area at the Pacific Fleet Command Center for demonstration. Further basic research on the problems of natural language interfacing is continuing, and the results of this and future research will be incorporated into a next generation natural language interface system called JANUS to be delivered to the Pacific Fleet Command Center at a later date. JANUS will share most of its domain-dependent data with IRUS-86, and it will share other modules as well; IRUS-86 will therefore be able to evolve gradually into the final version of JANUS.

2 IRUS-86: THE INITIAL TESTBED SYSTEM

The architecture of IRUS (Bates and Bobrow 1983) is a cascade consisting of a sequence of translation modules:

- An ATN parser, which produces a syntactic tree.
- A semantic interpreter, which produces a formula of the meaning representation language MRL.
- A postprocessor for resolving anaphora and ellipsis.
- A translation module, which produces a formula of the relational database language ERL (Extended Relational Language).
- A translation module, which produces a sequence of commands for the underlying database access system.

Now installed at the Pacific Fleet Command Center, IRUS-86 is a version of IRUS that has been extended in several ways. Two of these extensions are especially worth mentioning:

- IRUS-86 uses the NIKL system (Moser 1983) to represent its domain model, i.e., the relationships between the predicates and relations of the meaning representation language MRL. The NIKL domain model supports the system's treatment of semantic anomaly, anaphora, and nominal compounds.
- IRUS-86 contains a new module *that* exploits this NIKL domain model to simplify MRL expressions; this makes it possible to translate complex MRL-expressions into ERL constraints, thus allowing for significant divergences between the input English and the structure of the underlying data base (Stallard 1986).

In addition to accessing the NIKL domain model, the parser, semantic interpreter, and MRL-to-ERL translator access other knowledge sources *that* contain domain-dependent information:

- the lexicon,
- the semantic interpretation rules for individual concepts,
- the MRL-to-ERL mapping rules for individual MRL constants, which introduce the details of underlying system structure, such as file and field names.

To port IRUS to the Navy domain, the relevant domain-dependent data had to be supplied to the system. This task is being accomplished by personnel at the Naval Ocean Systems Center (NOSC). In August 1985, BBN provided NOSC with an initial prototype system containing small example sets of lexical entries, semantic interpretation rules, and MRL-to-ERL rules; using acquisition tools provided by BBN, NOSC personnel have been entering the rest of the data.

IRUS-86 was delivered to the FRESH developers at Texas Instruments in January 1986, was installed in a testbed area of the Pacific Fleet Command Center in April 1986, and will be demonstrated in June 1986. Currently, the lexicon and the domain-dependent rules of the system only cover a relatively small part of the OSGP capabilities and the files and attributes of the Integrated Data Base. Once enough data have been entered so that the system covers a sufficiently large part of the data base, it will be tried out in actual use by Navy personnel. This will enable us to gather data about the way the system performs in a real environment, and to fine-tune the system in various respects. For instance, IRUS-86 makes use of shallow heuristic methods to address some aspects of natural language understanding such as anaphora and ellipsis for which general solutions are still research issues. The FCCBMP application provides a test bed in which such heuristic methods can be evaluated, and enhancements to them developed and tested, as part of the evolutionary technological growth intended to continue throughout the Natural Language Technology effort of the Strategic Computing Program.

3 FUNCTIONAL GOALS FOR JANUS

The IRUS-86 system excels by its clean, modular structure, its broad syntactic/semantic coverage, its sophisticated domain model, and its systematic treatment of discrepancies between the English lexicon and the database structure. We thus expect that it will demonstrate considerable utility as an interface component in the FCCBMP application. Nevertheless, IRUS-86 shares with other current systems several limitations that should be overcome if natural language interfaces are to become truly "natural". In developing JANUS, the successor of IRUS-86, we shall attempt to overcome some of those limitations. The areas of increased functionality we are considering are: semantics and knowledge representation, ill-formedness, discourse, cooperativeness, multiple underlying systems, and knowledge acquisition.

3.1 SEMANTICS AND KNOWLEDGE REPRESENTATION

IRUS-86, like most other current systems, represents sentence meanings as formulas of a logical language *that* is a slight extension of first-order logic. As a consequence, many important phenomena in English have no equivalent in the meaning representation language, and cannot be dealt with correctly, e.g., modalities, propositional attitudes, generics, collective quantification, and context-dependence. Thus, one foregoes one of the most important potential assets of a natural language interface:

the capacity of expressing complex semantic structures in a succinct and comfortable way.

In JANUS, therefore, we will adopt a new meaning representation language *that* combines features from PHLIQA1's enriched lambda-calculus (Scha 1976) with ideas underlying Montague's Intensional Logic (Montague 1970), and possibly a distributed quote-operator (Haas 1986). It will have sufficient expressive power to incorporate a version of Carlson's treatment of generics (Carlson 1979), a version of Scha's treatment of quantification (Scha 1981), Montague's treatment of modality, and various possible approaches to propositional attitudes and context-dependence.

In adopting a higher order logic as proposed, one confronts problems of formula simplification and the need to apply meaning postulates to reduce the semantic representation of an input sentence to an expression appropriate to the underlying system, e.g., a relational algebra expression in the case that the underlying system is a data base. To do this, we will investigate the limited inference mechanisms of KL-TWO (Moser 1983, Vilain 1985), following up on our previous work (Stallard 1986). The advantage of these inference mechanisms is their tractability; discovering their power and limitations in this complex problem domain should be an interesting result.

3.2 DISCOURSE

The meaning of a sentence depends in many ways on the context set up by the preceding discourse. IRUS and other systems, however, currently ignore most of these dependencies, and employ a rather shallow model of discourse structure. To allow the user to exploit the full expressive potential of a natural language interaction, the system must track topics, reference times, possible antecedents for anaphora, etc.; it must be able to recognize the constituent units of a discourse and the subordination or coordination relations obtaining between them. A substantial amount of work has been done already on several of these issues, much of it by BBN researchers (Sidner 1985, Hinrichs 1981, Polanyi 1984, Grosz and Sidner 1986). Research in this area continues under a separate DARPA-funded contract. We expect to be able to integrate some of the results of that research in the JANUS system.

3.3 ILL-FORMEDNESS

A natural interface system should be forgiving of a user's deviations from its expectations, be they misspellings, typographical errors, unknown words, poor syntax, incorrect presuppositions, fragmentary forms, or violated selection restrictions. Empirical studies show that as much as 25% of the input to database query systems is ill-formed.

IRUS currently handles some classes of ill-formedness by using a combination of shallow heuristics and user interaction. It can correct for typographical misspellings, for omitted determiners or prepositions, and for some ungrammaticalities, like determiner-noun and subject-verb disagreement. The JANUS system will employ a more general approach to ill-formedness that will handle a larger class of ungrammatical constructions, a larger class of word selection problems, and that will also explore correcting several types of semantic ill-formedness.

These capabilities have major implications for the control of the understanding process, since considering such possibilities can exponentially expand the search space. Maintaining control will require care in integrating the ill-formedness capability into the rest of the system, and also in making maximal use of the guidance that can be derived from a model of the discourse and user's goals to constrain the search.

3.4 COOPERATIVENESS

A truly helpful system should react not to the literal meaning of a sentence but to its perceived intent. If in the context of a given application it is possible to characterize the goals that a user may be expected to be pursuing through his interaction with the system, the system should try to infer from the user-input what the underlying goal could be. A system can do this by accessing a goal-subgoal hierarchy *that* links the speech acts expressed by individual utterances to the global goals that the user may have. This strategy has been applied successfully to rather small domains (Allen 1983, Sidner 1985). We wish to investigate whether it carries over to the FCCBMP applications.

3.5 MODELLING THE CAPABILITIES OF MULTIPLE SYSTEMS

The way in which IRUS-86 decides whether an input sentence translates into an IDB query or an OSGP command may be refined. There is a need for work on what kind of knowledge would be necessary to interface smoothly and intelligently to multiple underlying systems. A reasoning component is needed that can determine which underlying system or systems can best fulfill a user's request. Such a reasoning component would have to combine a model of the capabilities of the underlying systems with a model of the user goals and current intentions in the discourse context in order to choose the correct system(s). Such a model would also be useful for providing supporting information to the user.

3.6 KNOWLEDGE ACQUISITION

Further research is also called for to expand the power of the knowledge acquisition tools used in adding to the lexicon, the set of case frame rules, the model of domain predicates, and the set of transformation rules between the Meaning Representation Language and the languages of the underlying systems. The acquisition tools available in IRUS, unlike those in some other systems, are not tied to the specific fields and relations in the underlying data base. The acquisition tools should work on the higher level of the domain model, since that provides a more general and transportable result. The knowledge acquisition facilities for JANUS will also need to be redesigned to support and to make maximal use of the power of the new meaning representation language based on intensional logic.

4 NEW UNDERLYING TECHNOLOGIES

4.1 COPING WITH AMBIGUITY

The new functionalities we described in the previous section, and the techniques we intend to use to achieve them, raise an issue *that* has important consequences for the design of JANUS: we will be faced with an explosion in the number of interpretations that the system will have to process; every sentence will be manifold ambiguous. One source of this phenomenon is the improvement of the semantic coverage and the broadening of the discourse context. Distinctions and ambiguities which so far were ignored will be dealt with: for instance, different interpretation and scopes of quantifiers will be considered, and different antecedents for pronouns. Even more serious is the processing of ill-formed sentences, which may require that some constraint be relaxed, while the only way to find out which one is to try all partial interpretations to see which one can be extended to a complete interpretation after relaxing one or more constraints.

To cut down on the processing of spurious interpretations, it is very important that interpretations of sentences and their constituents be tested for plausibility at an early stage. Different techniques must probably be used in conjunction:

- Simplification transformations may show that an interpretation is absurd, by reducing it to *true* or *false* or the empty set.
- The discourse context and the model of the user's goals impose constraints on expected sentences.

4.2 PARALLEL PARSING

Since some of the techniques we intend to use to fight the ambiguity explosion are themselves rather computation-intensive, it is clearly unavoidable that the improved system functionality we aim for will lead to a considerable increase in the amount of processing required. To avoid a serious decrease of the new system's response times, we will therefore move it to a suitable parallel machine such as BBN 's Butterfly or Monarch, running a parallel COMMON LISP. This in itself has rather serious consequences for the software design. It means that from the outset we will keep parallelizability of the software in mind.

We have begun to address this issue in the area of syntax. A new declarative grammar is being written, which will ultimately have a coverage of English larger than the current RUS grammar. The grammar is written in a side-effect-free formalism (a context-free grammar with variables) so that we may explore different parsing algorithms that are easily parallelizable. The first such algorithm was implemented in May 1986 on BBN's Butterfly.

5 CONTRIBUTIONS FROM OTHER SITES

5.1 ISI/UMASS: GENERATION

We should not expect that JANUS will always be able to assess correctly which interpretation of a sentence is the intended one. In light of such situations, it is very important that the system can give a paraphrase of the input to the user, which shows the system's interpretation. This may be done either explicitly or as part of the answer. To be able to develop such capabilities, work on Natural Language Generation is needed. At USC/ISI a project directed by William Mann and Norman Sondheimer is underway to develop the generation system PENMAN, using the NIGEL systemic grammar. PENMAN will be integrated to become the generation component of JANUS. PENMAN itself consists of several subcomponents. Some of these, specifically the "text planning" component, will be developed through joint work between USC/ISI and David McDonald at the University of Massachusetts, based on the latter's experience with the MUMBLE system.

5.2 UPENN: COOPERATION AND CLARIFICATION

Under the direction of Aravind Joshi and Bonnie Webber of the University of Pennsylvania, several focussed studies have been carried out at UPenn to investigate various aspects of cooperative system behaviour and clarification interactions. (For more detail, see their report below.) As part of the Strategic Computing Natural Language effort, UPenn will eventually develop this into a module *that* can be integrated into JANUS to further enhance its capabilities.

REFERENCES

Allen, J.F. 1983 Recognizing Intentions from Natural Language Utterances. In Brady, M. and Berwick, R.C., Eds, *Computational Models of Discourse.* Massachusetts Institute Technology Press, Cambridge, Massachusetts: 107-166.

Bates, M. and Bobrow, R.J. 1983 A Transportable Natural Language Interface for Information Retrieval. In *Proceedings of the 6th Annual International ACM SIGIR Conference.* ACM Special Interest Group on Information Retrieval and American Society for Information Science, Washington, D. C. (June).

Carlson, G. 1979 *Reference to Kinds in English.* Garland Press, New York.

Grosz, B.J. and Sidner, C.L. 1986 The Structures of Discourse Structure. In Polanyi, L., Ed., *Discourse Structure.* Ablex Publishers, Norwoods, New Jersey.

Haas, A.R. 1986 A Syntactic Theory of Belief and Action. *Artificial Intelligence.*

Hinrichs, E. 1981 Temporale Anaphora im Englischen. Unpublished ms., University of Tuebingen.

Montague, R. 1970 Pragmatics and Intensional Logic. *Synthese* 22: 68-94.

Moser, M.G. 1983 An Overview of NIKL, the New Implementation of KL-ONE. In Sidner, C. L., et al., Eds., *Research in Knowledge Representation for Natural Language Understanding – Annual Report, 1 September 1982 - 31 August 1983.* BBN Laboratories Report No. 5421: 7-26.

Polanyi, L. and Scha, R. 1984 A Syntactic Approach to Discourse Semantics. In *Proceedings of International Conference on Computational Linguistics.* Stanford University, Stanford, California.

Scha, R.J.H. 1976 Semantic Types in PHLIQA1. In *Proceedings of the 6th International Conference on Computational Linguistics.*

Scha, R.J.H. 1981 Distributive, Collective and Cumulative Quantification. *Formal Methods in the Study of Language, Part 2.* Mathematisch Centrum, Amsterdam: 483-512. Reprinted in Groenedijk, J.A.G.; Janssen, T.M.V.; and Stokhof, M.B.J., Eds., *Truth, Interpretation and Information.* GRASS 3. Dordrecht, Foris.

Sidner, C.L. 1985 Plan Parsing for Intended Response Recognition in Discourse. *Computational Intelligence* 1(1): 1-10.

Stallard, D.G. 1986 A Terminological Simplification Transformation for Natural Language Question-Answering Systems. In *Proceedings of the 24th Annual Meeting of the Association for Computational Linguistics.* Association for Computational Linguistics (July).

Vilain, M. 1985 The Restricted Language Architecture of a Hybrid Representation System. In *Proceedings of IJCAI85, International Joint Conference on Artificial Intelligence, Inc.* Morgan Kaufmann Publishers, Inc., Los Angeles, California: 547-551.

UNIVERSITY OF PENNSYLVANIA

Research in Natural Language Processing

University of Pennsylvania
Department of Computer and Information Science

Faculty: Aravind Joshi, Tim Finin, Dale Miller, Lokendra Shastri, and Bonnie Webber

Students: Brant Cheikes, John Dowding, Amy Felty, Ellen Hays, Robert Kass, Ron Katriel, Sitaram Lanka, Megan Moser, Gopalan Nadathur, MaryAngela Papalaskaris, Martha Pollack, Robert Rubinoff, Yves Schabes, Ethel Schuster, Sunil Shende, Jill Smudski, Vijayshankar, David Weir, Blair Whitaker

Facilities: LINC (Langauge, Information, and Computation) laboratory, which consists of a dedicated VAX 11/785, 10 Symbolics Lisp machines, 7 HP 68020-based AI workstations, a SUN workstation, several Macintoshes, and a laser printer. These machines are networked together and to other research facilities in the department.

This is a brief report summarizing our work to date and our intermediate and long term goals. See **Abstracts of Current Literature** in this issue for a summary of some of our publications on this work.

1 MAJOR THRUST

Natural language interfaces providing support for many different communicative functions:

- Providing definitions of concepts;
- Recognizing and correcting user misconceptions;
- Providing explanations;
- Offering to provide information later, when known;
- Verifying and demonstrating understanding;
- Exploiting and enriching the context of natural language discourse between user and system.

1.1 WORK TO DATE

- Integration of RUS-TEXT-MUMBLE (RTM) – This effort involves integrating three natural language system components (BBN's RUS parser-interpreter, McKeown's TEXT system (developed at Penn), and McDonald's MUMBLE system (received from U. Mass in January 1985). This integration of three independently developed systems has required substantial effort. This version of RTM (to be completed in May 1986)
  - accepts a limited number of English language requests for definitions of, descriptions of, or comparisons between terms in the ONR database used by Kathy McKeown in her development of TEXT;
  - formulates appropriate responses using TEXT and outputs those responses in English using MUMBLE; and
  - runs on a SYMBOLICS Lisp machine.

  This work has been done by Moser, Whitaker and Rubinoff.
- Initial work on incorporating a sense of relevance in monitor offers. Mays's dissertation work on monitor offers was limited to issues of competency. This work is being done by Cheikes and Schabes.
- Completion of McCoy's dissertation work on correcting certain types of object-related misconceptions and implementation of a system called ROMPER, which generates such corrections. (MUMBLE is used as the tactical generation component of this system as well.)
- Completion of Hirschberg's dissertation work on scalar implicatures and their use in constructing non-misleading responses.
- Completion of Pollack's dissertation work on plan inference in which user and system beliefs about actions and plans are decoupled.
- Continuation of work on integrating scalar-implicatures-based reasoning within a general framework of circumscription-based non-monotonic reasoning.
- Development of methods for converting proofs in a system akin to first-order resolution into natural deduction (ND) proofs, which are then reorganized into cohesive paragraphs using Chester's 1976 algorithm.
- Development of methods of converting modal resolution proofs into modal ND proofs and higher-order resolution proofs into higher-order ND proofs.
- Initial development of domain-independent tools for expressing and reasoning about user models – in particular, for defining hierarchies of stereotypical users, representing individual users, and drawing inferences about them using a default logic.
- Continuation of basic research on local coherence of discourse using the notions of centering and syntax, semantics, and parsing of tree adjoining grammars.

2 FUTURE PLANS

Having gained the experience of integrating three natural language systems and carrying out some of the basic research as described in the previous section, we have now developed the plan described below, which summarizes the near- and long-term goals.

2.1 NEAR-TERM GOALS

We have three tangible goals for the next year:

- Completing the RTM demonstration system (using the existing domain and knowledge representation) and producing a videotape *that* explains and demonstrates it.
- Developing TEXT into a more modular tool for defining and comparing terms, on the order of RUS and MUMBLE. This will eliminate its tie to a particular knowledge representation and increase its portability.
- Acquiring familiarity with the PENMAN approach to NL generation through acting as a beta-test site for NIGEL.

2.2 LONG-TERM GOALS

2.2.1 SUPPORT FOR NL DEFINITIONS – ENRICHED KNOWLEDGE REPRESENTATION

In our original proposal, we stated our intention of employing a richer knowledge representation as the basis for our work on text generation, especially for constructing definitions. Our original idea was to make use of BBN's NIKL system. In the past year though, we have become aware of some of NIKL's limitations, which essentially make it non-optimal, even as a next step, for our text generation work. On the other hand, we have identified several features with which a NIKL-like language could be enriched to make it more suitable for our work:

- associating non-definitional information with concepts in a way that maintains the underlying structure of that information, without interfering with NIKL's automatic classification mechanism;
- associating "evidential" information with concepts, especially frequency information – how often the concept is known to display particular features;
- allowing for what appears to be conflicting information coming down through inheritance – e.g., information that is contrary to expectations grounded in an alternative perspective on a concept;
- allowing mutual definition of concepts – each being defined with reference to the others in a set;
- incorporating notions of time and change – allowing the defining properties and evidential properties of concepts to include how they change over time;
- allowing assertions about usual relations between properties of subtypes.

Work on an enriched knowledge representation that includes all these features in a well-motivated way will take several years. However one that includes at least the first three of them can probably be developed over the next two years, with work on employing it in text generation beginning six months to a year after the start of that work.

2.2.2 SUPPORT OF NL DEFINITIONS – USE OF DISCOURSE AND USER MODELS

The TEXT system, as it is currently structured, will produce the same definition for a concept (or comparison between two concepts) whenever it is asked. It does not take into account what the user may have already found out about the concept, or what it is implicitly being contrasted with (e.g., some other concept the user has recently asked about), or what the user's goal is in making the request. Hence, other directions in which we would like to take this definitional/clarificational capability is to increase its sensitivity to

- the discourse history, to avoid repetition and possibly to take advantage of the additional clarity brought by contrasting a new term to one explained before;
- the user's level of expertise, to avoid either stating the obvious or going more deeply into a concept than the user can understand; and
- the user's goals, to focus on those aspects of the concept being defined (or concepts being compared) that are significant to the current task. (The latter is related to the notion of "perspective" used in Kathy McCoy's recent thesis here.)

For both these aspects of user modelling (in contrast with the first point, which can be developed using the current discourse alone), we will draw on the other work being done here on domain-independent user-modelling mechanisms. This proposed work must be done in a domain in which tasks can be characterized and recognized. Thus we plan to do this initially in an investment advising domain we have started to develop. Work on incorporating and using discourse history will involve about a one-year effort, once the knowledge base is built. Work on incorporating and using a model of a user's expertise and goals will take more time, on the order of two to three years.

2.2.3 EXPLANATIONS

Again in our original proposal, we proposed work on constructing natural language explanations – more specifically, on ways to loosen the current tight coupling between the form of the system's proof of some statement to the form of its explanation of why the statement is true. This coupling has kept systems that should be able to explain their reasoning from employing stronger proof methods that do not have a natural, understandable form of presentation to their human users.

Our immediate goals involve:

- developing a demonstration system *that* responds to NL queries posed to RUS by doing an efficient first-order resolution-based proof, transforming that proof into an ND proof, organizing that proof according to an improved version of the Chester algorithm, and then producing an English version of the text using MUMBLE or NIGEL.
- abstracting from the three separate sets of proof conversion methods (noted under WORK TO DATE) into general methods of transforming any resolution-

style proof in any logic into its corresponding ND proof.

- determining whether existing methods of organizing first-order ND proofs into paragraphs are applicable to ND proofs in these stronger logics or whether more must be done to produce high-quality, cohesive, understandable text.

Our long–term goals remain as stated in our original proposal – the production of explanations sensitive to users' beliefs, expertise, desired level of detail and expectations. In this long-term research, we see taking expertise and desired level of detail into account in determining how much of the ND proof is made explicit. Of more interest is how users' beliefs and expectations should affect the explanations. Work on scientific explanation has shown that central to the explanation of what is the case is a set of alternative situations *that* are not the case. One explains *what is* in contrast to *what is not*. However, this requires additional work, to prove of each of the alternatives (which may be given explicitly by the user – *"why this and not that?"* – or inferred from the system's model of the user's expectations) that it is not true. Our planned approach involves guiding the (failing) proof of each alternative against the successful proof. The point is that although there may be many failing proofs of each alternative, the most relevant of these in the current situation is the one that is analogous – up to the point of failure – to the original successful proof. This technique should not only provide relevant information but should also be efficient in reducing the search space. We expect this work to take on the order of two to three years, provided we have enough resources to pursue it in parallel with our more near-term goals.

2.2.4 NATURAL LANGUAGE PARSING AND GENERATION

While using the RUS system, we will continue our work on tree adjoining grammar (TAG) both from the parsing and generation points of view. TAGs lead to some attractive approaches to parallelizing parsing and also seem to provide natural planning units for generation. This work will be integrated with our future work on parsing and generation. Our first language generator (used by TEXT) was one based on Kay's Functional Unification Grammar. While theoretically elegant, it was unacceptably slow (in its straightforward implementation), leading us last year to import the MUMBLE generator from McDonald at University of Massachusetts and adapt it to work with TEXT. Using MUMBLE has produced a 60-fold speed-up in generation time. However, adapting MUMBLE to work with TEXT and, independently, with two other systems has made us aware of MUMBLE's limitations, primarily its lack of knowledge of words or grammar. Essentially, MUMBLE's knowledge is limited to how to realize particular message units (i.e., to choose an acceptable one from an a priori specified set of choices), given constraints already imposed by message units that have already been realized. The large amount of work that must be invested in building a MUMBLE lexicon and the lack of inter-application portability of anything but the control structure comes from this fact – that one has to completely specify each set of choices beforehand for each message unit and the sets are completely application specific. We propose to work on the development of a new architecture, including our work on tags, that avoids these limitations by having more knowledge of syntax and words and hence is more portable between applications. The time frame for this project is approximately three years.

2.2.5 ANAPHORA RESOLUTION

The RUS parser/interpreter we received from BBN uses a limited method of resolving definite pronouns and noun phrases that is only a bit more advanced than the one originally developed for BBN's LUNAR system back in 1971. Since then, there have been major theoretical advances in our understanding of discourse anaphora (in the works of Grosz (at SRI), Joshi, Sidner (at BBN), Webber, and Weinstein), but these theoretical advances have not yet found their way into natural language understanding systems. We feel strongly qualified to undertake this work, having two of the major participants (Joshi and Webber) here at Penn already, and want to do so. For us, it is both of research interest and of practical importance, since it can mean a major improvement in system's understanding abilities. We will also integrate our work on tags with this effort as it relates to parsing and generation. This work will also complement additional work being done here on a theoretical and computational account of anaphoric reference to actions and events. We see this work as taking about two to two and a half years.

2.2.6 USER MODELING

The need for systems to model the knowledge and beliefs of their users has already been pointed out. We plan to address a number of issues *that* underlie the successful development and incorporation of explicit user models. Our current domain-independent user-modelling system, GUMS, provides mechanisms for defining hierarchies of stereotypical users, representing individual users, and drawing inferences about them using a rich default logic. We will continue to develop this system as a tool *that* will support the user modeling needs of various applications. We also plan to study the problem of how new knowledge of individual users can be derived from their regular interaction; that is, how relevant information about users can be inferred from their queries and responses. In other situations it may become necessary for the system to explicitly pose a few crucial questions to the user to determine what he or she does and does not know.

2.2.7 SYSTEM INTEGRATION

Finally, we plan to begin work on system integration. In recent years, we have identified many types of behavior that interfaces to database systems and expert systems should demonstrate. Beginning with Kaplan's work on recognizing and responding to existential presupposition failures in his COOP system, we have developed and produced several modules, each demonstrating another type of desired behavior. These include the ability to recognize and respond to type failures, the ability to respond to object-related misconceptions, the ability to calculate and offer competent database monitors, the ability to use scalar implicatures to convey additional information, and the ability to respond to a class of "inappropriate" queries, and various paraphrase abilities.

Following the publication of Kaplan's thesis, the features of his COOP system were soon incorporated into several database interfaces (both natural slanguage and formal query language). This gave the resulting systems the ability to produce two types of responses: either a direct answer, if there was one, or a statement concerning the absence of individuals satisfying some description in the given query. Now we plan to tackle the more significant problem related to this:

Given a system that is able to call upon a variety of response strategies, how does it decide what to do in a given circumstance? This is the issue we plan to explore by investigating the integration of multiple communicative behaviors. Given a system with several different types of useful behaviors, which can be combined in various ways, can one efficiently and effectively coordinate a response that is better (i.e., more useful, more helpful and more understandable) than simply a (direct) answer? While we speculate that it will be the case that identifying what one might consider the best response might take complex reasoning about the user's goals, level of expertise, and need-to-know with respect to what the answer (if any) actually is, we also plan to look at how, with more limited resources, we can still improve system behavior.

This aspect of our future plans is the most long term, involving both the actual component integration itself (in which, in many cases, it is only the basic ideas that can be carried over, where the component must be re-programmed entirely to fit into the integrated system) and the development of that part of the total system that reasons about what kind of response(s) to give. The time frame here is approximately four years.

2.2.8 ARCHITECTURE

We plan to investigate parallel and connectionist architectures and algorithms for realizing our systems, especially those for knowledge representation, reasoning, explanations, and integrated parsing and generation.

UNIVERSITY OF MASSACHUSETTS

The COUNSELOR Project at the University of Massachusetts

David D. McDonald and James D. Pustejovsky
Department of Computer and Information Science
University of Massachusetts,
Amherst, Massachusetts 01003

Participants in the COUNSELOR Project,
Fall 1984 through Summer 1986:
Principal Investigators: Edwina L. Rissland, David D. McDonald, Wendy G. Lehnert
Research Associates: Beverly Woolf, James D. Pustejovsky
Graduate Students: Marie M. Vaughan, Brian Stucky, Penelope Sibun, Seth Rosenberg, Kelly Murray, Kevin Gallagher, JoAnn M. Brooks, John Brolio, Sabine Bergler, Kevin D. Ashley, Scott D. Anderson

1 INTRODUCTION

The COUNSELOR Project began in the fall of 1984 with the goal of exploring basic problems in discourse structure and text processing within an integrated interface to a strong expert system. The program we have developed, COUNSELOR, integrates separately developed components for natural language generation (MUMBLE: see McDonald and Pustejovsky 1985a,b,c), parsing (PLUM: Lehnert and Rosenberg 1985), and case-based legal reasoning (HYPO: Ashley 1986, Ashley and Rissland 1985). It adds a newly developed component, CICERO (Pustejovsky 1986), positioned between the two text processors and the expert system; CICERO is responsible for managing textual inferences ("reading between the lines") by using common sense models of legal events. COUNSELOR can provide advise to an attorney about how to argue cases involving violations of trade secret law in the computer field. The attorney presents the facts of their case to the system, which may ask questions to elicit other facts that it knows to be relevant. The system then suggests lines of argument that the attorney might use, drawing on its library of litigated cases to find ones with analogous dimensions.

At its present state of development, COUNSELOR can handle simple variations on a single scenario, exemplified by the following dialog:

User:	I represent a client named HackInc, who wants to sue SwipeInc and Leroy Soleil for misappropriating trade secrets in connection with software developed by my client. HackInc markets the software, known as Autotell, a program to automate some of a bank teller's functions, to the banking industry.
COUNSELOR:	Did Soleil work for HackInc.?
User:	Yes, he was a key employee on the Autotell project.
COUNSELOR:	Did he later work for SwipeInc.?
User:	Yes.
COUNSELOR:	You can argue that there is an implied agreement arising out of Soleil's employment with HackInc that he not disclose any trade secret information to which he gained access by virtue of his employment.

2 MOTIVATIONS

Consequential results in natural language research will only come from working with a strong underlying program whose communicative needs will challenge the capabilities of state of the art of language interfaces. As a group, we are not interested in building yet another question answering system: our goal is to understand the structure of discourse. We believe that an effective place to begin is with task specific, mixed initiative dialog where the participants' goals cannot be satisfied by single utterances. Working with a legal reasoning system like Kevin Ashley and Edwina Rissland's HYPO provides particular challenges to natural language research:

1. Legal text is structurally complex. The need to avoid ambiguity leads to deeply embedded clauses and heavy noun phrases.
2. As both the user and the system have a thorough knowledge of the law, they communicate vastly more information in conversations about legal arguments than ever appears in their literal utterances.
3. HYPO's role as an advisory system creates a natural motivation to communicate through language.
4. Legal cases are large, complex objects that can be viewed from many alternative perspectives. The purpose for which a case is being described strongly influences which of its attributes are salient and how that information should be structured as a text.

3 COMPONENT PARTS

We began the project with three partially developed components, HYPO, MUMBLE, and PLUM, each designed with independent motivations. An initial tension was whether to convert aspects of these programs that did not seem apt in their new setting, or alternatively to interpose new components between them to smooth out the differences. We concluded that the motivations underlying each component were strong enough that we should not change them just because they were now working together. HYPO reasons with cases and hypotheticals. Actually litigated legal cases are encoded and indexed by "dimensions", which capture the utility of a case for making a particular kind of argument. When evaluating new cases, HYPO first analyzes them in terms of the dimensions they involve. Relevant cases are then retrieved to guide the reasoning. The system may ask pertinent questions about facts now found to be relevant. When the analysis is complete, HYPO describes the arguments available to the user, and responses and counter responses that may follow.

MUMBLE, the linguistic component for generation, is responsible for realizing conceptual specifications as grammatical text cohesive with the discourse that proceeds it. MUMBLE works within a description directed framework. Its input specification is a description of the message the underlying program wants to communicate. This description is executed incrementally, producing an intermediate linguistic representation which defines the text's grammatical relations and imposes constraints on further realization. This surface structure description is concurrently executed, producing the actual text.

PLUM is a conceptual analyzer *that* has been given a well defined schematic structure so it can be easily extended. It parses by doing prediction and completion over semantic concepts implied by the words rather than over syntactic categories. As in other conceptual analyzers, no explicit surface structure is recovered. PLUM's output is the set of completed frames.

CICERO is a new component, a discourse and inference manager between the language components and the expert system. From the understanding side, CICERO must integrate the clause by clause output of the parser into the larger discourse context, recognizing, for example, when noun phrases refer to the same object. In interpreting these small, lexically derived frames, CICERO draws on its own representation of events which bridges the gap between the way such information is expressed in language and the way it is organized for expert legal reasoning. For generation, CICERO is responsible for planning the message that is given to the generator. In particular, it determines what information should be included and what may be omitted as inferable, and it selects pivotal lexical items with appropriate perspective and rhetorical force.

4 FUTURE DIRECTIONS

While the accomplishments of the individual components of COUNSELOR are interesting in their own right, the greatest effect of the project has been to provide a workbench for studying the problems of language in an integrated context. Perennial problems in anaphora, lexical semantics, aspect, etc. become more tractable in an integrated system where there is a discourse context and intensional motivation. There are also semantic generalizations between the level at which the text processors operate and the level of the expert system which are more easily captured when parsing and generation can be studied in unison. On a larger scale, an explicit discourse manager, a requisite for more complex dialogs, can only be developed once an integrated system exists.

REFERENCES

Ashley, Kevin D. 1986 Modelling Legal Argument: Reasoning with Cases and Hypotheticals – a Thesis Proposal. Technical Report 10, The COUNSELOR Project, Department of Computer and Information Science, University of Massachusetts at Amherst.

Ashley, Kevin D. and Rissland, Edwina L. 1985 Toward Modelling Legal Argument. *Proceedings of the 2nd International Congress LOGICA, INFORMATICA, DIRITTO.* Instituto Per La Documentazione Giuridica, Florence, Italy.

Brooks, JoAnn M. 1985 Themis: A Discourse Manager. Unpublished Master's thesis, Department of Computer and Information Science, University of Massachusetts at Amherst.

Gallagher, Kevin 1986 The Design and Implementation of CICERO. Unpublished Master's thesis, Department of Computer and Information Science, University of Massachusetts at Amherst.

Lehnert, Wendy G. and Rosenberg, Seth 1985 The PLUM User's Manual. Technical Report 1, The COUNSELOR Project, Department of Computer and Information Science, University of Massachusetts at Amherst.

McDonald, David D. 1986 Natural Language Generation: Complexities and Techniques. To appear in Nirenburg, Ed., *Theoretical and Methodological Issues in Machine Translation.* Cambridge University Press.

McDonald, David D. and Pustejovsky, James 1985a Description Directed Natural Langauge Generation. *Proceedings of IJCAI-85:* 799-805.

McDonald, David D. and Pustejovsky, James 1985b TAGs as a Grammatical Formalism for Generation. *Proceedings of the 23rd Meeting of the Association for Computational Linguistics:* 94-103.

McDonald, David D. and Pustejovsky, James 1985c SAMSON: A Computational Theory of Prose Style for Natural Language Generation. *Proceedings of the 1985 Meeting of the European Association for Computational Linguistics.*

Pustejovsky, James 1986 An Integrated Theory Discourse Analysis. Technical Report 11, The COUNSELOR Project, Department of Computer and Information Science, University of Massachusetts at Amherst.

Rissland Edwina L.; Valcarce, Edward; and Ashley, Kevin 1984 Explaining and Arguing with Examples. *Proceedings of AAAI-84.*

Vaughan, Marie M, and McDonald, David D. 1986 A Model of Revision in Natural Language Generation. *Proceedings of the 24th Meeting of the Associaton for Computational Linguistics.*

NEW YORK UNIVERSITY AND SYSTEM DEVELOPMENT CORPORATION

PROTEUS and PUNDIT: Research in Text Understanding

Department of Computer Science, New York University
System Development Corporation – A Burroughs Company

Prepared by Ralph Grishman (NYU) and
Lynette Hirschman (SDC)

1 INTRODUCTION

We are engaged in the development of systems capable of analyzing short narrative messages dealing with a limited domain and extracting the information contained in the narrative. These systems are initially being applied to messages describing equipment failure. This work is a joint effort of New York University and the System Development Corporation for the Strategic Computing Program. Our aim is to create a system reliable enough for use in an operational environment. This is a formidable task, both because the texts are unedited (and so contain various errors) and because the complexity of any real domain precludes us from assembling a "complete" collection of the relationships and domain knowledge relevant to understanding texts in the domain.

A number of laboratory prototypes have been developed for the analysis of short narratives. None of the systems we know about, however, is reliable enough for use in an operational environment (the possible exceptions are expectation-driven systems, which simply ignore anything deviating from these built-in expectations). Typical success rates reported are that 75-80% of sentences are correctly analyzed, and that many erroneous analyses pass the system undetected; this is not acceptable for most applications. We see the central task of the work to be described below as the construction of a substantially more reliable system for narrative analysis.

Our basic approach to increasing reliability will be to bring to bear on the analysis task as many different types of constraints as possible. These include constraints related to syntax, semantics, domain knowledge, and discourse structure. In order to be able to capture the detailed knowledge about the domain that is needed for correct message analysis, we are initially limiting ourselves to messages about one particular piece of equipment (the "starting air compressor"); if we are successful in this narrow domain, we intend to apply the system to a broader domain.

The risk with having a rich set of constraints is that many of the sentences will violate one constraint or another. These violations may arise from problems in the messages or in the knowledge base. On the one hand, the messages frequently contain typographical or grammatical errors (in addition to the systematic use of fragments, which can be accounted for by our grammar). On the other hand, it is unlikely that we will be able to build a "complete" model of domain knowledge; gaps in the knowledge base will lead to constraint violations for some sentences. To cope with these violations, we intend to develop a "forgiving" or flexible analyzer which will find a best analysis (one violating the fewest constraints) if no "perfect" analysis is possible. One aspect of this is the use of syntactic and semantic information on an equal footing in assembling an analysis, so that neither a syntactic nor a semantic error would, by itself, block an analysis.

2 APPLICATION

This work is work is a component of the Fleet Command Center Battle Management Program (FCCBMP), which is part of the Strategic Computing Program. The FCCBMP has two natural language components: one for interactive natural language access, the other for message processing. The interactive component – which is to provide access to a data base and multiple expert systems – is being integrated by Bolt Beranek and Newman. The message processing component is being integrated as a joint effort of New York University and the System Development Corporation.

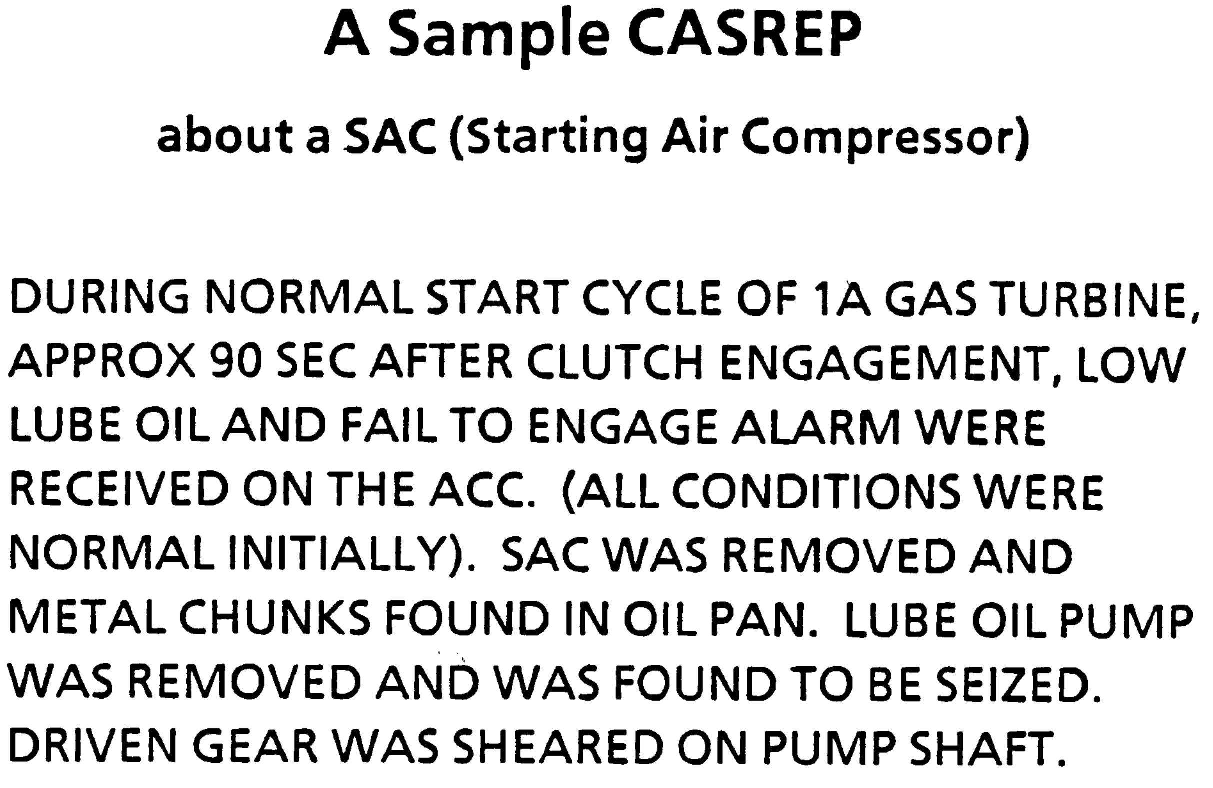
Much of the information received by the Fleet Command Center is in the form of messages. Some of these messages have a substantial natural language component. Consequently, natural language analysis is required if the information in these messages is to be recorded in a data base in a form usable by other programs. The specific class of messages we are studying are CASREPs, which are reports of equipment failures on board ships. These messages contain a brief narrative, typically 3 to 10 sentences in length, describing the symptoms, diagnosis, and possibly the attempts at repair of the failure. A typical narrative is shown in Figure 1. The problems we face in analyzing these messages are similar to those in analyzing short messages and reports in other technical domains, and we therefore expect that the solutions we develop will be widely applicable.

3 PROJECT ORGANIZATION

This work is a joint research effort of New York University and the System Development Corporation. NYU has principal responsibility for development of the domain knowledge base; SDC has principal responsibility for development of the flexible parser and for the domain-independent discourse components. The division of the other tasks is noted in the detailed component descriptions below. We will also be integrating work on the knowledge base being done by SRI, which is a component technology developer for the FCCBMP natural language work.

The work by NYU is being done in LISP (primarily in COMMON LISP), as is most of the Strategic Computing research. SDC is doing its development in PROLOG because PROLOG provides a powerful framework for writing grammars; it also provides the inference engine necessary for knowledge structuring and reasoning about the discourse structures in text processing. This division will permit us to make some valuable comparisons between the LISP and PROLOG development environments, and between the resulting systems.

The system being developed in LISP by NYU is called PROTEUS (PROtotype TExt Understanding System; Grishman et al., submitted for publication); the SDC system is called PUNDIT (Prolog UNDerstander of Integrated Text; Palmer et al. 1986). Notwithstanding the difference in implementation languages, we have tried to maintain a high level of compatibility between the two systems. We use essentially the same grammar and have agreed on common representations for the output of the syntactic analyzer (the regularized syntactic structure) and the output of the semantic analyzer. This commonality makes it possible to assign primary responsibility for the design of a component to one group, and then to take



A Sample CASREP

about a SAC (Starting Air Compressor)

DURING NORMAL START CYCLE OF 1A GAS TURBINE, APPROX 90 SEC AFTER CLUTCH ENGAGEMENT, LOW LUBE OIL AND FAIL TO ENGAGE ALARM WERE RECEIVED ON THE ACC. (ALL CONDITIONS WERE NORMAL INITIALLY). SAC WAS REMOVED AND METAL CHUNKS FOUND IN OIL PAN. LUBE OIL PUMP WAS REMOVED AND WAS FOUND TO BE SEIZED. DRIVEN GEAR WAS SHEARED ON PUMP SHAFT.

Figure 1

the design developed for one system and port it to the other in a straightforward way.

We are currently developing baseline systems *that* incorporate substantial domain knowledge but use a traditional sequential processing organization. When these systems are complete, we will begin experimenting with flexible parsing algorithms. The systems currently being developed (Figure 2) process input in the following stages: lexical look-up, parsing, syntactic regularization, semantic analysis, integration with the domain knowledge representation, and discourse analysis. These components, and other tasks that are part of our research program, are described individually below.

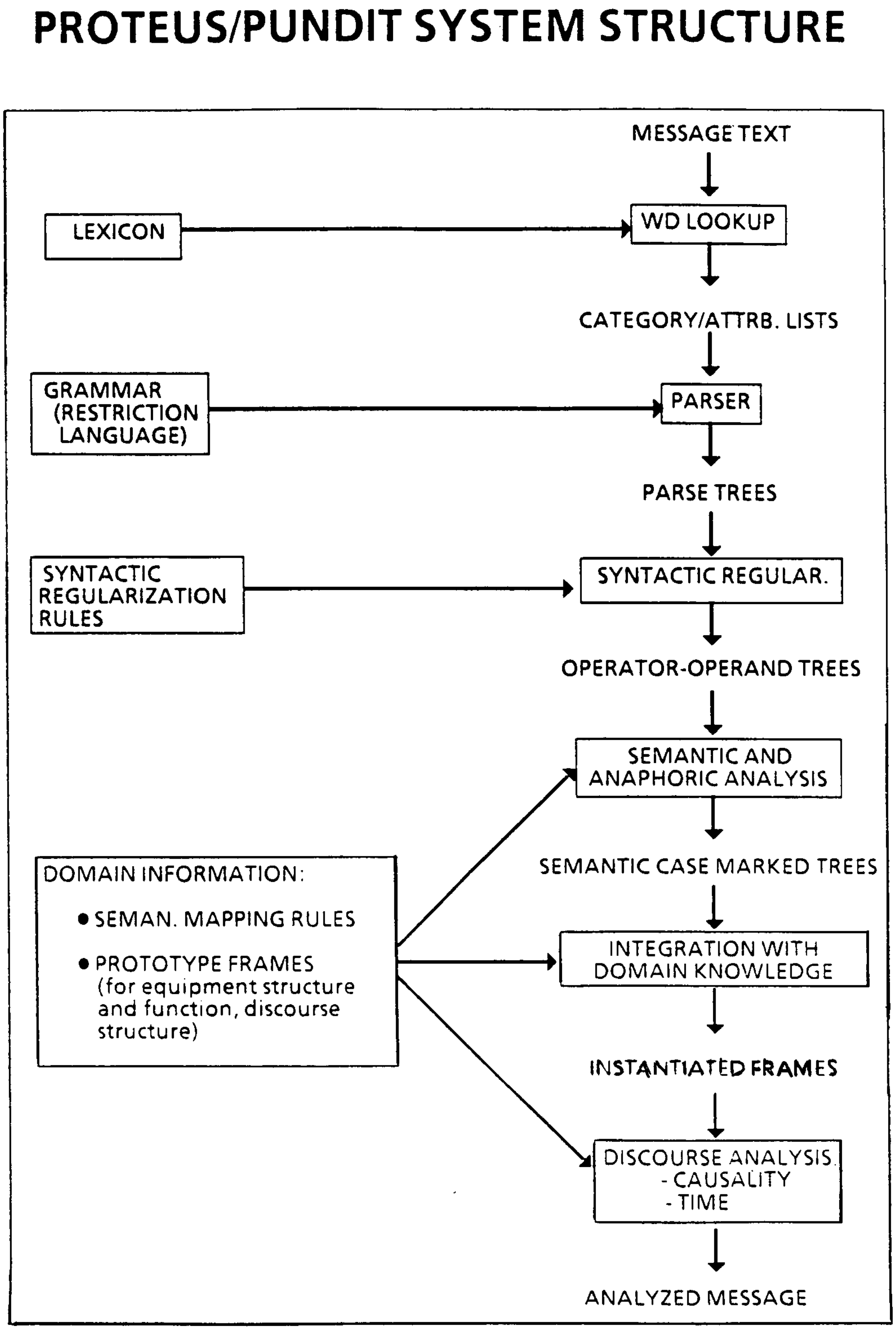


Figure 2

4. SYSTEM COMPONENTS

4.1 LEXICON (SDC + NYU)

The lexicon consists of a modified version of the lexicon of the NYU Linguistic String Project, with words classified as to part of speech and subcategorized for various grammatical properties (e.g., verbs and adjectives are subclassified for their complement types).

4.2. LEXICAL ACQUISITION (SDC)

The message vocabulary is large and will grow steadily as the system is modified to handle a wider range of equipment; several measures are planned to manage the growth of the lexicon. An interactive lexical entry program has been developed to facilitate adding words to the dictionary. Special constructions such as dates, times, and part numbers are processed using a small definite clause grammar defining special shapes. Future plans include addition of a component to use morphological analysis and selectional patterns to aid in classification of new lexical items.

4.3. SYNTAX ANALYSIS (NYU + SDC)

4.3.1. GRAMMAR

The syntactic component uses a grammar of BNF definitions with associated restrictions that enforce context-sensitive constraints on the parse. This grammar is generally modelled after that developed by the NYU Linguistic String Project (Sager 1981). The grammar has been expanded to cover the fragmentary constructions and complex noun phrases characteristic of the Navy message domain. A wide range of conjunction types is parsed by a set of conjunction rules which are automatically generated by metarules (Hirschman, in press). To serve as an interface between the syntactic and semantic components, an additional set of rules produces a normalized intermediate representation of the syntax.

4.3.2. TOP-DOWN PARSERS

Two top-down parsers have been implemented using the common grammar just described. In each case, the analyzer applies the BNF definitions and their associated constraints to produce explicit surface structure parses of the input; the analyzer also invokes the regularization rules *that* produce the normalized intermediate representation.

In the NYU (LISP-based) system the basic algorithm is a chart parser, which provides goal-directed analysis along with the recording (for possible re-use) of all intermediate goals tried. The context sensitive constraints are expressed in a version of Restriction Language (Sager 1975) compiled into LISP. The SDC (PROLOG-based) system uses a top-down left-to-right PROLOG implementation of a version of the restriction grammar (Hirschman and Puder 1986).

4.4. FLEXIBLE ANALYZER (SDC)

A major research focus for SDC during the first two years will be to produce a flexible analyzer that integrates application of syntactic and semantic constraints. The flexible analyzer will focus more quickly on the correct analysis and will have recovery strategies to prevent syntactic analysis from becoming a bottleneck for subsequent processing.

4.5. SEMANTIC ANALYSIS

The task of the semantic analyzer is to transform the regularized syntactic analysis into a semantic representation. This representation provides unique identifiers for specific equipment components mentioned in the text. It consists of predicates describing states and events involving the equipment, and higher-order predicates capturing the syntactically-expressed time and causal relations. Roughly speaking, the clauses from the syntactic analysis map into states and events, while the noun phrases map into particular objects (there are several exceptions, including nominalizations, e.g., "loss of pressure", and adjectives of state, such as "broken valve"). Accordingly, the semantic analysis is divided into two major parts, clause semantics and noun phrase semantics. In addition to these two main parts, a time analysis component captures the time information which can be extracted from the input.

4.5.1. CLAUSE SEMANTICS (SDC)

Semantic analysis of clauses is performed by Inference Driven Semantic Analysis (Palmer 1985), which analyzes verbs into their component meanings and fills their semantic roles, producing a semantic representation in predicate form. This representation includes information normally found in a case-frame representation, but is more detailed. The task of filling in the semantic roles is used to integrate the noun phrase analysis (described in the next section) with the clausal semantic analysis. In particular, the selection restriction information on the roles can be used to reject inappropriate referents for noun phrases.

The semantics also provide a filtering function, by checking selectional constraints on verbs and their arguments. The selectional constraints draw on domain knowledge for type and component information, as well as for information about possible relationships between objects in the domain. This function is currently used to accept or reject a completed parse. The goal for the flexible analyzer is to apply selectional filtering compositionally to partial syntactic analyses to rule out semantically unacceptable phrases as soon as they are generated in the parse.

4.5.2. NOUN PHRASE SEMANTICS (SDC + NYU)

A noun phrase resolution component determines the reference of noun phrases, drawing on two sources: a detailed equipment model, and cumulative information regarding referents in previous sentences. SDC has concentrated on the role of prior discourse, and has developed a procedure *that* handles a wide variety of noun phrase types, including pronouns and missing noun phrases, using a focusing algorithm based on surface syntactic structure (Dahl, submitted for publication). NYU, as part of its work on the domain model, has developed a procedure *that* can identify a component in the model from any of the noun phrases *that* can name that component (Ksiezyk and Grishman, submitted for publication). After further development, these procedures will be integrated into a comprehensive noun phrase semantic analyzer.

4.5.3. TIME ANALYSIS (SDC)

SDC has started to develop a module to process time information. Sources of time information include verb tense, adverbial time expressions, prepositional phrases, co-ordinate and subordinate conjunctions. These are all mapped into a small set of predicates expressing a partial time ordering among the states and events in the message.

4.6. DOMAIN MODEL (NYU)

The domain model captures the detailed information about the general class of equipment, and about the specific pieces of equipment involved in the messages; this information is needed in order to fully understand the messages. The model integrates part/whole information, type/instance links, and functional information about the various components (Ksiezyk and Grishman, submitted for publication).

The knowledge base performs several functions:

- It provides the domain-specific constraints needed for the semantics to select the correct arguments for a predicate, so that modifiers are correctly attached to noun phrases.
- It enables noun phrase semantics to identify the correct referent for a phrase.
- It provides the prototype information structures which are instantiated in order to record the information in a particular message.
- It provides the information on equipment structure and function used by the discourse rules in establishing probable causal links between the sentences. And finally, associated with the components in the knowledge base are procedures for graphically displaying the status of the equipment as the message is interpreted.

These functions are performed by a large network of frames implemented using the Symbolics Zetalisp flavors system.

4.7. DISCOURSE ANALYSIS (NYU)

The semantic analyzer generates separate semantic representations for the individual sentences of the message. For many applications it is important to establish the (normally implicit) intersentential relationships between the sentences. This is performed by a set of inference rules *that* (using the domain model) identify plausible causal and enabling relationships among the sentences. These relationships, once established, can serve to resolve some semantic ambiguities. They can also supplement the time information extracted during semantic analysis and thus clarify temporal relations among the sentences.

4.8. DIAGNOSTICS (NYU)

The diagnostic procedures are intended to localize the cause of failure of the analysis and provide meaningful feedback when some domain-specific constraint has been violated. We are initially concentrating on violations of local (selectional) constraints, and have built a small component for diagnosing such violations and suggesting acceptable sentence forms; later work will study more global discourse constraints.

REFERENCES

Dahl, Deborah A. (submitted for publication) Focusing and Reference Resolution in PUNDIT.

Grishman, Ralph; Ksiezyk, Tomasz, and Nhan, Ngo Thanh (submitted for publication) Model-based Analysis of Messages about Equipment.

Hirschman, Lynette and Puder, Karl 1986 Restriction Grammar: A PROLOG Implementation. In Warren, D.H.D. and Van Caneghem, M., Eds., *Logic Programming and its Applications.* Ablex Publishing Company, Norwood, New Jersey: 244-261.

Hirschman, Lynette (in press) "Conjunction in Meta-Restriction Grammar." *Journal of Logic Programming.*

Ksiezyk, Tomasz and Grishman, Ralph (submitted for publication) An Equipment Model and its Role in the Interpretation of Nominal Compounds.

Palmer, Martha S. 1985 Driving Semantics for a Limited Domain. Ph.D. thesis. University of Edinburgh.

Palmer, Martha; Dahl, Deborah; Schiffman, Rebecca; Hirschman, Lynette; Linebarger, Marcia; and Dowding, John 1986 Recovering Implicit Information. To appear in *Proceedings of the 24th Annual Meeting of the Association for Computational Linguistics.*

Sager, Naomi and Grishman, Ralph 1975 The Restriction Language for Computer Grammars of Natural Language. *Communications of the ACM* 18: 390-400.

Sager, Naomi 1981 *Natural Language Information Processing: A Computer Grammar of English and its Applications.* Addison-Wesley, Reading, Massachusetts.

CALL FOR PAPERS

ESCOL 86

10-12 October 1986, University of Pittsburgh and Carnegie-Mellon University

The 1986 Eastern States Conference on Linguistics will include demonstrations of natural language processing software. The invited speakers are Charles Fillmore and Lily Wong Fillmore from the University of California at Berkeley, Martin Kay from the Xerox Palo Alto Research Center, and George Miller from Princeton University.

Original, unpublished papers on any topic of general linguistic interest are invited for the general sessions. For the special session, Linguistics at Work, we invite papers on applied linguistics, especially in the areas of language teaching and computational linguistics.