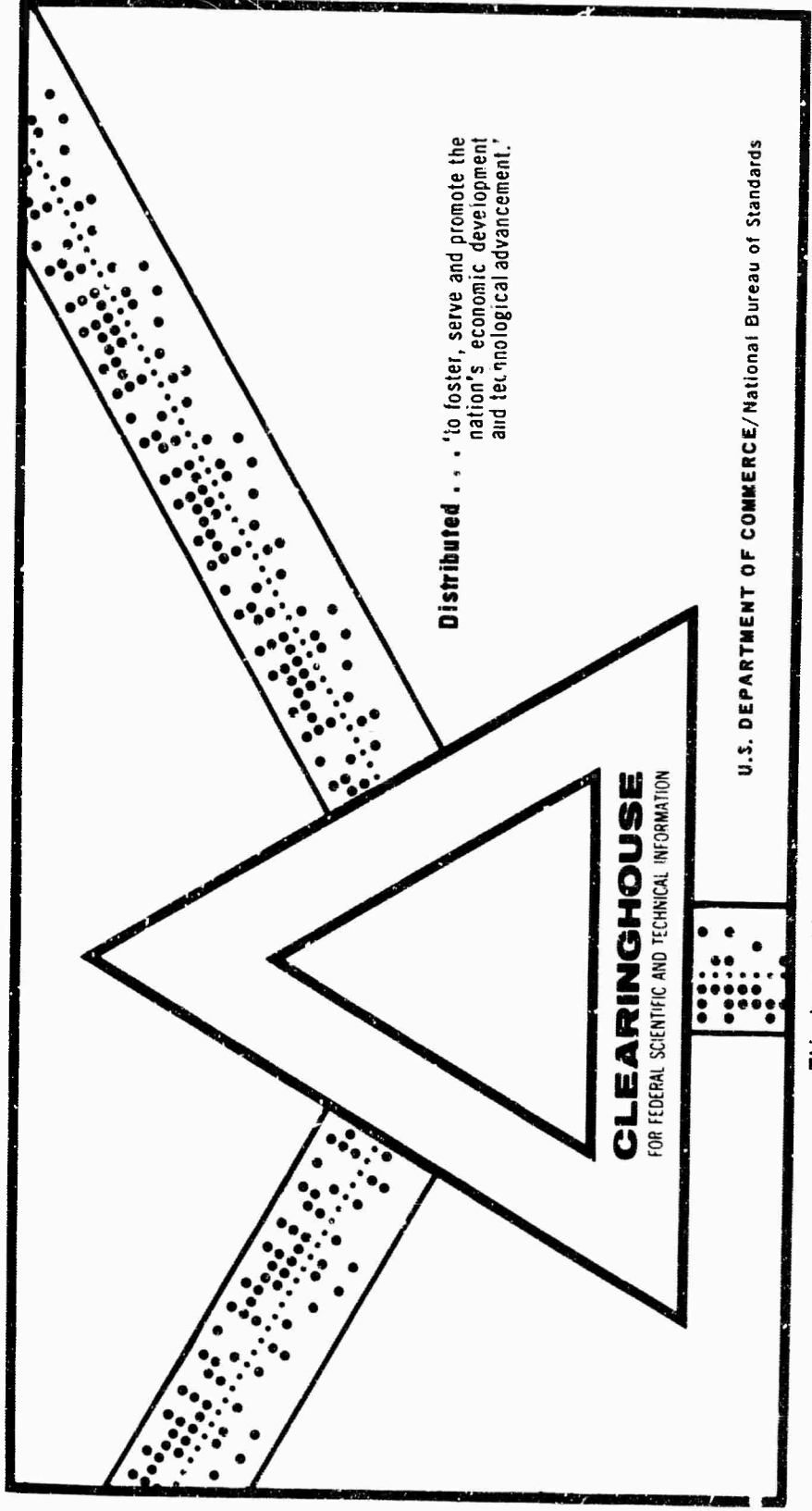


NATURAL COMMUNICATION WITH COMPUTERS II

Daniel G. Bobrow

Bolt Beranek and Newman, Incorporated
Cambridge, Massachusetts

31 October 1969



This document has been approved for public release and sale.

AD700817

AFCRL-69-0523

NATURAL COMMUNICATION WITH COMPUTERS II

Daniel G. Bobrow

Bolt, Beranek and Newman Inc.
50 Moulton Street
Cambridge, Massachusetts 02138

Contract No. F19628-68-C-0125
Project No. 8668



Final Report

Period Covered: 1 August 1967 through 30 September 1969

31 October 1969

This research was supported by the Advanced Research Projects
Agency under ARPA Order No. 627

This document has been approved for public release and sale, its
distribution is unlimited

Contract Monitor: Hans Zschirnt
Data Sciences Laboratory

Prepared for:

AIR FORCE CAMBRIDGE RESEARCH LABORATORIES
OFFICE OF AEROSPACE RESEARCH
UNITED STATES AIR FORCE
BEDFORD, MASSACHUSETTS 01730

SEARCHED U.I.
SERIALIZED FILED

AFCRL-69-0523

NATURAL COMMUNICATION WITH COMPUTERS II

Daniel G. Bobrow

Bolt Beranek and Newman Inc.
50 Moulton Street
Cambridge, Massachusetts 02138

Contract No. F19628-68-C-0125

Project No. 8668

Final Report

Period Covered: 1 August 1967 through 30 September 1969

31 October 1969

This research was supported by the Advanced Research Projects
Agency under ARPA Order No. 627

This document has been approved for public release and sale, its
distribution is unlimited

Contract Monitor: Hans Zschirnt
Data Sciences Laboratory

Prepared for:

AIR FORCE CAMBRIDGE RESEARCH LABORATORIES
OFFICE OF AEROSPACE RESEARCH
UNITED STATES AIR FORCE
BEDFORD, MASSACHUSETTS 01730

1) AD-658 829 *2061* 2) *9/2, 5/8*
 4) \$3.00/\$0.65
 5) Bolt Beranek and Newman, Inc., Cambridge, Mass.
 6) NATURAL COMMUNICATION WITH COMPUTERS. (8) 1
 9) Final rept. 15 Mar 65-31 Aug 67,
 10) by Daniel G. Bobrow.
 11) 31 Aug 67, (12) 38p (14) BBN-1548
 15) Contract AF 19(628)-5065, ARPA Order-627-2
 16) Proj. AF-8668 (17) -
 18) AFCLRL (19) 67-0485
 21) -
 22) -

rwg/ M/M

DO NOT PUNCH

Check List for Fields

Added Entry

	DC	SA
1	/	/
2	/	/
4	/	/
5	/	/
6	/	/
8	/	/
9	/	/
10	/	/
11	/	/
12	/	/
14	/	/
15	/	/
16	/	/
17	/	/
18	/	/
19	/	/
21	/	/
22	/	/
23	/	/
25	/	/
27	/	/
29	/	/
30	/	/
33	/	/
34	/	/
35	/	/

Release Date
5 Oct 67

23) Descriptors

(* Computers, * Man-machine systems)

Abstracts,

Semantics,

Syntax,

Programming languages,

Time sharing,

Data storage systems,

Real time,

Compilers,

Information retrieval,

Data processing systems,

Display systems,

Digital computers.

25) Identifiers

LISP,

On-line systems.

List processing.

27) Abstract: See reverse Page (s) --

29) Inventory 19

30) Annotated Title

33) Dist. Code 1

34) Serial F

35) Code 060 100

The report discusses results concerning problems affecting computer communications with people, other computers, and real-time devices, as found within the 2 years of the contract.

ABSTRACT

For four and a half years Bolt Beranek and Newman has been engaged in a continuing research program whose goal is to develop techniques to facilitate Natural Communications between computers and people, other computers, and real time devices. This work was started under a contract monitored through the Air Force Cambridge Research Laboratory, Contract No. AF19(628)-5065 and continued under Contract No. F19628-68-C-0125. This work is continuing under new contracts sponsored by the Advanced Research Projects Agency.

In accordance with our responsibility to report the results of our investigation we have prepared a number of documents, some of which have been distributed as scientific reports under the contract, some of which have been published as technical articles in professional journals, and a few of which are internal memoranda which we have made available to interested outside parties. Following a brief summary of our research we provide a bibliography and abstracts of these documents for further information.

TABLE OF CONTENTS

	page
ABSTRACT	111
I. INTRODUCTION	1
II. SUMMARY OF RESEARCH	3
Representation of semantic information	3
Special purpose languages	3
Real time input-output, and pattern recognition ..	4
Time-sharing research and development	4
Modeling time-sharing systems and networks	5
III. PUBLICATIONS SUPPORTED UNDER CONTRACT F19628-68-C-0125	6
IV. ABSTRACTS	10

SECTION I

INTRODUCTION

For four and a half years Bolt Beranek and Newman has been engaged in a continuing research program whose goal is to develop techniques to facilitate Natural Communications between Computers and people, other computers and real time devices. This work was started under a contract monitored through the Air Force Cambridge Research Laboratory, Contract No. AF19(628)-5065 and continued under Contract No. F19628-68-C-0125. This work is continuing under new contracts sponsored by the Advanced Research Projects Agency.

Our efforts to facilitate man-machine communication has centered around two efforts: an English language comprehension model utilizing a semantic memory network; and development of special purpose languages to facilitate communication within certain problem domains. In order to make feasible direct communication with the real world we have built a real-time interface to our time-sharing computer facility, and made this facility available in several higher level languages, including LISP, which provides the symbolic processing capabilities necessary for three-dimensional pattern recognition, speech recognition and other symbolic communication.

Our time sharing system will participate in the ARPA network of time-sharing systems. In order to be able to understand how such systems work, both individually and in the network, we have done significant work on modeling of a time sharing system, and its users, and of a

network of such systems. In order to provide appropriate input-output facilities for our time-sharing system we have made several design passes and have now procured a high quality display processor, and we have considered problems associated with attaching many low cost display terminals to the system.

In accordance with our responsibility to report the results of our investigation we have prepared a number of documents, some of which have been distributed as scientific reports under the contract, some of which have been published as technical articles in professional journals, and a few of which are internal memoranda which we have made available to interested outside parties. Following a brief summary of our research we include abstracts of these documents for further information.

SECTION II

SUMMARY OF RESEARCH

A. Representation of semantic information

In this task we have been concerned with the parameters and forms necessary to represent the information which can be obtained from English sentences. We have concerned ourselves with some linguistic properties of words and techniques for representing these kinds of relationships in a semantic memory network. Facts implicitly or explicitly stored in this semantic memory can be accessed through questions input in English. The Teachable Language Comprehender (TLC) is a program developed under this task which has illustrated some of the comprehension properties which we feel will be necessary in any English language comprehender.

B. Special purpose languages

Under this task we have developed several special purpose languages which are well matched to certain problems, and have developed a number of techniques to facilitate the building of such languages. We have included languages for display, for signal processing, for symbolic mathematics and for Natural Language grammar and phonological rule specification and testing. Our principal tools have been the programming language LISP, a higher level language which has been used to embed many of the other special purpose languages and FLIP, a Format Directed List Processing extension of LISP. Much effort has been expended to make LISP an excellent on-line interactive language system; the BBN-LISP system includes an interpreter, compiler, editing and debugging facilities and techniques for automatic error correction. Our goal is to build a well equipped automated programming assistant.

C. Real time input-output, and pattern recognition

Under this task we have developed a capability within our time-sharing system for doing real time I/O utilizing a special purpose "hybrid processor" which can communicate with a variety of physical devices, and can service a number of real-time processes simultaneously. Utilizing this device we have worked toward achieving a capability for three-dimensional pattern recognition by attaching an image dissector as an external visual sensor for the computer (an eye). In addition we have incorporated equipment which facilitates the recognition of speech messages in real time, and have studied the properties of speech messages in real-time, and have studied the properties of speech signals as processed through this filter bank (an ear). Programs to recognize a limited vocabulary have been developed with the aim to using this additional communication channel with the computer. The real-time I/O hybrid computer processor has also been interfaced to an analog computer which allows mixed modes of computation, an unusual facility for such a powerful time-sharing system.

D. Time-sharing research and development

Under this task we have extended the time-sharing capabilities of the SDS 940 computer by attaching to it a number of special pieces of test equipment which facilitated communication with people in real time mode and through displays, as well as serving the more conventional needs of programmers and computer users. However the limitations forced on us by the SDS 940 hardware and software have forced us to a new time-sharing system based on DEC PDP-10 hardware. We have completed the design of an extended time-sharing multi-processor system which will be able to handle large files, real-time devices and many users. It will provide a better interactive environment that is now available on the SDS 940 computer. The design includes hardware modifications to the

central processor, and additional equipment such as a paging box to be attached to the PDP-10. In addition, we are building a completely new operating system which can utilize this new equipment efficiently. This design will be implemented under continuing ARPA support. Design documents for this system and for our paging box are being made available to other members of the ARPA community who wish to utilize the result of our efforts.

E. Modeling time-sharing systems and networks

Under this task we have been developing and trying to validate models for computers and information processing models of man essential to the operation of man-computer systems. These include queueing models of the time-sharing system and decision models of the human operator at a console. We have also investigated the problems associated with the design and implementation of computer networks, and developed a simulation program which allows exploration of various configurations of the network and different routing algorithms.

SECTION III

PUBLICATIONS SUPPORTED UNDER CONTRACT F19628-68-C-0125

Reports and Articles Available

Bell, A., Quillian, M.R., "Capturing Concepts in a Semantic Net", AFCRL-69-0438 (BBN Report #1885), October 1969; Associative Information Processing (forthcoming).

Bobrow, D.G., "Information Reduction as a Technique for Limited Speech Recognition", Proceedings of Australian Conference on Picture-Language Machines, Australian National University, February 1969.

Bobrow, D.G., "Natural Language Interaction Systems", Proceedings of Australian Conference on Picture-Language Machines, Australian National University, February 1969.

Bobrow, D.G., Fraser, B., "An Augmented State Transition Network Analysis Procedure", Proceedings of the International Joint Artificial Intelligence Conference, Washington, D.C., May 1969.

Bobrow, D., Fraser, B., "A Phonological Rule Tester", AFCRL-68-0085 (BBN Report #1589), September 1967; Comm. of ACM 11:(11) 766-772, November 1968.

Bobrow, D.G., Murphy, D.L., Teitelman, "The BBN-LISP System", (BBN Report 1677) April 1968, revised April 1969.

Bobrow, D.G., Teitelman, W., "Debugging in an On-Line Interactive LISP", BBN Technical Report, November 1967.

Bobrow, R.T., Bobrow, D.G., "EDMS - An Experimental Data Management System", Proceedings of the Third Hawaii Conference on System Sciences, January 1970.

Carbonell, J., "On Man-Computer Interaction: A Model and Some Related Issues", AFCRL-68-0053, (BBN Report - September 1967); Transactions of System Science and Cybernetics of the IEEE, Vol. SSC-5-No. 1, January 1969.

Carbonell, J., Elkind, J., Nickerson, R., "The Dynamics of Time-Sharing Systems from a Human Factors Point of View", Human Factors 10(2) 127-134, 1968.

Carbonell, J., Elkind, J., Nickerson, R., "Human Factors and the Design of Time-Sharing Computer Systems", AFCRL-68-0054, (BBN Report November 1967); presented at the 11th Annual Mtg. Human Factors Soc., Boston, Mass.

Carbonell, J., Elkind, J., Nickerson, R., "On the Psychological Importance of Time in a Time-Sharing System", AFCRL-68-0120 (BBN Report - September 1967); Human Factors 10, No. 2, 135-142, 1968.

Cooper, G., "A Semantic Analysis of English Locative Prepositions", AFCRL-68-0056 (BBN Report #1568), November 1967.

Fraser, B., Klatt, M., "A Selected Psycholinguistic Bibliography", AFCRL-68-0605 (BBN Report - November 1968).

Grignetti, M., "Microanalysis and Probabilistic Modeling of a Time-Shared System", AFCRL-69-0207 (BBN Report #1756), April 1969.

Kahn, R., Teitelman, W., "A Network Simulation and Display Program", Presented at Third Annual Princeton Conference on Information Sciences and Systems, March 1969. Abstract to be published in the Conference Proceedings.

Myer, T., Sutherland, I., "On the Design of Display Processors", Comm. of ACM, 11:(6) 410-414, June 1968.

Myer, T., Sutherland, I., "A Display Processor Design", Proceedings of the Fall Joint Computer Conference, AFIPS Press, Montvale, N.J., 1969.

Quillian, M.R., "The Teachable Language Comprehender: A Simulation Program and Theory of Language", AFCRL-69-0108 (BBN Report - January 1969); Comm. of ACM 12(8) August 1969.

Stevens, K., "Study of Acoustic Properties of Speech Sounds II", AFCRL-69-0339 (BBN Report #1871), April 1969.

Stevens, K., Klatt, M. "Study of Acoustic Properties of Speech Sounds", AFCRL-69-0339 (BBN Report #1669), August 1968.

Strollo, T., Tomlinson, R., Fiala, E., Elkind, J., "The BBN Hybrid Processor", AFCRL-67-0485, (BBN Report #1686), June 1968.

Strollo, T.R., Tomlinson, R.S., Fiala, E.R., "A Time-Shared I/O Processor for Real-Time Hybrid Computation", Proceedings of the Fall Joint Computer Conference. AFIPS Press, Montvale, N.J., 1969.

Teitelman, W., "Toward a Programming Laboratory", Proceedings of the International Joint Artificial Intelligence Conference, Washington, D.C., May 1969.

Report No. 1893

Bolt Beranek and Newman Inc.

Teller, P., "Some Discussion and Extension of Manfred Nierwisch's Work on German Adjectivals", AFCRL-68-0055 (BBN Report #1568), November 1967.

Report No. 1893

Bolt Beranek and Newman Inc.

SECTION IV

ABSTRACTS

CAPTURING CONCEPTS IN A SEMANTIC NET

T. Bell
R. Quillian

A working memory model based on a semantic network is described in detail. Some advantages and disadvantages of such a model are discussed. An attempt is made to enable a reader to learn to perform the formidable task of representing data in the memory format. Since the actual memory is not easily read (or written), a set of LISP programs are included which make these tasks manageable.

INFORMATION REDUCTION AS A TECHNIQUE FOR LIMITED SPEECH RECOGNITION

D. Bobrow

Most approaches to speech recognition have first attacked the problem of fragmenting words into sub-units of phonemes. However these abstract units do not exist outside the context of the syllables and words in which they appear. This paper describes a system which operates within a limited task environment and recognizes a limited vocabulary of words or phrases by considering both the local context of the sounds and the global context of the messages. This systems approach allows a high recognition rate of the input message set.

NATURAL LANGUAGE INTERACTION SYSTEMS

D. Bobrow

A number of English language question-answering and comprehension systems have been developed. The principal problems involved are the representation of the question in sufficient detail to be able to extract the information from the data base; and the parallel question of how to organize data bases in a flexible enough fashion to make information readily accessible. In this paper, we survey a number of such systems and describe the techniques used.

AN AUGMENTED STATE TRANSITION NETWORK ANALYSIS PROCEDURE

D. Bobrow
B. Fraser

A syntactic analysis procedure is described which obtains directly the deep structure information associated with an input sentence. The implementation utilizes a state transition network characterizing those linguistic facts representable in a context free form, and a number of techniques to code and derive additional linguistic information and to permit the compression of the network size, thereby allowing more efficient operation of the system. By recognizing identical constituent predictions stemming from two different analysis paths, the system determines the structure of this constituent only once. When two alternative paths through the state transition network converge to a single state at some point in the analysis, subsequent analyses are carried out only once despite the earlier ambiguity. Use of flags to carry feature concordance and previous context information allows merging of a number of almost identical paths through the network.

A PHONOLOGICAL RULE TESTER

D. Bobrow
B. Fraser

In this paper, we report on the design and implementation of a system to alleviate the problem of rule evaluation for the linguist in the area of phonology. The system permits the user to define, on-line, sets of rules statable within the framework presented in The Sound Patterns of English (N. Chomsky and M. Halle, 1968), to define phonemes as bundles of specified distinctive features, to define data as strings of phonemes with associated grammatical structure, to test the effect of applying rules to the data, and to store both the definitions and results. The system was written in BBN-LISP (Bobrow et al., 1968), a format directed list processor embedded in LISP. This made the system construction easy while providing very sophisticated capabilities for the linguist. The system is designed to be used on-line in interactive fashion, with control returned to the user after each command is executed.

THE BBN-LISP SYSTEM

D. Bobrow
D. Murphy
W. Teitelman

This document describes the BBN-LISP system currently implemented on the SDS 940. It is a dialect of LISP 1.5 and the differences between IBM 7090 version and this system are described in Appendix 1 and 2. Principally, this system has been expanded from the LISP 1.5 on the 7090 in a number of different ways. BBN-LISP is designed to utilize a drum for storage and to provide the user a large virtual memory, with a relatively small penalty in speed using special paging techniques described in Bobrow and Murphy (1967). Secondly, this system has been designed to be a good on-line interactive system. Some of the features provided include sophisticated debugging facilities with tracing and conditional breakpoints, a sophisticated LISP oriented editor within the system, and compatible compiler and interpreter. Utilization of a uniform error processing through a user accessible function has allowed the implementation of a do-what-I-mean feature which can correct errors without losing the context of the computation. The philosophy of the DWIM feature is described in Teitelman (1969). In addition to the sub-systems described in this manual, a complete format directed list processing sub-system (FLIP, Teitelman, 1967) is available within BBN-LISP. There is also an assembler for inserting machine code sub-routines within BBN-LISP, and facilities for using the CRT display and CALCOMP plotter.

DEBUGGING IN AN ON-LINE INTERACTIVE LISP

D. Bobrow
W. Teitelman

LISP 1.5 was originally designed to run in a batch system on the IBN 7090, not in an interactive time-sharing system. BBN-LISP for the SDS-940 has been designed to take advantage of its time-shared on-line environment. There are several aspects which must be considered in making an on-line system convenient to use. One of these is to provide a convenient way within the system for modifying function definitions. This is considered in the paper on the LISP Editor, by Peter Deutsch, which is part of the BBN system. The second involves facilitating error detection and debugging in LISP programs.

There are two basically different classes of errors which a user can make, and these must be handled in different ways. The first is an error which causes the LISP system itself to become unhappy. Typical of this type of error is one occurring when a function is called when it is not yet defined, or one where the value of a variable is needed and this variable has not yet been bound. For these types of errors, it would be most useful if the system would not throw up its hands in disgust and just type an error message as in LISP 1.5. LISP should do something graceful to allow the user to correct the error and continue from the point where the error was discovered. The technique for achieving this effect in BBN-LISP is described below.

EDMS - AN EXPERIMENTAL DATA MANAGEMENT SYSTEM

R. Bobrow
D. Bobrow

This paper describes an experimental data-management system (EDMS) implemented within BBN-LISP, a high-level, time-shared, interactive list-processing system which allows associative data structures to be conveniently manipulated in a large virtual memory. Our purpose in constructing this system was to obtain both a useful tool for in-house problems such as contract management, and a flexible instrument for experimentation. The BBN-LISP system was selected because of its unique and powerful capabilities for handling data structures, and because of the convenience of the BBN-LISP environment. The purpose of this paper is two-fold: to describe EDMS, and in particular those aspects of its data structure which represent extensions of the structures of previously available systems and the aspects of its design which make it a convenient tool for experimental investigation of various problems in the design of data management systems; and to discuss the advantages and disadvantages of constructing such a system within a higher level language and interactive operating system such as BBN-LISP, both from the point of view of ease of construction and modification of an experimental system, and from the point of view of efficiency of the end product as an operational data management system.

We have implemented in BBN-LISP a data management system which is both a useful tool and an experimental system; the flexibility introduced through the BBN-LISP environment has made the study and investigation of the following subjects extremely convenient:

1. Languages for querying hierarchically and recursively structured data bases.

2. New data structures, including items whose description allows arbitrary recursive hierarchial nesting of subitems (recursive groups) as well as arbitrary repetition of subitems on the same level of the hierarchy (Repeating groups as in SDC/TDMS).
3. New types of elementary data items, (e.g., functions embedded within the data file) allowing improvement in the flexibility and generality of data storage, access and deduction.
4. Search and indexing techniques which make efficient use of the known hierarchial and recursive structure of the data base.
5. Flexible mechanisms for file security - permitting independent access keys for different items and classes of data items, and for the hierarchial relations between items as well.
6. Techniques for system design to permit the final user to tailor the system to his own needs in a flexible manner, without causing system performance to deteriorate drastically.

The needs of users and prospective users have suggested several of the new and experimental features, and feedback from the users on the performance of the system is providing a useful evaluation of such features.

The second class of error which we must consider are those logical errors which do not upset the LISP system but still prevent a program from accomplishing its intended task. By far, most of the effort of debugging is spent in locating these often subtle and well hidden bugs. The TRACE function usually provided in LISP gives some help, but it is not selective enough, and is designed primarily for non-interactive use. Preferred would be a facility comparable to the standard debugging features associated with machine language code and such debugging aids as DDT. These allow "break points" to interrogate the status of his program at selected points. Unfortunately, it is unusual to have such facilities available in higher level languages. Ever scarcer is the capability in debugging machine languages programs of "halting" the computer safely at almost any time, and examining symbolically the transient state of the program and data. It has been our intention in designing the debugging features in BBN-LISP to provide such capabilities.

A critical point in the design of BBN-LISP for achieving this goal is that even when running compiled code, the name of each variable is retained along with its value. Thus all items can be referenced symbolically. In addition, since these items are all kept on the push-down list, one can examine and change the status of this push-down list either under program control, or from a break-point.

With this overview, we now discuss the facilities available in BBN-LISP for debugging and error handling. Section V provides a summary of all of the functions discussed in Sections II, III and IV, and can be used as a quick reference.

ON MAN-COMPUTER INTERACTION:
A MODEL AND SOME RELATED ISSUES

J. Carbonell

A survey of the literature related to man-computer interaction reveals the many aspects of this problem, which appears to be in the crossroads among such diverse fields as computer languages, computer systems operational characteristics, control theory, decision theory, information theory, applied psychology, computer display and interface engineering, etc. In this paper, we have chosen to present the on-line interaction from an information and decision point of view. After a brief discussion of classes of on-line situations and tasks, we propose a model of the case in which a human operator is engaged on-line in the solution of a problem like debugging a program, testing a model in a scientific application, or performing a library search. In this model the human operator is considered to seek to maximize the information he possesses about his problem at a minimum overall cost. This cost is obtained by adding the operational cost of both man and computer to a remnant terminal cost originated by the remaining uncertainty. This analysis, performed for each of a set of possible alternatives for action, any lead to select and execute one of them, to terminate the process, or to re-evaluate the possible alternatives and/or hypotheses in a search for new ones. Some practical applications in terms of response time and other characteristics of a computer utility will be discussed, as well as some theoretical implications from an informational point of view.

THE DYNAMICS OF TIME-SHARING SYSTEMS
FROM A HUMAN FACTORS POINT OF VIEW

J. Carbonell
J. Elkind
R. Nickerson

One of the most important aspects in the design and/or operation of a computer utility is to obtain dynamical characteristics acceptable and convenient to the on-line user. In this paper we are concerned with the problems of access to the computer utility and general availability of it, response time and its effect upon conversational use of the computer, and the effects of the load on the system (and its fluctuations) upon the other aspects.

Primary attention is placed upon response time. Some of the difficulties in its definition are pointed out through examination of the typical interaction process. It is concluded that rather than a single measure a set of response times should be measured in a given computer utility, in correspondence to the different types of operations requested. Next, it is tentatively assumed that the psychological value of short response times stems from a subjective cost measure of the user's own time, largely influenced by the value of concurrent tasks being postponed. A measure of cost (to the individual and/or his organization) of the time-on-line required to perform a task might thus be derived.

More subtle is the problem of the user's acceptability of given response times. This acceptability is a function of the service requested (e.g., length of computations), and variability with respect to the expectations due both to uncertainty in the user's estimation and to variations in the response time originated by variable loads on the system.

This paper concludes with a strong advocacy that an effort be made by computer-utility designers to include dynamic characteristics (such as prediction of loads and their effects) among their design specifications. To achieve this goal, more research both on the human factors and systems aspects of this problem is urgently needed.

HUMAN FACTORS AND THE DESIGN OF TIME SHARING COMPUTER SYSTEMS

J. Carbonell
J. Elkind
R. Nickerson

The advent of computer time sharing poses an extraordinary challenge to human factors research during the next decade. Before time sharing, two facts combined to de-emphasize the importance of human factors considerations in the design of computer systems: (1) the cost of the computer's time was exorbitantly high relative to the cost of users' time, and (2) the users constituted a select, highly skilled and highly motivated group of specialists. Two of the promises of time sharing, however, are (1) a drastic reduction in the cost of computer time to the individual user, and (2) the large scale availability of computer facilities to individuals untrained in any areas of computer technology. Human factors considerations then become important both for economic and psychological reasons. This paper briefly notes what a few of these considerations are.

ON THE PSYCHOLOGICAL IMPORTANCE OF TIME IN A
TIME SHARING SYSTEMJ. Carbonell
J. Elkind
R. Nickerson

One of the most important problems in the design and/or operation of a computer utility is to obtain dynamical characteristics that are acceptable and convenient to the on-line user. In this paper we are concerned with the problems of access to the computer utility, response time and its effect upon conventional use of the computer, and the effects of the load on the system (and its fluctuations) upon the other aspects.

Primary attention is placed upon response time. Some of the difficulties in its definition are pointed out through examination of the typical interaction process. It is concluded that rather than a single measure a set of response times should be measured in a given computer utility, in correspondence to the different types of operations requested. Next, it is tentatively assumed that the psychological value of short response times stems from a subjective cost measure of the user's own time, largely influenced by the value of concurrent tasks being postponed. A measure of cost (to the individual and/or his organization) of the time-on-line required to perform a task might thus be derived.

More subtle is the problem of the user's acceptability of given response times. This acceptability is a function of the service requested (e.g., length of computation), and variability with respect to expectations due both to uncertainty in the user's estimation and to variations in the response time originated by variable loads on the system.

This paper presents a strong advocacy that an effort be made by computer-utility designers to include dynamic characteristics (such as prediction of loads and their effects) among their design specifications. To achieve this goal, more research both on the human factors and systems aspects of this problem is urgently needed.

A SEMANTIC ANALYSIS OF ENGLISH LOCATIVE PREPOSITIONS

G. Cooper

Locative prepositions are members of a small closed class of English formatives. We present here an analysis of the thirty three English locative prepositions in the context of a theory which provides a basis for extracting semantic readings for phrases containing these words.

The meaning of a preposition is first analyzed with function concepts which pick out the relevant characteristics of objects to be related. Then a relation concept is specified which describes the special relation between the value of functions. The resulting complex relation marker will thus have the form $R(f(x),g(y))$ where f and g are function markers, R is a relation marker, and x and y are the objects to be related

A SELECTED PSYCHOLINGUISTIC BIBLIOGRAPHY

B. Fraser
D. Klatt

The compilation of this bibliography was born from the frustration in locating relevant references in the organization and teaching of a graduate course in psycholinguistics during the 1967-68 academic year. To alleviate this lack of collected references, we have attempted in this document to provide a representative, un-dogmatic, and fairly thorough coverage of selected areas of the psycholinguistic literature which are not adequately covered in existing bibliographies. Because of our view that psycholinguistics consist of the study of the acquisition, production and understanding of a natural language, we have deliberately excluded from consideration any literature in areas such as language and culture, phonetic symbolism, paired association testing, concept formation, anthropological linguistics, and speech pathology. This is not to deny the importance of any of these but to indicate that they appear to us to lie well on the periphery of the field of psycholinguistics. We have, however, included short sections on animal communication and language disorder, mostly because the material was interesting to us. Since there are a number of good and reasonably current bibliographies in the area of syntax and phonology, we have omitted these references and have cited the reviews in Sec. I. Moreover, a forthcoming very complete and up-to-date bibliography by Daniel I. Slobin on child language acquisition has permitted us to leave this area less complete than we ordinarily would have done. Each citation occurs only once in this bibliography, and in each case we have attempted to place it in the most obvious section even though two or more sections might be applicable.

At the top of each page appears an abbreviated section title (e.g., Anthologies, Bibliographies, Surveys; Psychology, Psycholinguistics; Speech Analysis; Semantics; etc.). We believe that this referencing system will facilitate access to the subject bibliographies.

MICROANALYSIS AND PROBABILISTIC MODELING OF A TIME SHARED SYSTEM

M. Grignetti

A series of measurements was performed on a version of a Telcomp system operating on a DEC-PDP-1 computer. These measurements were designed to gain insight into the dynamic behavior of a time-shared system and to define and find the operating point of the system. In particular, we wanted to find out what system parameter or combination of parameters was limiting system performance. We also wanted to formulate a dynamic model of the system which could be used to predict what would happen if the system parameters were changed.

In this report we describe the static structure of the Telcomp system and present a probabilistic model of its behavior. We also describe the measuring system and our attempts to establish the extent to which the measuring system perturbed the Telcomp system performance. We avoided compounding the problem of merging two sets of statistics (those of the user's world and those of the system itself) by designing an experiment with fictitious users performing one and the same benchmark task. Our data is presented and we point out significant features of operations of the system. From the data there emerges a dynamic picture of the system, which is illustrated with the help of diagrams consisting of the sequences of events from the point of view of the user and from the point of view of the system. From the data we refine our preliminary model for system behavior and use this new model as the basis for simulation studies in which the observed results are reproduced; system performance with various parameter changes is then predicted from the model.

A NETWORK SIMULATION AND DISPLAY PROGRAM

R. Kahn
W. Teitelman

We describe a network simulation and display program written in BBN-LISP which provides a convenient and flexible way to study several important aspects of networking. The program is designed to serve as a tool for studying the ways in which networks become clogged and to explore algorithms and heuristics for unclogging them. In addition, it is a useful tool to illustrate the effect on message delay of routing algorithms and the network structure and topology.

The network program deals with nodes, lines and message packets. The user enters the coordinate location of each node, a node name, and a list of nodes to which it is connected. The network program then finds the various routes from each node to every other and makes a record of the shortest route, next shortest, and so forth. The program displays the nodes as small circles and the connections as dotted lines. Nearby each node is displayed the node name. Message packets are generated by the user. If he types STOP, the network program completes its current cycle and is then ready to accept input commands from the user. One input command generates a message of $N > 0$ from Node A to Node B. Another input command connects or disconnects a line which was initially specified. A third command allows the user to affect the routing of specific message packets or to modify the global routing strategies. A fourth set of commands allows the user to interrogate the status of the network. The program operation may be continued by typing OK.

Each node can store a finite number of message packets, where the capacity of a node may be selected by the user. A packet proceeds from Node A to Node B - one node at a time. At each node, queues are formed of packets awaiting transmission to neighboring nodes. The transfer of a packet from one node to another is shown on the display by a brightening line between the two nodes. When a packet reaches the destination node it waits until all the remaining packets finally arrive. At that time an end-to-end acknowledgement is transmitted to the originating node, a message is printed on the teletype which indicates the arrival of the message and the number of cycles used, and the message is deleted from the system.

The speed of the network activity may be adjusted by selecting the length of time a line is to be brightened. In addition, along with the node name at each node are displayed two numbers, the total number of packets at that node waiting to go out to another node, and the total number waiting at that node as a destination node for other packets to arrive. Thus, at a speed which is convenient for the user, the essential network operation will be observable directly on the display.

ON THE DESIGN OF DISPLAY PROCESSORS

T. Myer
I. Sutherland

The flexibility and power needed in the data channel for a computer display are considered. To work efficiently, such a channel must have a sufficient number of instructions that it is best understood as a small processor rather than a powerful channel. As it was found that successive improvements to the display processor design lie on a circular path, by making improvements one can return to the original simple design plus one new general purpose computer for each trip around. The degree of physical separation between display and parent computer is a key factor in display processor design.

A DISPLAY PROCESSOR DESIGN

T. Myer
I. Sutherland
M. Vosbury
R. Watson

This paper describes a general purpose display system for the SDS-940 time-shared computer. The design reported here departs in a number of ways from previous practice and current trends. Most significantly, it includes a push-down stack system in which saved information is marked with the register of its origin, and which allows the programmer to create disjoint stacks whose segments are automatically chained together.

A close coupling was desired between display processor and parent computer. To achieve this, the display shares the main computer's core memory, and the two processors are also tied closely together. This coupling has been achieved without serious degradation of the time-sharing computer.

The display processor was viewed more as a computer than an I/O device. This approach is reflected in the handling of display consoles and related equipment, and in the use of microcoding and a uniform bussing scheme in the display processor's internal design.

THE TEACHABLE LANGUAGE COMPREHENDER:
A SIMULATION PROGRAM AND THEORY OF LANGUAGE

R. Quillian

The Teachable Language Comprehender (TLC) is a program designed to be capable of being taught to "comprehend" English text. When text which the program has not seen before is input to it, it comprehends that text by correctly relating each (explicit or implicit) assertion of the new text to a large memory. This memory is a "semantic network" representing factual assertions about the world.

The program also creates copies of the parts of its memory which have been found to relate to the new text, adapting and combining these copies to represent the meaning of the new text. By this means, the meaning of all text the program successfully comprehends is encoded into the same format as that of the memory. In this form it can be added into the memory.

Both factual assertions for the memory and the capabilities for correctly relating text to the memory's prior contents are to be taught to the program as they are needed. TLC presently contains a relatively small number of examples of such assertions and capabilities, but within the system notations for expressing either of these are provided. Thus, the program now corresponds to a general process for comprehending language, and provides a methodology for adding the additional information this process requires to actually comprehend text of any particular kind.

The memory structure and comprehension process of TLC allow new factual assertions and capabilities for relating text to such stored assertions to generalize automatically. That is, once such

an assertion or capability is put into the system, it becomes available to help comprehend a great many other sentences in the future. Thus, adding a single factual assertion or linguistic capability will often provide a large increment in TLC's effective knowledge of the world, and in its overall ability to comprehend text.

The program's strategy is presented here as a general theory of language comprehension.

STUDY OF ACOUSTIC PROPERTIES OF SPEECH SOUNDS

D. Klatt
K. Stevens

The spectral and temporal characteristics of American English vowel and consonant sounds in a variety of phonetic contexts are examined and compared with data reported in the literature. Spectrograms and sampled spectra (obtained from an analog filter bank connected to a digital computer) were assembled for a number of monosyllabic and bisyllabic utterances generated by three talkers, and a variety of measurements were made from these displays. The characteristics examined include durations of vowels, duration of various phases of consonants in prestressed and post-stressed positions and in clusters, spectra of vowels and diphthongs and their variation with time, spectra of consonants during constricted intervals, and time-variation of spectra during the release of consonants. The aim of the study is not to present an exhaustive acoustic-phonetic description of American English speech sounds but rather to indicate the kinds of acoustic properties that need to be utilized in schemes for machine recognition of speech.

STUDY OF ACOUSTIC PROPERTIES OF SPEECH SOUNDS II

K. Stevens

The acoustic properties of a number of different speech sounds as they appear in several phonetic contexts are described. This report supplements an earlier report on the same topic and presents data for stop and nasal consonants in prestressed position, for the timing of vowels, and for acoustic events following stressed vowels. The aims of this survey are to provide an indication of the kinds of acoustic attributes that should be extracted from the speech signal in a potential scheme for machine recognition of speech. Also included is a discussion of the roles that must be played by acoustic data and by linguistic constraints in schemes for automatic speech recognition.

THE BBN HYBRID PROCESSOR

J. Elkind
E. Fiala
T. Strollo
R. Tomlinson

Real-time experiments present special problems to a time-shared computer system. For example, simulations, waveform analysis, waveform generations, and displays require very high-rate, closely timed input/output between the digital computer's core memory and external devices. In many cases, conventional input/output methods fail to satisfy the requirements of the experiment. For this reason, Bolt Beranek and Newman Inc., has developed the concept of a special purpose "Hybrid Processor" to perform real-time input/output.

This report describes the hardware and software for the Hybrid Processor which has been constructed for BBN's SDS-940 computer, and the revisions planned for the Hybrid Processor which will be designed for the PDP-10.

A HYBRID PROCESSOR FOR A TIME-SHARE SYSTEM

J. Elkind
T. Strollo

The synchronization of a digital computer to the demands of the real world can use a large percentage of a digital computer's capacity. In a system where time-sharing of several real-time processes is involved, the real-time demands may saturate the digital computer. We have designed a special purpose time-shared hybrid processor which relieves the CPU of the burden of synchronization to real-world demands. This processor accurately synchronizes all I/O transfers between the core memory of the digital computer and real-world devices such as A/D converters, D/A converters, digital inputs and digital outputs. Up to four real-time processes can be controlled simultaneously by the Hybrid Processor. These processes may be operating at independent rates so long as the aggregate input/output rate does not exceed 50KC.

A software system is under development which permits use of the Hybrid Processor while both real-time and non-real-time processes are using the computer system. This software will permit up to four real-time processes to operate independently. They will each refer to the hardware in their programs by invariant numbers (e.g. channel numbers on the A/D converter); but they may be physical assigned to different channels. Users will have time-shared access to a number of devices including a visual input device, a hybrid/analog computer, a display, and a number of miscellaneous devices such as relays and shaft encoders.

TOWARD A PROGRAMMING LABORATORY

W. Teitelman

This paper discusses the feasibility and desirability of constructing a "programming laboratory" which would cooperate with the user in the development of his programs, freeing him to concentrate more fully on the conceptual difficulties of the problem he wishes to solve. Experience with similar systems in other fields indicates that such a system would significantly increase the programmer's productivity.

The PILOT system, implemented within the interactive BBN-LISP system, is a step in the direction of a programming laboratory. PILOT operates as an interface between the user and his programs, monitoring both the requests of the user and the operation of his programs. For example, if PILOT detects an error during the execution of a program, it takes the appropriate corrective action based on previous instructions from the user. Similarly, the user can give directions to PILOT about the operation of his programs, even while they are running, and PILOT will perform the work required. In addition, the user can easily modify PILOT by instructing it about its own operation, and thus develop his own language and conventions for interacting with PILOT.

Several examples are presented.

SOME DISCUSSION AND EXTENSION OF MANFRED BIERWISCH'S WORK
ON GERMAN ADJECTIVALS

P. Teller

This article presents some discussion, criticism, and extension of Manfred Bierwisch's recent paper, "Some Semantic Universals of German Adjectivals". Bierwisch draws conclusions about the structure of semantic theory as the result of his attempt to analyze the meaning of selected words and expressions. Correspondingly, the present paper will separate into two parts: (1) an extension of Bierwisch's study of semantic theory, and (2) further analysis of particular expressions. In Part I, I will present certain difficulties that arise from Bierwisch's restructuring of semantic theory along with suggestions for dealing with these difficulties. In Part II, I will present the analysis of some words and expressions related to those that Bierwisch has examined. My analysis will use many of the semantic markers he has proposed as well as others that I have found necessary to add. These examples are further evidence of the usefulness of the semantic markers Bierwisch introduced.*

*Of necessity, a thorough understanding of the present paper will require a thorough familiarity with Bierwisch's work on the part of the reader. Furthermore, all my statements will concern semantics in English rather than German. Most of Bierwisch's examples and analyses seem to have close counterparts in English, so many of the claims I will make about English should be equally valid for German. But, in general, I will not examine the question of whether the two languages require different treatment. Some of the examples treated in Part II will be peculiar to English.

DOCUMENT CONTROL DATA - R & D

Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified

1. ORIGINATING ACTIVITY (Corporate author) Bolt Beranek and Newman Inc. 50 Moulton Street Cambridge, Massachusetts		2a. REPORT SECURITY CLASSIFICATION Unclassified	
3. REPORT TITLE NATURAL COMMUNICATION WITH COMPUTERS II		2b. GROUP	
4. DESCRIPTIVE NOTES (Type of report and, inclusive dates) Scientific. Final. 1 August 1967-30 September 1969			
5. AUTHOR(S) (First name, middle initial, last name) Daniel G. Bobrow		Approved 31 December 1969	
6. REPORT DATE 31 October 1969	7a. TOTAL NO. OF PAGES 42	7b. NO. OF REFS 25	
8a. CONTRACT OR GRANT NO. Contract No. F19628-68-C-0125	9a. ORIGINATOR'S REPORT NUMBER(S) BBN Report No. 1893		
b. PROJECT NO. 8668	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) AFCRL -69-0523		
c. DoD Element 61101D			
d. DoD Subelement n/a			
10. DISTRIBUTION STATEMENT 1-This document has been approved for public release and sale; its distribution is unlimited.			
11. SUPPLEMENTARY NOTES This research was supported by the Advanced Research Projects Agency under ARPA Order No. 627		12. SPONSORING MILITARY ACTIVITY Air Force Cambridge Research Laboratories (CRB) L.G. Hanscom Field Bedford, Massachusetts 01730	
13. ABSTRACT For four and a half years Bolt Beranek and Newman has been engaged in a continuing research program whose goal is to develop techniques to facilitate Natural Communications between computers and people. other computers, and real time devices. This work was started under a contract monitored through the Air Force Cambridge Research Laboratory, Contract No. AF19(628)-5065 and continued under Contract No. F19628-68-C-0125. This work is continuing under new contracts sponsored by the Advanced Research Projects Agency. In accordance with our responsibility to report the results of our investigation we have prepared a number of documents, some of which have been distributed as scientific reports under the contract, some of which have been published as technical articles in professional journals, and a few of which are internal memoranda which we have made available to interested outside parties. Following a brief summary of our research we provide a bibliography and abstracts of these documents for further information.			

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
computer systems						
time-sharing						
man-computer interaction						
phonology						
linguistic rule testers						
linguistics						
transformational grammar						
LISP						
list processing						
format directed list processing						
on-line systems						
semantics						
syntax						
linguistic theory						
human engineering						
man-machine systems						
human factors						
natural language						
language comprehension						
computational linguistics						
psycholinguistics						
speech analysis						
language acquisition						
phonetics						
acoustic phonetics						
speech recognition						
modeling of computer systems						
simulation						
semantic network						
English language comprehension						
memory models						
real time systems						