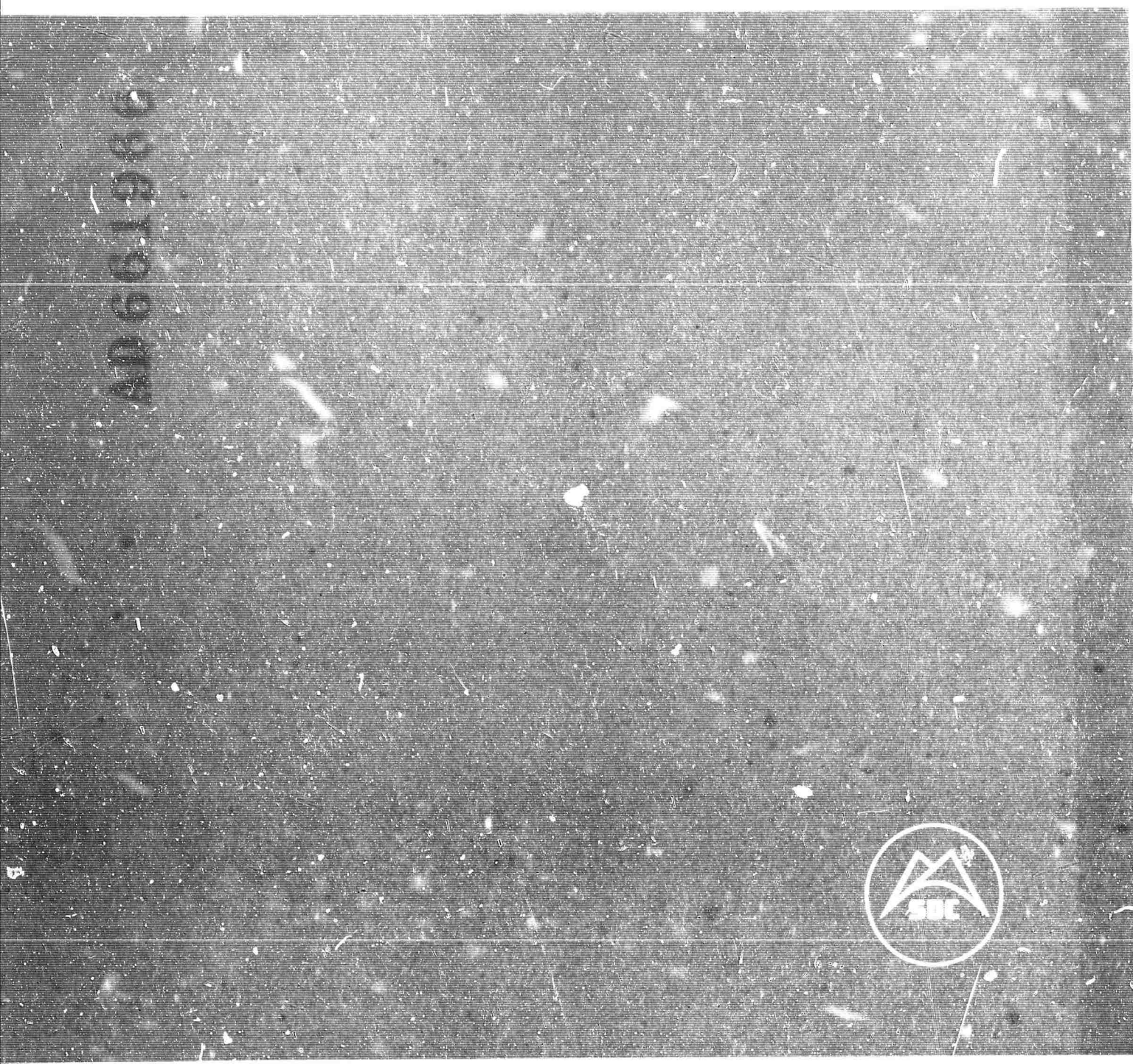


SP-2431 000/00

An Approach to the On Line Interrogation of Structured Files of Facts Using Natural Language

SP-2431/000/00

SP *a professional paper*

This document was produced by SDC in performance of contract AF 19(628)-5166 with the Electronic System[illegible]sion, Air Force Systems Command, in performance of [illegible]rder 773 for the Advanced Research Projects Agency I[illegible]mation Processing Techniques Office.

An Approach to the On-Line Interrogation of Structured Files of Facts Using Natural Language

by

Charles H. Kellogg

April 29, 1966

SYSTEM DEVELOPMENT CORPORATION
2500 COLORADO AVE.
SANTA MONICA
CALIFORNIA
90406

BLANK PAGE

ABSTRACT

Two principal objectives provide the main focus for development of an on-line capability for fact retrieval: (1) formulation of a conceptual framework within which certain issues and problems of fact retrieval may be viewed and clarified; and (2) achievement of a practical and useful fact retrieval capability in the reasonably near future. A balance should be obtained between these two partially interactive, partially conflicting objectives.

Recent insights and work in the theories of linguistic description and in logical explication have aided in pursuit of the first objective. Although much of this theory is tentative in nature, a core of concepts now exists which can be brought to bear on the problem of computer retrieval of facts.

The second objective was approached by selecting a large existing collection of descriptive information as a 'data base' and then, as much as possible, using experimental data management procedures that have been developed for the System Development Corporation/Advanced Research Projects Agency Time-Sharing System.

The conceptual framework was shaped and given substance by introducing and using current conceptions concerning descriptions of the grammar and semantics of natural languages and formalizations of the notions of question and of fact.

In practice, this approach employs an extensible English microgrammar which enables a user to phrase questions in a variety of syntactic patterns. Procedures for coping with semantic ambiguity and paraphrase are also incorporated. The notions of 'explicit fact,' 'general fact,' 'implicit fact,' and 'fact assertion' are introduced and employed to recast the original data into a more useful informational structure.

This information structure is linked to English Question terms through an associative lexical structure--a semiotic network--which reflects grammatic, semantic, and pragmatic features of lexical items.

The principal dichotomy in the framework is between an algorithm for processing questions and the information store. The algorithm can be specialized to deal with varying subsets of English terms, their syntactic patterns, and their associated semantic interpretations. The information store consists of an interpretive information file and a representative information file. The former consists primarily of the information required for algorithm specialization. The latter comprises the 'data' which is available to respond to user questions.

TABLE OF CONTENTS

Illustrations

Illustrations (Cont.)

1. INTRODUCTION

The advent of time-shared computer systems presents the computing community with the new and challenging opportunity of providing users with more powerful and effective tools for problem solving. For example, having facilities for rapidly accessing large files of stored information implies a concomitant need for developing better methods for interrogating the content of these files. User/computer interaction in formulating problems depends on such improvements in communication effectiveness and, consequently, the cooperative problem solving venture itself.

On-line interrogation of structured files is valuable only in proportion to a user's ability to get at sets of relevant facts, to perceive pertinent relationships among these facts, and to manipulate, rearrange, and combine them as required by the task at hand.

This paper is concerned with development of an approach and implementation of a vehicle to enable users to formulate requests more conveniently and to gain access to relevant facts.

Simmons (20)[1] conducted a comprehensive survey of computer-based systems which respond to questions phrased in natural language. Kellogg (16) surveyed systems which utilize more restrictive artificial languages for the expression of queries. Both kinds of systems are examples of structurally fixed question

[1] Numbers in parentheses indicate references presented in the bibliography at the rear of this report.

answering (SFQA) systems[1] An SFQA system responds to questions within a structured environment which is not capable of automatic modification over time with respect to its inputs, i.e., it exhibits no learning. If we contrast the SFQA natural language system (NLS) with the SFQA artificial language system, both of which have been designed to access structured (i.e., nontextual) data, several extralinguistic aspects of these systems become apparent.

To date, natural language systems are experimental in nature. Their question analyzing algorithms are often quite complex; in spite of this fact, their range of application has been largely restricted to narrow subject areas and small data bases. Further, they have not demonstrated a faculty for generalization to different subject areas. An awesome gap exists between present systems and some future general purpose NLS which could, in principle, accept any English question in any one of its conceivable forms. The complexity of the issues at stake in dealing with natural languages has led some theoreticians such as Bar Hillel (1) to pessimism regarding attainment of such goals.

Artificial language SFQA systems are exemplified by Data Management Systems (DMS). Many systems in this category are either operational or in a stage of advanced development. They are designed for the economical handling of large structured information files that are often diverse in content and structure. They also provide for creating and maintaining files and for report preparation.

---

[1] The scope of SFQA systems is described in (21).

The query language component of a DMS often permits the use of English terminology within the scope of severely restricted syntactic patterns. These patterns are often very different from patterns used in conventional English grammar, and permit little use of the multitude of linguistic devices available in natural language. However, query languages are applicable to files of widely varying content because the vocabulary of a query language typically consists of a small set of function words--i.e., Boolean and relational connectives--that are used in any application, and a larger set of content words which stem directly from a specific application.

Contrasting these SFQA systems, we see that the NLS suffers from its present lack of generalizability to serve in operational contexts, and the DMS is limited by the restrictions imposed by its query language. Developers of both kinds of SFQA systems can benefit from the constructive criticism of Giuliano (10) who emphasizes the need for work with larger data bases of more varied structure and scope. Either type of system can be considered as exhibiting an ability for fact retrieval if a useful characterization of the notion of 'fact' is introduced and employed. Such a characterization may be expected to yield a well defined unit of information of intermediate complexity between a 'datum' and a 'file,' much as a sentence serves a similar role between the document in which it occurs and the words which constitute its elementary building blocks.

If we contrast natural language questions with their corresponding formulations in a query language, we see that they differ most often in the explicitness in

which they spell out the conditions for the desired data. For example, the meanings of query language function words are always subject to the same exact interpretation; similarly, content words have unambiguous referents in the file of data. On the other hand, natural language questions will often contain words whose meanings are a function of the context in which they occur. The terms nonprocedural and procedural, respectively, will be used to characterize the difference between questions in which some of the meanings and associations among terms are implicit and queries in which a file searching procedure must be specified explicitly. Ideally, a user should be able to formulate a question in nonprocedural or in procedural terms, whichever best fits his needs and the circumstances.

Existing query languages, then, may be improved in two ways: The first way is to develop a nonprocedural natural language capability employing a subset of the terms and patterns permissable in ordinary English. The second way is to construct improved and more highly interactive procedural languages in which the query formulation process involves both the computer and the user as active partners. This paper discusses the first of these alternatives.[1]

Can a subset of English be developed that will permit the phrasing of a significantly large number of questions likely to be asked of the data base?

---

[1] A method for formulating queries based on the use of a cathode ray tube display device and a light pen for selecting computer displayed options via light buttons has been described elsewhere (16).

Will system users be content to stay within the bounds of a restricted part of their natural language? Significant arguments on the negative side, insofar as syntax analysis is concerned, have been given by Oettinger (17) and Watt (23). The plausible nature of such arguments provides support for the rationale of pursuing additional request formulation procedures such as the light buttor capability just mentioned to supplement and complement a partial natural language interrogation technique. However, the gap between the currently impractical general purpose NLS and currently practical DMS is so great that it seems unwise to prejudge the potential of an English subset of intermediate complexity as a source language for a DMS. This seems especially true in light of recent promising trends in the investigation of syntax and semantics.

In essence the system described in the paper is:

- a structurally fixed question answerer
- specifically designed for fact retrieval
- able to admit nonprocedural requests in a subset of English terms and syntax patterns
- embedded within a time-sharing system environment for on-line use
- capable of driving a large data base.

2. RELEVANT THEORY

The following four areas of theoretical development in linguistics and logic are relevant to this effort:

a. Generative Grammars

The most well developed area of theory and the area behind much of the current work in mathematical linguistics is that of generative grammars, in particular the phrase structure and transformational models due largely to Chomsky (4). Here a grammar is specified in a formalized notation as a device for the recursive generation of sentences through the successive application of sequences of rewrite rules or directed productions. This leads to characterizing the rules of a grammar by a sufficiently explicit and precise means to enable a computer to generate and recognize sentences together with their structural descriptions (i.e., to analyze sentences syntactically).

b. Semantic Theory

Although syntactic analysis of a question is an important first step, semantic analysis is equally important. An outline for a semantic meta theory which seems to provide an adequate basis for developing means for interpreting the meaning of sentences has been proposed by Katz and Fodor at MIT (13). Katz and Fodor indicate how information of a syntactic nature can be associated with a semantic theory. For example, consider the sentences "Gourmets do approve of people eating." and "Gourmets do approve of eating people." Although these two sentences use exactly the same words, the different syntactic constructions cause the sentences to be interpreted differently by most people. Therefore, a theory of semantics must take into account certain kinds of syntactic information.

Katz and Fodor propose two basic components for use in semantic theories: (1) a normal form for entries in a dictionary or lexicon, and (2) a series of rules (projection rules) for selecting and combining the various senses of words presented in lexical normal form. An example of the form of lexical entry proposed by Katz and Fodor is shown in Figure 1. This is the entry for four senses of the word 'bachelor.' Briefly, the entry consists of the word 'bachelor,' followed by a grammatic marker (in this case 'noun') followed by a series of semantic markers, distinguishers, and selection restrictions. In this example, semantic markers are surrounded by parentheses, distinguishers by square brackets, and selection restrictions by angle brackets.

A semantic marker is a sign standing for a semantic property or relation which in turn represents some or all of the meaning of a particular sense of a word. Thus, at least one semantic marker is required for each nonidiosyncratic word sense. In the case of our example, 'bachelor' has four meanings: (1) a human male who has never married; (2) a human male as a young knight serving under the standard of another knight; (3) a human who has the first academic degree; and (4) a young male fur seal without a mate during breeding time. Semantic markers are the prime means for selecting one or several senses of a word from the context in which the word is embedded. For example, the first sense of the word 'bachelor' (a human male who has never married) is identical with one sense of the word 'spinster' with the exception that, for the latter term the marker '(female)' replaces the marker '(male).'

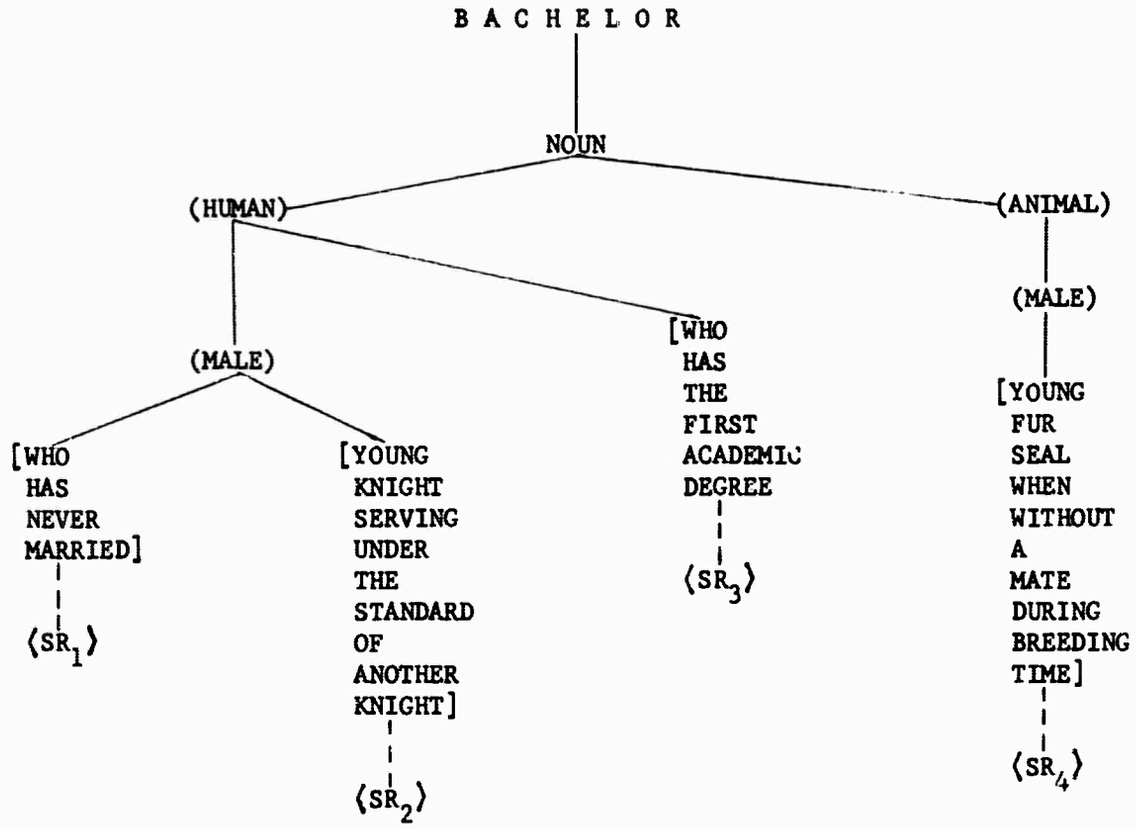


Figure 1. The Structure of a Dictionary Entry

Distinguishers contain information that represents the idiosyncratic meaning of a word. Selection restrictions provide a means for selecting one or more senses of syntactically linked words within a sentence by means of boolean combinations of grammatic markers and/or semantic markers. The selection is performed through the use of projection rules--rules which project onto the infinite variety of sentences possible in a natural language just those interpretations or readings of a sentence which are semantically legitimate. These rules apply the selection restrictions given in lexical entries as part of the process of projecting the known meanings of components onto combinations of components.

Katz and Fodor indicate how projection rules may act upon lexical entries to cope with semantic anomalies, semantic ambiguities, and paraphrase. A semantic anomaly is a grammatical expression of English such as 'the shadow burned up' which does not ordinarily receive a semantic interpretation. Semantically ambiguous combinations of words, on the other hand, have more than one meaningful interpretation. A phrase such as 'the bill is large' is ambiguous; however, its meaning becomes clear in a larger context--'the bill is large but need not be paid.' Users of a natural language, in addition to a capability for recognizing semantically anomalous constructions and semantically ambiguous constructions, are able to paraphrase--to say the same thing in different words, or in different syntactic arrangements of the same words.

Theoretical work on grammar and semantics provides some guide lines about how to attack the problems of constructing a recognition grammar, how to construct entries in a lexicon, and how to determine the meanings of words and phrases from their context. However, the question of how the syntactic and semantic analysis of a question can be related to a structured data file remains. This is precisely the area where several recent efforts in applied logic are useful. We refer to recent explications of fact, question and answer.

c. A Logic of Questions

Belnap's logic of questions (2) considers questions as a subclass of the class of propositions. He discusses an initial framework to be used for investigating the logical structure of questions and for classifying questions according to their completeness and the kinds of answers that are required. According to Belnap, a question consists of a set of alternatives and a request. A request specifies an appropriate form for a direct answer to a question by indicating which of the alternatives in a set of alternatives may be included in an answer. He presents a formal analysis for six types of questions. Besides presenting some insight into the logical structure of questions and their answers, these question types represent a rudimentary classification scheme for questions. Such a scheme may be of use in characterizing the question answering capability of SFQA systems. We will introduce and employ those six question types later in this article when we discuss question analysis.

d. A Logic of Facts

Travis' article on analytic information retrieval (21) presents a similar clarification of the notion of what constitutes a 'fact.' This explication also draws on the logic of propositions. By linking the structures of computer based files of information and the field of logic, Travis has established the beginnings of a foundation upon which fact retrieval systems may be rationally designed and constructed. His starting point is the definition of a fact as a nonarbitrary association between a member of the domain of discourse (i.e., an object like "Los Angeles," "Florida," or "Northeast"), and a datum (a sequence of symbols--a "value"). In turn, this nonarbitrary association may be characterized more precisely in terms of a fact function, as follows: "By fact function, we mean a grouping which associates for each moment of time and according to some definite rule each member of a domain of discourse (or each N-tuple of members for some N) with one and only one particular datum type" (21).

A fact, then, is simply a member of the set of N-tuples which define a fact function (e.g., "Chicago, 1960, 3,550,404"). If we ignore time, and the notion of fact function is further restricted to the datum type that admits only "true" or "false," the familiar notion of a propositional function remains. The semantic significance of both fact and propositional functions hinges on the concept of a "definite rule." The definite rule behind a fact or propositional function is determined by the meaning of

some word or string of words that constitutes a predicate (a name for a property or a relation).[1] The existence of a particular fact is indicated by a fact assertion. For example, the fact "Chicago, 1960, 3,550,404" may be asserted as "population (Chicago, 1960) is 3,550,404." A fact assertion makes the predicate (i.e., the property: population), the argument of a fact function, and the associated value explicit. The fact assertion, an information unit that can meaningfully stand by itself, is valuable as an answer to a question.

3. TRANSLATING SOURCE LANGUAGE QUESTIONS INTO TARGET LANGUAGE REQUESTS--CONSTRUCTION OF A FRAMEWORK

A framework for interpreting questions with respect to a structured data file should provide a superstructure within which one can characterize the information to be stored, the source language used to ask questions, the target language for representing file searching requests, and the several procedures required to effect translation from question to request.

Two basic kinds of information must be stored. The first kind consists of the actual data comprising a data base; this information constitutes the data file part of the information store or the representative information file. The second kind of information is comprised of rules and of the definitions of lexical entries; this information constitutes the interpretive information file. This second file of information is used by a translation algorithm to

[1] These predicates might be formally characterized through the use of meaning postulates as proposed by Travis in his article.

analyze and interpret questions with respect to the basic data stored in the first type of file.

a. Representative Information File

The data base for the representative information file portion of the store was selected on the basis of the following criteria: (1) A data base with a fairly simple structure was desired so that the more basic problems of file interrogation could be attacked first. However, the data base should have a structure similar to the data bases used for computer searching. (2) The data base should be large enough so that the demonstration of retrieval capabilities would be on a realistic scale. (On-line interrogation capabilities may only be economically justified for fairly large files.) (3) The data base should be both inherently interesting and easily understood.

A data base (available on magnetic tape) was found that seemed to meet these requirements. This base was obtained from the Bureau of the Census, and consists of information (derived from the 1960 census) about certain social and economic characteristics of the political subdivisions of the United States. The information in this data base is also available in reference (3). The base consists of approximately 600,000 items of information concerning political subdivisions such as urban areas, counties, states, regions, etc. Since the base is available in both human and machine readable form, direct comparison of human versus machine searching of the base can be made if desired. The base comes

organized as a series of tabular arrays of data--values largely numerical in nature. Various political subdivisions comprise the objects in the data base. These objects are characterized by properties (some simple in nature such as 'population' and others more complex) and by values associated with these properties. This data base, therefore, consists of a series of object-property-value symbolic triples.

The principal part of the data base for the representative information file can be viewed as consisting of two large lists of information. Each list is broken down into 50 rectangular arrays or matrixes of information. The first list consists of a matrix for each state with rows for each city in each state. The second list also consists of a matrix for each state, but here the rows designate counties. Data file matrixes therefore have columns representing the names of properties and rows naming the political subdivisions; the intersection of a row and a column contains a value for a particular property of a particular object. The two principal lists of properties, the list of city properties and the list of county properties are approximately 130 and 160 items long, respectively. The data base is essentially static with respect to time.

b. Interpretive Information File

The interpretive information file contains a lexicon that has an entry for each term in the vocabulary of the source language, the language used for phrasing questions. These entries define the operational meanings of the words, the one or several meanings of a term which are significant in

the source language, and the referents for these terms in the information store. The lexicon, then, specifies a general relation which maps the terms of the source language into direct or indirect pointers to the representative information in the data file. A direct pointer is a pointer to an item or several items of information in the data file. An indirect pointer points to an expression in the target language (a procedure designed to carry out some part of the request implicit in a question).

Paths in the lexicon must be selected and combined according to semantic and grammatic constraints of the source language as reflected in a series of grammar and semantic rules. These rules, along with a third type of rule-- request construction rules--form the second principal part of the interpretive information file. These rules are selected and sequenced by the translation algorithm, and control the fetching and composition of grammatical and semantic information necessary to synthesize a procedure in the target language to retrieve the desired data. The translation algorithm should be general in the sense that it should work with any rules and lexical entries supplied to it in a suitable form. Specific usage of the translation algorithm is accomplished by supplying it with rules and a lexicon. For example, grammar rules particularize the algorithm to enable the parsing of input strings; semantic rules further select particular senses of terms in these strings; and request construction rules generate target language expressions representing the meaning of the input question.

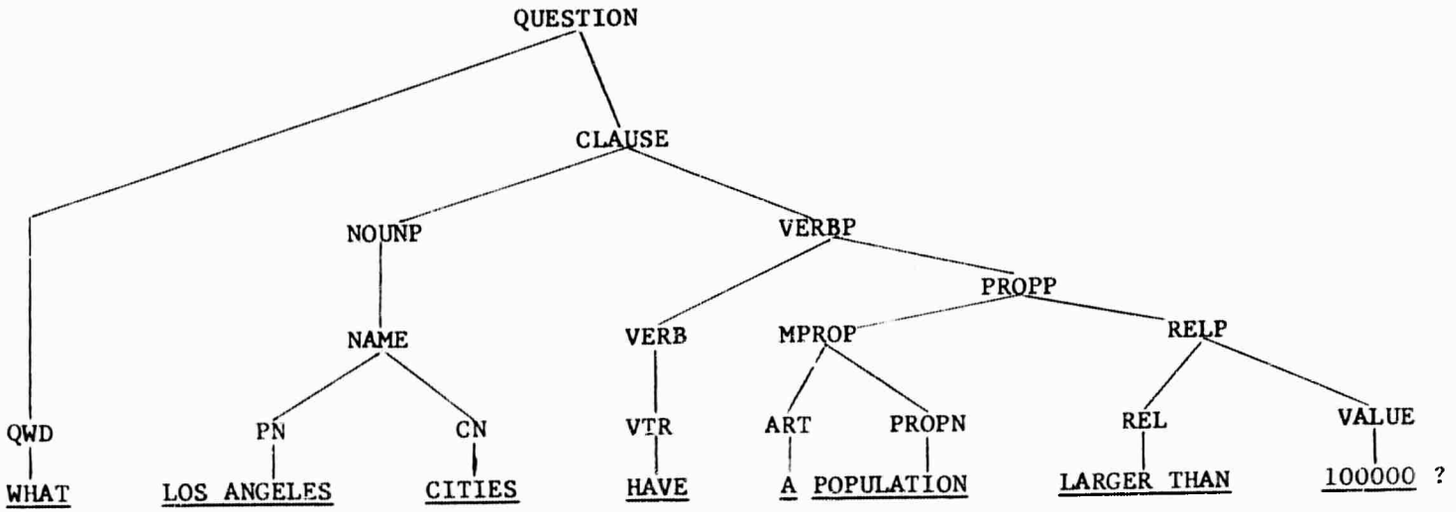
The interpretive information file is, of course, just another file to the time-sharing system, and it may be augmented or otherwise modified easily. Therefore, the rules and vocabulary of our framework for information storage and retrieval, and consequently the scope of the source language, the target language, or both may be modified to reflect changes in the content or structure of the data file, or to extend the capabilities of the source language.

c. Development of a Source Language

The structure and scope of the source language has been influenced by three primary factors: (1) the organization and content of the data base; (2) the characteristic features of existing query languages; and (3) the rudiments of English grammar. The scope of the source language might best be delineated as being of intermediate complexity between unconstrained ordinary English and a query language. The source language was constructed by starting with a grammar suitable for simple questions and then enlarging its scope by a step-by-step procedure. The source language may be construed as being either an improved query (information retrieval) language for use with suitably structured data base files, or as being a subset of natural language. In the first case, this may lead to practical schemes for data management systems. In the second case, our efforts may be viewed as a first step towards developing a natural language question/answering scheme. The grammar of the source language is designed to reflect many of the structure patterns of English which

relate to discourse concerning simple descriptions of objects. After the simple structural patterns are fairly well understood, more attention can be paid to the structure of English questions of a more complex nature.

Figure 2 presents a parsing or tree diagram for one of the first simple kinds of questions which were formulated in the source language. The question is: "What Los Angeles cities have a population larger than 100,000?" In this question the meaning of "Los Angeles" is ambiguous because "Los Angeles" is the name of both a city and a county. Therefore, the computer should in some way determine that "cities in the county of Los Angeles" is what is called for. If the question were phrased: "Do the cities of Los Angeles and San Francisco have ----------?" then "cities" would refer to the two cities of "Los Angeles" and "San Francisco." A series of context-free grammar rules to describe the syntax of the first question is given in Figure 2. Here the equals sign is used for the production (or rewrite) symbol, and the plus sign is used to indicate concatenation. The dollar sign is used as a sequence operator (i.e., 1$ followed by a term in parentheses indicates that a sequence of one or more occurrences of the term is permissible; a $1 followed by a term indicates that the term may or may not occur in the specified position). Quotation marks are used to surround actual terms in the source language to differentiate them from the names of syntactic categories. The slash '/' is used to indicate an alternative; e.g., a 'name' is composed of a sequence of one or more 'PN's' and/or 'CN's.' The parts of speech shown in Figure 2 represent, respectively: question words, proper names, common names,



QUESTION	= QWD + CLAUSE	VERB	= VTR	QWD	= 'WHAT'
CLAUSE	= NOUNP + VERBP	PROPP	= MPROP + RELP	PN	= 'LOS ANGELES'
NOUNP	= NAME	MPROP	= ART + PROPN	CN	= 'CITIES'
NAME	= 1 $ (PN/CN)	VALUE	= 1 $ (DIGIT)	VTR	= 'HAVE'
VERBP	= VERB + PROPP	RELP	= REL + VALUE	ART	= 'A'
				PROPN	= 'POPULATION'
				REL	= 'LARGER THAN'

Figure 2. An Example of a Tree Diagram and Syntax Rules for a Question

transitive verbs, articles, names of properties and relation names.

d. Development of a Target Language

The target language must carry out all legitimate searching, combining, and inferring operations which can be formulated within the scope of the source language. A suitable target language can therefore be expected to fetch any item of data and to combine sets of data values according to any of several criteria. No ideal language or set of file searching procedures exists today. Therefore, several forms of target languages were investigated.

e. Development of a Translation Procedure

The translation strategy should be able to deal effectively with the several one-many relationships which may occur during the processing of an English question. They may be described as follows: (a) A question is composed of many words. (b) A word may be assigned to several parts of speech. (c) A question may have several alternative parsings or structural descriptions (SD) indicating syntactic ambiguity. (d) Each word in an SD may have several senses or meanings (i.e., lexical readings) assigned to it. (e) Each SD may result in the production of several derived readings or semantic interpretations, indicating semantic ambiguity.

The translation process arrived at consists of three principal steps which convert: (a) a question to zero or more structural descriptions (SD's); (b) an SD to zero or more derived readings; and (c) a derived

reading to zero or more file searching procedures. These steps correspond to one stage of syntax analysis followed by two stages of synthesis. An outline of the framework for fact retrieval is illustrated in Figure 3. This figure shows the general information flow between the user, the question translator, and the data storage files. The first phase of question translation employs the set of grammar rules and lexical information to syntactically analyze a given question. The output from analysis might consist of a comment to the user to the effect that no SD was possible for his question because a lexical anomaly exists or because the grammatical structure of the question exceeded the bounds of the grammar of the system. In either event, as much pertinent information as possible is presented to the user so that he may rephrase his question within the vocabulary and/or grammatical pattern constraints. If one or more SD's are generated during this phase, they are processed, in turn, by the next phase of processing, semantic interpretation.

This first stage of synthesis comprises the successive amalgamation of derived readings from the individual lexical readings of the separate words in a question. If this does not result in a reading for the question, there is a semantic anomaly in the question. When this happens, the system will display the partly synthesized reading to the user in a format which will enable the user to determine if his question needs reformulation, if the lexicon requires updating, or both. If two or more derived readings for a given question result, a case of semantic

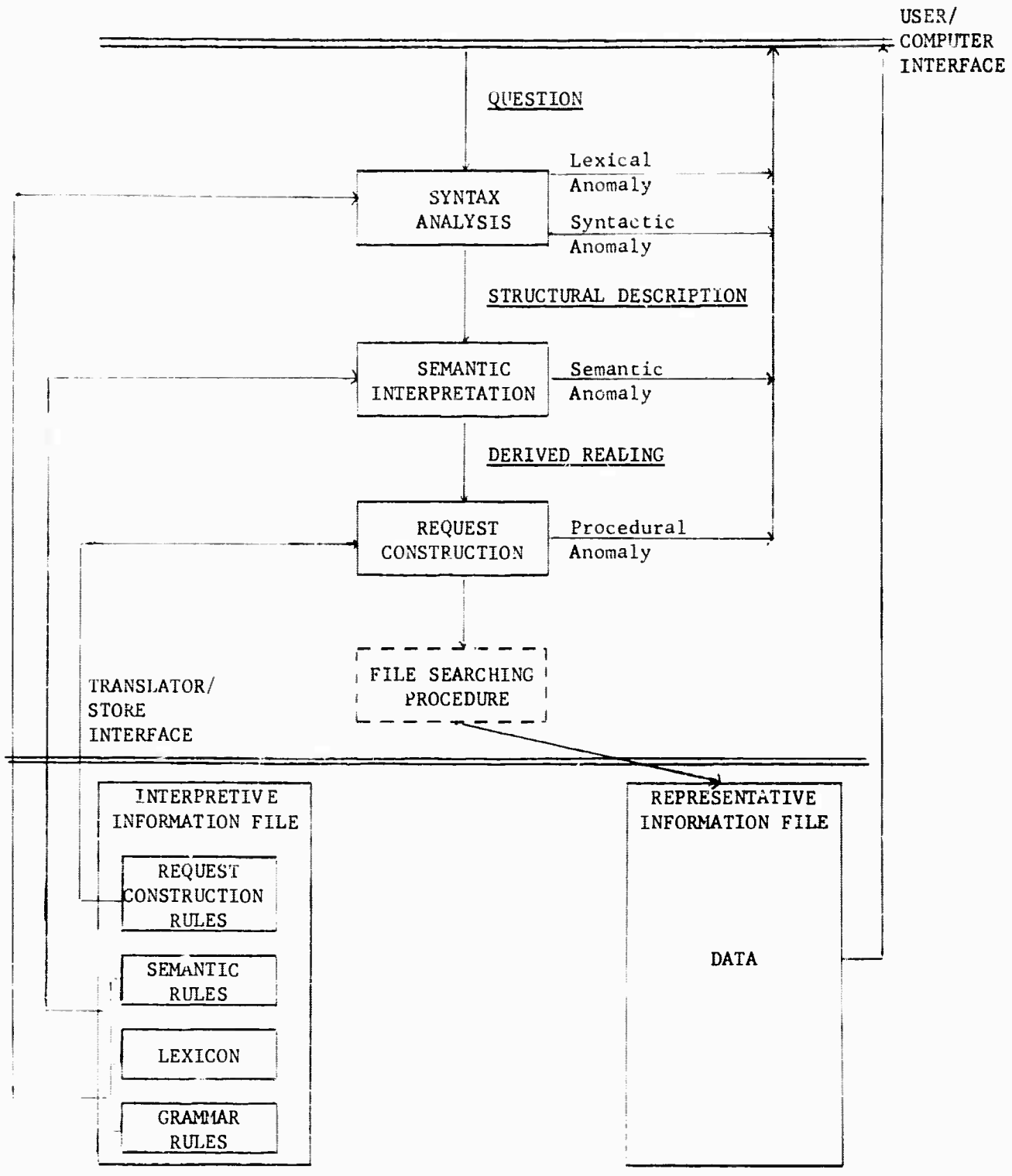


Figure 3. An Outline of the Framework for Fact Retrieval

ambiguity exists. These ambiguities may be resolved with respect to the data base during file searching. The second stage of synthesis provides for constructing a file searching procedure from a derived reading. The complexity of this process is highly dependent on the differences between the source language and the target language structures.

4. IMPLEMENTING THE FRAMEWORK

A specific implementation of the general approach to fact retrieval and the framework for question-to-request translation is currently being programmed. Some of the first steps towards this implementation are discussed in the next several sections. These steps include the parsing and interpretation of questions, the generation of file searching procedures, and the description of information in the data base.

Several preliminary programs have been constructed to perform syntax analysis and to translate questions into requests. These programs were written for the META5 Compiler Programming System (18). This compiler was used because it facilitated string manipulation and incorporated a syntactic analysis algorithm.

a. Recognizing Syntax

Two syntax recognition programs (parsing programs) have been implemented. The first program was capable of parsing and dividing questions into their grammatical constituents. The program accepts a small dictionary of words (terminal elements) and a set of syntax rules which spell out

the various relations admitted between nonterminal elements (syntax categories) and terminal elements. Figure 4 illustrates the parsing printout format. The META5 algorithm performs a left-to-right scan of the question, prints out the syntax category from the top to the bottom of the page, and, when applicable, prints out the terminal elements (compare Figure 4 with Figure 2). The program also has the capability (as shown in the second example: "How many cities have a TV station") of recognizing lexical or grammatical anomalies, i.e., words which are not in the vocabulary and questions which do not fall within the scope of the grammar or the vocabulary. In this example, "TV" and "station" are recognized as undefined symbols. Other terms in the question which are recognized, however, are printed out with their identified categories so that a user can isolate vocabulary terms or grammatical constructions which are not allowable. This feature is not only potentially useful to a user, but was of value in checking out the grammar.

The construction of syntax categories both at the part-of-speech level and at the phrase level was influenced by the structures of various query languages. From this starting point, work proceeded toward the acceptance of a larger number of typical English constructions. This approach is indicated by the parts of speech used. For example, the initial nominal category designations were "PN" for proper name, "CN" for common name and "PROPN" for property name as in the examples: "Los Angeles," "city," and "population," respectively. Similarly, the

```
WHAT LOS ANGELES CITIES HAVE A POPULATION LARGER THAN 100000 .
QWORD-WHAT
PN-LOS ANGELES
CN-CITIES
---------NOUNP
VTR-HAVE
ART-A
PROPN-POPULATION
---MODPROP
REL-LARGER THAN
VALUE-100000
---RELP
------PROPP
---------VERBP
------------CLAUSE
---------------QUESTION
END SYNTAX ANALYSIS

HOW MANY CITIES HAVE A TV STATION .
QWORD-HOW MANY
CN-CITIES
---------NOUNP
VTR-HAVE
ART-A
(TV) IS UNDEFINED
(STATION) IS UNDEFINED
END SYNTAX ANALYSIS
```

Figure 4. Output From the First Parsing Program

initial complement of verbs consisted only of the linking verb forms "is" and "are," and the transitive verb forms "has" and "have."

Since the information in the data base is essentially static with regard to time, only the present tense of verbs is considered, and the problems involved with handling verb tense did not arise. The English expressions for the relations of equality and inequality are also assigned to one part of speech category. The initial phrase marking capabilities enable recognition of noun, prepositional, and verb phrases, as well as property phrases--phrases in which properties are compared with other properties or with quantities given in a question. The ability to handle question words sueh as "what," "which," ard "how many," and simple clauses completes the initial recognition capability.

A second, more complex parsing program has been developed. This program is charaeterized by improved recognition, error detection, and printout formatting procedures.[1] The list of parts of speech has been considerably expanded. For example, the variety of prenominal modifiers has been expanded to include determiners (e.g., "all," "some," "none," "no two of the," "the third," "some one of the three," etc.). A number of adjectives were also introduced. Two additional nominal classes were added, "ON" (Operation Name) and "UN" (Unit Name). These classes permit reference to operators such as "percent," and "average," and units such as "tons,"

[1] Some of the grammar rules incorporated into this second syntactic recognition procedure are shown in Appendix A.

"acres," and "years." Much more extensive coordination of words and phrases is permitted at several levels in nominal and prepositional constructions. The class of verbs has been expanded, and a class of adverbs has been added. Adjectival and adverbial phrases are now incorporated, as well as dependent and independent clauses; these additions increase the variety of syntactic constructions available for formulating questions.

The printout format was changed to emphasize some kinds of structural information. In addition, the representation of constituents in a question was changed from the vertical list format shown in Figure 4 to that shown in Figure 5. Labeled brackets from the figure are listed below; these brackets identify the scope of constituents as follows:

Constituent Brackets

*Q(	)*Q	QUESTION	
*CL(	)CL*	CLAUSE	
*NP(	)NP*	NOUN	PHRASE
*VP(	)VP*	VERB	PHRASE
*PP(	)PP*	PREPOSITIONAL	PHRASE
*CP(	)CP*	COMPARE	PHRASE
*RP(	)RP*	RELATION	PHRASE
*AJ(	)AJ*	ADJECTIVE	PHRASE
*AD(	)AD*	ADVERBIAL	PHRASE

```
0000100-  001.
0000200-  WHAT LOS ANGELES CITIES HAVE A POPULATION
0000300-  LARGER THAN 100000  .
0000400-  002.
0000500-  WHAT IS THE POPULATION OF LOS ANGELES  .
0000600-  003.
0000700-  WHAT IS A CITY WITH A POPULATION OVER 100000 PEOPLE  .
0000800-  004.
0000900-  DOES THE CITY OF LOS ANGELES SPEND MORE FOR
0001000-  PUBLIC WELFARE THAN FOR EDUCATION  .
0001100-  005.
0001200-  HOW MANY RESTAURANTS AND BARS ARE THERE IN THE CITY
0001300-  OF LOS ANGELES  .
0001400-  006.
0001500-  WHAT EVIDENCE IS THERE TO SUGGEST THAT PARKINSONS
0001600-  LAW APPLIES TO THE POLITICAL SUBDIVISIONS OF THE
0001700-  UNITED STATES  .
0001800-  007.
0001900-  HOW MANY GOVERNMENT EMPLOYEES ARE THERE IN NEW YORK STATE  .
0002000-  008.
0002100-  HOW MANY PEOPLE ARE THERE IN CALIFORNIA  .
0002200-  009.
0002300-  WHAT STATES IN THE NORTHWEST REGION OF THE US WITH
0002400- A FARM LAND AREA LESS THAN 1000000 ACRES
0002500- HAVE A MEDIAN INCOME FOR FAMLIES IN EXCESS OF
0002600- 6000 DOLLARS  .
0002700-  010.
0002800-  WHICH ONE OF THE CITIES OF LA AND SF HAS THE LARGER
0002900-  POPULATION  .
0003000-  011.
0003100-  DOES EITHER LOS ANGELES OR SAN FRANCISCO OR CHICAGO
0003200-  HAVE A POPULATION DENSITY GREATER THAN THAT OF THE
0003300-  CITY OF NEW YORK  .
0003400-  012.
0003500-  DOES SOME ONE OF THE CITIES OF LA OR SF OR CHICAGO
0003600-  HAVE A POPULATION OVER THAT OF NEW YORK CITY  .
0003700-    .
0003800
READY-
```

Figure 5. Questions and Output From The Second Parsing Program
(Sheet 1 of 3)

```
001.
*Q(WHAT *CL(*NP(LOS ANGELES CITIES /M/ )NP* *VP(HAVE *CP(*NP(A POPULATIO
N /M/ )NP* *RP(LARGER THAN 100000/PM/ )RP* /PM/ )CP* /CM/ )VP* /PD/ )CL*
 /M/ )Q*
END SYNTAX ANALYSIS
002.
*Q(WHAT *VP(IS *NP(THE POPULATION /M/ *PP(OF LOS ANGELES /M/ )PP* /PM/ )
NP* /CM/ )VP* /PD/ )Q*
END SYNTAX ANALYSIS
003.
*Q(WHAT *VP(IS *NP(A CITY /M/ *PP(WITH A POPULATION /M/ *RP(OVER 100000*
NP(PEOPLE )NP* /M/ /PM/ )RP* /PM/ /M/ )PP* /PM/ )NP* /CM/ )VP* /PD/ )Q*
END SYNTAX ANALYSIS
004.
*Q(DOES *CL(*NP(THE CITY /M/ *PP(OF LOS ANGELES /M/ )PP* /PM/ )NP* *VP(S
PEND *AJ(MORE *PP(FOR PUBLIC WELFARE *RP(THAN *PP(FOR EDUCATION /M/ )PP*
 /PM/ )RP* /PM/ /M/ )PP* /PM/ )AJ* /CM/ )VP* /PD/ )CL* /M/ )Q*
END SYNTAX ANALYSIS
005.
*Q(HOW MANY *CL(*NP(RESTAURANTS (AND BARS /CR/ ) )NP* *VP(ARE *AV(THERE
*PP(IN THE CITY /M/ /M/ )PP* /M/ *PP(OF LOS ANGELES /M/ )PP* /M/ )AV* /C
M/ )VP* /PD/ )CL* /M/ )Q*
END SYNTAX ANALYSIS
006.
**ERROR DETECT MODE**
WHAT FWD
(EVIDENCE) IS UNDEFINED
IS   CWD
THERE   CWD
(TO) IS UNDEFINED
(SUGGEST) IS UNDEFINED
*NP(THAT )NP*  NOUNP
(PARKINSONS) IS UNDEFINED
(LAW) IS UNDEFINED
(APPLIES) IS UNDEFINED
(TO) IS UNDEFINED
THE FWD
(POLITICAL) IS UNDEFINED
(SUBDIVISIONS) IS UNDEFINED
*PP(OF THE UNITED STATES /M/ /M/ )PP*  PREPP
END SYNTAX ANALYSIS
007.
*Q(HOW MANY *CL(*NP(GOVERNMENT EMPLOYEES /M/ )NP* *VP(ARE *AV(THERE *PP(
IN NEW YORK STATE /M/ /M/ )PP* /M/ )AV* /CM/ )VP* /PD/ )CL* /M/ )Q*
END SYNTAX ANALYSIS
008.
*Q(HOW MANY *CL(*NP(PEOPLE )NP* *VP(ARE *AV(THERE *PP(IN CALIFORNIA /M/
)PP* /M/ )AV* /CM/ )VP* /PD/ )CL* /M/ )Q*
END SYNTAX ANALYSIS
```

Figure 5.
(Sheet 2 of 3)

```
009.
*Q(WHAT *CL(*NP(STATES *PP(IN THE NORTHWEST REGION /M/ /M/ /M/ )PP* /PM/
 *PP(OF THE US /M/ /M/ )PP* /PM/ *PP(WITH A FARM LAND /M/ AREA /M/ /M/ *
RP(LESS THAN 1000000*NP(ACRES )NP* /M/ /PM/ )RP* /PM/ /M/ )PP* /PM/ )NP*
 *VP(HAVE *NP(A MEDIAN INCOME/M/ /M/ *PP(FOR FAMLIES *RP(IN EXCESS OF 60
00*NP(DOLLARS )NP* /M/ /PM/ )RP* /PM/ /M/ )PP* /PM/ )NP* /CM/ )VP* /PD/
)CL* /M/ )Q*
END SYNTAX ANALYSIS
010.
*Q(WHICH *CL(*NP(ONE OF THE /M/ /M/ CITIES /M/ *PP(OF LA (AND SF /CR/ )
/M/ )PP* /PM/ )NP* *VP(HAS *NP(THE LARGER  POPULATION /M/ /M/ )NP* /CM/
)VP* /PD/ )CL* /M/ )Q*
END SYNTAX ANALYSIS
011.
*Q(DOES *CL(*NP(EITHER LOS ANGELES (OR SAN FRANCISCO /CR/ ) (OR CHICAGO
/CR/ ) )NP* *VP(HAVE *CP(*NP(A POPULATION DENSITY /M/ /M/ )NP* *RP(GREAT
ER THAN *NP(THAT *PP(OF THE CITY /M/ /M/ /PP* /PM/ *PP(OF NEW YORK /M/ )
PP* /PM/ )NP* /PM/ )RP* /PM/ )CP* /CM/ )VP* /PD/ )CL* /M/ )Q*
END SYNTAX ANALYSIS
012.
*Q(DOES *CL(*NP(SOME ONE OF THE /M/ /M/ /M/ CITIES /M/ *PP(OF LA (OR SF
/CR/ ) (OR CHICAGO /CR/ ) /M/ )PP* /PM/ )NP* *VP(HAVE *CP(*NP(A POPULATI
ON /M/ )NP* *RP(OVER *NP(THAT *PP(OF NEW YORK CITY /M/ /M/ )PP* /PM/ )NP
* /PM/ )RP* /PM/ )CP* /CM/ )VP* /PD/ )CL* /M/ )Q*
END SYNTAX ANALYSIS
```

Figure 5.
(Sheet 3 of 3)

The names of the parts of speech are eliminated from this printout since they are attached to the selected lexical entries. Other information has been incorporated in their place into the bracketed structure. This information indicates the kind of syntactic relation holding among the several constituents shown in a phrase, clause, or question.

Francis (9) discusses four basic kinds of syntactic structures: (1) modification structures composed of a modifier and a head; (2) coordination structures formed from equivalent syntactic units joined by conjunctions; (3) complementation structures consisting of a verbal element and its complement; and (4) structures of predication showing the relation of subject and predicate. The symbols "/M/," "/PM/," "/CR/," "/CM/," and "/PD/," stand for, respectively, premodification (large population), postmodification (city of Los Angeles), coordination, complementation, and predication. Explicit recognition of these syntactic relationships is helpful since the types of semantic rules which must be called upon to interpret the meaning of a question depend upon these syntactic relations.

One limitation built into the syntactic recognition algorithm in the META5 program is the lack of an ability for providing a multiple syntactic analysis of a question. Only the first parsing of a question that is encountered is produced by using this algorithm. Although this limitation has not posed any difficulty to date, as work progresses a multiple syntactic analysis capability will become required to a greater degree.

Figure 5 shows 12 examples of questions analyzed by this syntactic program. Currently, a file of several hundred questions of varying complexity is available for checking out syntactic recognition and semantic interpretation procedures. The syntactic recognition program will accept either a single question at a time or a file of questions such as those shown in Figure 5. These questions, with the exception of the sixth example, are typical of the questions which can be posed to the data base.

Question six demonstrates the program response to lexical and grammatical anomalies. In addition, the question is "unanswerable" by the techniques described in this paper because it is in such a highly nonprocedural form. Indeed, question-answering systems will have to be extremely well developed before questions of this kind can be intelligently interpreted by a computer. Questions of this kind must be reformulated by rephrasing them as one or more simpler information requests. Fortunately, human beings are quite adept at this process. For example, a user could employ his knowledge of Parkinson's Law and of the United States to formulate questions such as numbers seven and eight with respect to an "old" state (New York) and a "new" state (California). The computer could then apprise him of the fact that there are many more government employees per capita in New York than in California. Then the user could either draw his own conclusions or could formulate more questions concerning the applicability of Parkinson's Law.

b. Characterizing the Data

By far the most difficult of the three stages of question processing is the one devoted to the semantic interpretation of the question. The meaning of question terms in the contexts in which they occur must be unraveled and associated with the words used in the data base for descriptive purposes.

Some of the apparatus constructed by Travis to deal with the notions of "fact" and "fact assertion" will be employed to characterize the essential informational structure of the data base. Then constructs from the Katz/Fodor Semantic Metatheory will serve as a guide in constructing lexical entries and rules for synthesizing derived readings from lexical readings.

Figure 6 illustrates the structure of the Census Bureau Data Base as it applies to the political subdivisions of the United States. The domain of discourse comprises several categories of members of different types (e.g., Los Angeles, Northeast, Nevada); several kinds of predicates are involved in describing these members. Four categories of members (subdomains or sets) are involved: city objects, county objects, state objects, and region objects. Such a domain of discourse is referred to as multisorted. Clearly, one important aspect of characterizing the notion of political subdivision concerns the possible relations which may obtain between the members of the different sets comprising the several political subdivisions, such as the fact that the city of Los Angeles is located in a particular state (California) or a particular region (Southwest).

Subdomain 1: City

Applicable predicates

Type 1: Membership: (Los Angeles, New York,-----) is a member of city
Type 2: Containment: (each city is contained in some county)
Type 3: City properties: (population,-----)

Subdomain 2: County

Applicable predicates

Type 1: (Westchester, Orange) is a member of county
Type 2: (each county is contained in some state)
Type 4: County/State properties (farm land area)

Subdomain 3: State

Applicable predicates

Type 1: (New York, Nevada) is a member of state
Type 2: (each state contains some city)
Type 4: (population, farm land area)

Subdomain 4: Region

Applicable predicates

Type 1: (Northeast, Southwest) is a member of region
Type 2: (each region contains some state)

Figure 6. Political Subdivisions of the United States
(The Data Base Domain of Discourse)

The terms "is contained in" and "contains" will be adopted in this paper to designate these relations. For example, "Beverly Hills is contained in Los Angeles County," and "California contains Beverly Hills." Fact retrieval systems capable of handling relations of this kind have recently been described by Raphael and Elliot (19, 8). In symbolic logic, a formalization of the notion "every city is contained in some county" may be represented as: $\bigwedge x(F(x) \longrightarrow \bigvee y(G(y) \wedge H(x,y)))$, where F,G, and H stand for the predicates: "is a city," "is a county," "is contained in" respectively. Expressions of this type may be used to represent general facts. At present, this study does not require such explicit use of symbolic logic; therefore, these descriptions are represented in a less formal way. However, some means of handling such relations is crucial for a fact retrieval system to provide even a modest degree of inferential capability.

As a minimum for the Census Data Base, an inferential capability to find answers to questions like "Is New York City the largest city in population in the Northeast?" should be implemented. The fact that New York City is or is not in the Northeast region is never explicitly given in the data base. Therefore, relationships between the several political subdivisions must be used to deduce an answer. The relations among members of different sets are labeled as Type 2 predicates in Figure 6. Type 1 predicates specify membership, and Type 3 and 4 predicates specify the

properties[1] which associate member objects with a specific datum. The only difference between these latter two predicate types is the fact that they occur on different lists of properties. One list consists of approximately 130 properties applying to cities, and the other list consists of approximately 160 properties that apply to counties and states. Values of properties for regions are not stored in the data file since they may be readily computed. The two lists of properties partially overlap.

Church (6) says: "The intension of a concept consists of the qualities or properties which go to make up the concept. The extension of a concept consists of the things which fall under the concept." Here the concepts refer to the sets which represent the several kinds of political subdivisions of interest. Thus the extension of the term "city" is given by the list of its members. This list forms a part of the data in the representative information file.

Clearly the extension of a set is not sufficient by itself to characterize a set for purposes of fact retrieval. Some description of the intension of the set is needed. This is provided through the use of predicates to describe facts concerning particular members. The intension of sets is mediated through the interpretive information file where a key requirement is the association of the meanings of source language terms with the meanings of predicates.

---

[1] More precisely, many of these 'properties" are functors, see (21).

The meaning which is assigned to a predicate in the interpretive information file may constitute an expression of a general fact. These general facts, in association with the specific facts represented in the data file, (explicit facts), may then be used to derive implicit facts from the data file. The use of "explicit" and "general" facts to arrive at an "implicit" fact as an answer to a question will be illustrated later. The form of inference to be employed is termed "informal inference" in contradistinction to "formal inference," which employs the symbolism and inference rules of logic.

Figure 7 shows some of the column headings for cities as they are actually given in the Census Bureau published version of the Data Base (3). In an earlier effort aiming at computer retrieval of similarly described data (14, 15), a generic specific outline form was used to provide the kind of heading and descriptive information supplied in Figure 7. A heading like "Civilian Labor Force" for example, was called a major subject category; expressions such as "Durable Goods Manufacturing" or "nondurable Goods Manufacturing" were called "fact names." A coordinate index allowed a user to coordinate terms such as "goods" and "manufacturing" to arrive at the subset of all fact names containing these two terms. In addition, a thesaurus was employed to indicate the associations between terms and between terms and the subject headings. These two aids provided considerable help in formulating requests for information which in turn could be written in a restricted query language format for subsequent retrieval from the computer.

Items 301–316

AREA, POPULATION

Codes: State	Codes: SEA	Codes: State and county	City	Land area	Population, 1960: U S rank	Total	Per square mile	Increase or decrease (—), 1950 to 1960	Nonwhite: 1960	Nonwhite: 1950	Living in group quarters	Age: Under 5 years	Age: 21 years and over	Age: 65 years and over	Age: Median age	Nativity: Foreign born	Nativity: Native of foreign or mixed parentage	Married couples: Total	Married couples: Without own household
				Sq. mi.				Percent	Percent	Percent	Percent	Percent	Percent	Percent	Years	Percent	Percent		Percent
301	302	303	304	305	306	307	308	309	310	311	312	313	314	315	316				

Items 317–332

VITAL STATISTICS, INCOME, EDUCATION, MIGRATION

City	Vital statistics: Live births, 1960	Vital statistics: Deaths, 1959	Number of families, 1960	Income in 1959 of families, 1960: Median income	Income in 1959 of families, 1960: Under $3,000	Income in 1959 of families, 1960: $10,000 and over	Aggregate income in 1959 of the population, 1960	Education, 1960: Population, 25 years old and over: Median school years completed	Education, 1960: Population, 25 years old and over: Completed less than 5 years of school	Education, 1960: Population, 25 years old and over: Completed high school or more	Education, 1960: Population, 25 years old and over: College graduates	Education, 1960: School enrollment, persons 5 to 34 years old: Kindergarten and elementary school	Education, 1960: School enrollment, persons 5 to 34 years old: High school	Education, 1960: School enrollment, persons 5 to 34 years old: College	Migration, 1960: Residents in same house [1]	Migration, 1960: Migrants from different county [1]
				Dol.	Percent	Percent	Mil. dol.	Years	Percent	Percent	Percent				Percent	Percent
	317	318	319	320	321	322	323	324	325	326	327	328	329	330	331	332

Items 333–348

LABOR FORCE

City	Civilian labor force, 1960: Total	Unemployed	Male	Employed persons: Total	Construction	Manufacturing: Durable goods	Manufacturing: Nondurable goods	Transportation	Communication and other public utilities	Wholesale and retail trade	Finance, insurance, and real estate	Educational services	Public administration	White-collar occupations [1]	Worked outside county of residence [1]	Using public transportation [1]
		Percent	Percent											Percent	Percent	Percent
	333	334	335	336	337	338	339	340	341	342	343	344	345	346	347	348

Items 349–364

HOUSING

City	Dwelling units, 1950	Housing units, 1960: Total	Median number of rooms per unit	In one-unit structures	In structures built in 1950 or later	Sound, with all plumbing facilities	Occupied units: Total	Population per unit	With 1.01 or more persons per room	Moved into unit during 1958 to 1960	Owner occupied: Total	Owner occupied: Median value	Renter occupied: Total	Renter occupied: Median gross rent per month	Vacant units, year round available: Total	Vacant units, year round available: For rent
				Percent	Percent	Percent			Percent	Percent		Dol.		Dol.		Percent
	349	350	351	352	353	354	355	356	357	358	359	360	361	362	363	364

Items 365–381

HOUSING, MANUFACTURES

City	Housing units, 1960—Con.: Occupied units with— Clothes washing machine	Home food freezer	Air conditioning	TV set	Telephone	Automobiles: One	Automobiles: Two or more	Manufactures, 1958: Establishments: Total	Establishments: With 20 or more employees	All employees: Number	All employees: Payroll	Production workers: Number	Production workers: Wages	Value added by manufacture, adjusted	Capital expenditures, new	Manufactures, 1954: All employees	Manufactures, 1954: Value added by manufacture, unadjusted
	Percent	Percent	Percent	Percent	Percent	Percent	Percent				$1,000		$1,000	$1,000	$1,000		$1,000
	365	366	367	368	369	370	371	372	373	374	375	376	377	378	379	380	381

Figure 7. Column Headings for Cities

Something similar to the intersection logic of a coordinate index and the association capability of a thesaurus would seem essential for converting questions into requests. The definitional, selectional, and combinational apparatus described by Katz and Fodor (12) is highly suggestive in this regard, and has influenced our work. However, we consider semantic interpretation in a very specific context, i.e., the computer translation of questions into file searching procedures, and do not wish to claim that we are attempting to follow their theory faithfully. We will employ some of their terminology such as semantic markers and lexical reading, but these terms will take on somewhat different meanings with reference to our specific problem. In particular, our main use of semantic markers will be to interrelate the meanings of question terms and request terms and not to define the meanings of these terms separately. In addition to the notion of word sense, we shall introduce and employ the further notions of concept and relation with respect to lexical readings.

Figure 8 illustrates the lexical structure that has been arrived at in order to make accessible the information required to effect question to request translation. The structure takes the form of a labeled, directed, acyclic, nonplanar graph, i.e., it is more complex than a tree since more than one branch may enter a particular node, yet no paths between branches and nodes can amount to a cycle.

Each path starts at the origin and ends at a terminal node (TN), not shown in Figure 8. The first node following the origin on each path specifies

(Page 44 Blank)

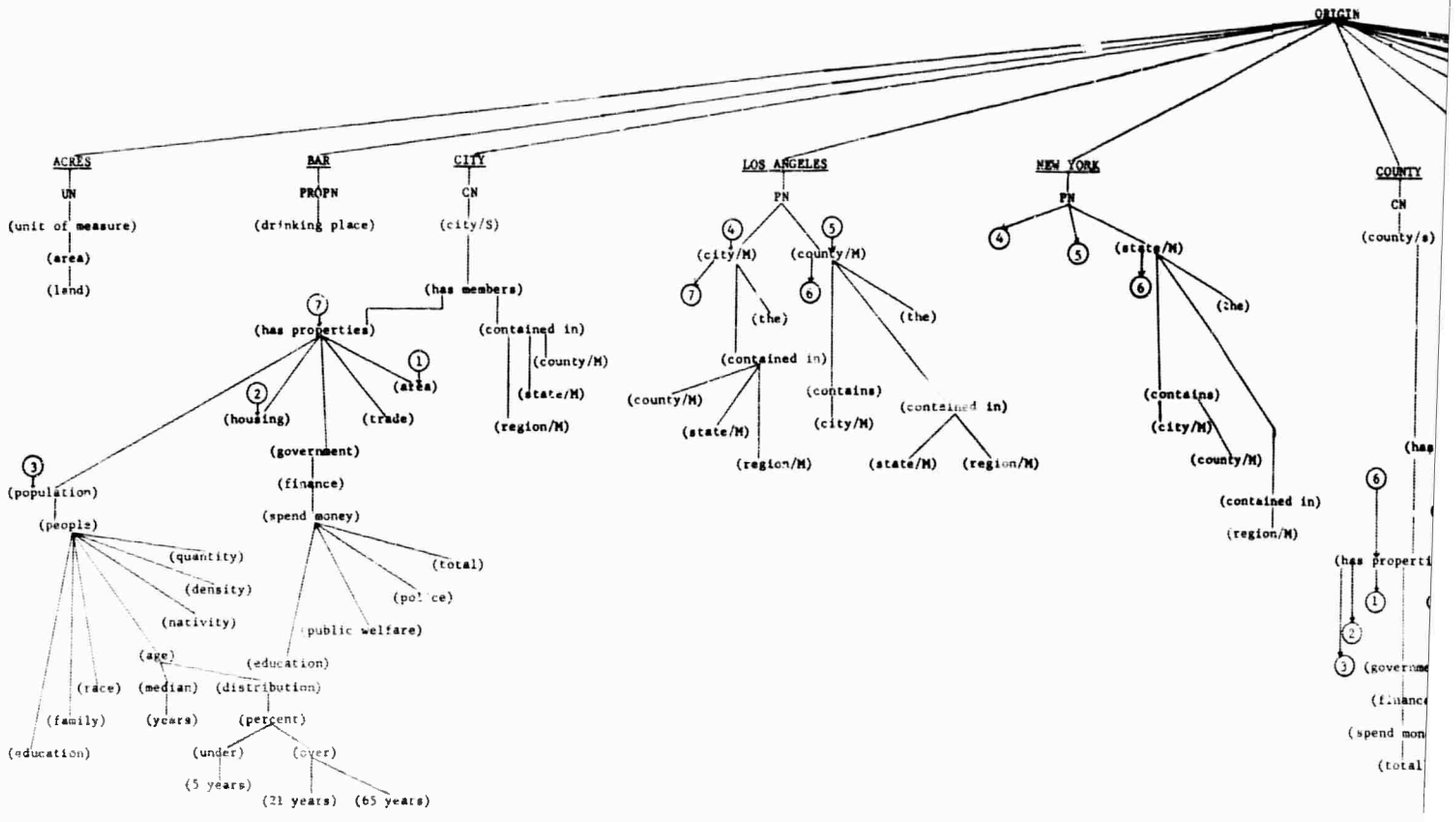
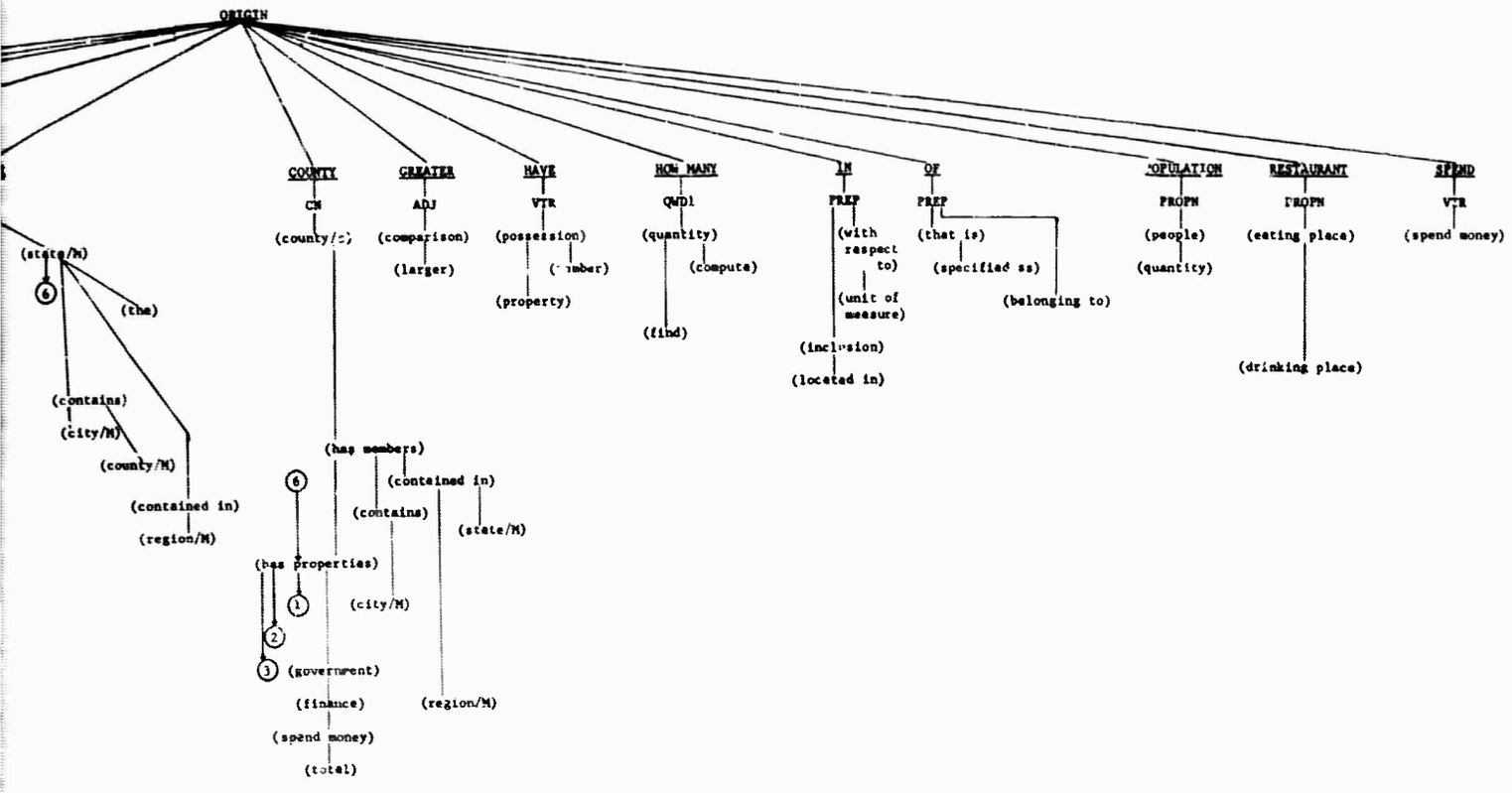


Figure 8. Lexical Items and Markers for a Semiotic Network

./000/00


ORIGIN
COUNTY
CN
(county/c)
GREATER
ADJ
(comparison)
(larger)
HAVE
VTR
(possession)
(property)
HOW MANY
QWD1
(quantity)
(find)
(compute)
IN
PREP
(with respect to)
(unit of measure)
(inclusion)
(located in)
OF
PREP
(that is)
(specified as)
(belonging to)
POPULATION
PROPN
(people)
(quantity)
RESTAURANT
PROPN
(eating place)
(drinking place)
SPEND
VTR
(spend money)
(state/M)
6
(the)
(contains)
(city/M)
(county/M)
(contained in)
(region/M)
(has members)
(contained in)
(contains)
(state/M)
(city/M)
6
(has properties)
1
2
3
(government)
(finance)
(spend money)
(total)
(region/M)

a lexical item; a term in the source language vocabulary. The subpaths subordinate to a lexical item specify the lexical entry for that item. Each subpath constitutes a lexical reading and consists of a series of at least one grammatical marker (GM), and one or more semantic markers (SM) before the terminal node is reached. The markers and terminal nodes used in the lexicon reflect grammatic, semantic, and pragmatic features of lexical items. These features correlate with the three branches of semiotic that deal with relations between signs, between signs and their designata, and between signs and their interpreters. Here "interpreter" is taken to mean the computer and its adjuncts (i.e., software) rather than a person. Therefore, the lexicon may be construed as representing a semiotic network. Information from this network is used in each of the several stages of question answering.

The semiotic network can be visualized in logical terms as follows: Consider each labeled node as representing a predicate applicable to the object designated by the associated lexical item. Let each branch between nodes be interpreted as a logical "and" (i.e., as a conjunct). Then each lexical reading may be viewed as a proposition composed of the conjunction of several elementary propositions formed by predicate-object associations. If, in addition, the association between two branches eminating from the same parent node is taken as the logical "or" (a disjunct), the entire network is the equivalent of a logical expression in disjunctive normal form. In logic, any conceivable

expression in the propositional calculus is equivalent to some expression in disjunctive normal form. Therefore, the semiotic network admits the same degree of combinatorial complexity it is present in the calculus of propositions. A lexical reading in the network can be considered as "true" when conditions are suitable for its selection and incorporation into a derived reading.

Semantic markers are constructed and assigned to lexical items only as required to make the discrimination of meaning that is essential for answering questions from the data base. Only those meanings of words potentially relevant to the domain of discourse are marked. This effort determines the kind and quantity of semantic markers which are required for the question-answering task. Once this is done, we can consider the more general problem of the transferability of lexical entries from one domain of discourse to another.

Each terminal node specifies a referent. A referent is a target language subexpression which represents the meaning of a lexical reading. A terminal node may also specify a selection restriction to be applied in the process of selecting lexical or derived readings. A lexical reading, by virtue of the semantic markers associated with it, can represent a word sense, concept, or relationship. A lexical reading specifies a sense if: (1) it contains a grammatic marker other than 'PN' or 'CN'; or (2) it contains the marker 'PN' or 'CN' and is not a concept or relationship. Concepts and relationships apply only to members of the

PN and CN word classes. They specify the meaning associations between lexical items, properties (predicate types 3 and 4), and relations (predicate type 2), respectively. Once the senses of lexical items in a particular question have been selected, the remaining senses cannot be accessed. Conceptual and relational information, on the other hand, can be accessed as often as required. These three forms of semantic representation allow considerable flexibility in prescribing lexical entries.[1]

c. Synthesizing Interpretations

The following five examples illustrate some of the steps required in synthesizing derived readings from a pair of constituents and their lexical readings:

1. New York New York
2. New York City
3. City of New York
4. City in New York
5. The population of New York City

In the Census Data Base, "New York" refers to a state, a county within that state (New York County contains only the borough of Manhattan) and a city (composed of five boroughs, each in a separate county).

Example 1--"New York, New York"--is subject to the following interpretations:

. "New York" City in "New York" state.

[1]Appendix B presents a listing of semantic markers for many of the column headings shown in Figure 7.

. "New York" City in "New York" county.

. "New York" County in "New York" state.

The semantic rule associated with this example treats the first occurrence of "New York" as a modifier of the second occurrence--the head. This rule applies the selection restriction component of each lexical reading of the modifier to the markers associated with the head. Since both constituents are proper names, only interset relations can be selected by this process. Thus "New York" as a city will have attached to it selection restrictions to find paths with the two markers "(contains)" and "(City/M)." The two paths of the second occurrence of "New York" that meet these conditions will yield the first two interpretations given above. The third interpretation is reached by a similar process.

The derived reading, therefore, consists of the phrase "New York, New York" followed by the grammatic marker "NAME" and the paths:

"(City/M)———(has: properties)———(contained in: state/M);"

"(City/M)———(has: properties)———(contained in: county/M);"

"(County/M———(has: properties)———(contained in: state/M)."

The second interpretation is a consequence of the assumption that every city is contained in some county. While it is logically possible for New York County to contain the City of New York, this is not the case. This fact will be discovered only when the data base is scanned. Viewing the first two interpretations as "questions," the first is answered by

"yes," and the second by "no." More precisely, this situation presents an instance where both the explicit fact: "New York City is contained in New York County," and the more general fact: "New York City is contained in some county," have the same datum--"false."[1] The third interpretation (New York County in New York State) is correct according to the data base, and therefore, the phrase "New York, New York" is ambiguous.

In a similar fashion, the second example given at the beginning of this section--"New York City"--is subject to the following three possible interpretations:[2]

- the City of New York
- a city in the County of New York
- a city in the State of New York

The second and third interpretations could be ruled out if desired. However, occurrences of the second and third interpretations in contexts such as "Does some New York city have...?" can be distinguished from the first interpretation in a context like "What is the population of New York City?"

---

[1] New York City is not "in" any county according to (3).

[2] We must ignore the important clue given by the capitalization of "City" since our on-line typewriters at present use only upper case alphabetic characters.

In this example, the selection restrictions attached to paths of the proper name are used to select paths from the common name. The semantic rules required for the first two examples are similar in form to projection rule R1 in (12). However, instead of combining distinguishers as is done in projection rules, semantic rules cause request construction rules to be called upon to effect the generation of an appropriate expression in the target language.

These two examples illustrate the association of meanings of proper names through the "contained in" relation and the meaning of a proper name modifying a common name. Reversal of these associations leads to semantic anomalies--the generation of no derived reading--in both cases. For example, consider "California Los Angeles." This phrase will not receive a reading because the selection restrictions associated with paths having the semantic marker "(contains)" specify that only paths with a "set" marker, i.e., "(City/S)" can be selected. Therefore, "California City" will receive an interpretation, but "California Los Angeles" will not. Similarly, "City California" will not be acceptable although "Los Angeles City California" will be accepted because the selection restrictions for the phrase "Los Angeles City" are compatible with a path for the term "California."

In example (3)--City of New York--"city" is postmodified by the prepositional phrase "of New York." The over-all phrase "City of New York" has only one interpretation--to specify the city "New York." One of the

meanings of "of"--meaning "that is"--has attached to it the selection restriction: (indicated by angle brackets) "⟨(the)⟩." Therefore, three paths are selected to represent "the City of New York," "the County of New York," and "the State of New York." When the phrase "of New York" is amalgamated with the head, "city," the second and third paths are rejected.

A different situation obtains in example 4--"city in New York." Here the only legitimate interpretation is: some city in the State or County of New York. One sense of the preposition "in"--the sense "located in" with the selection restriction "⟨contains⟩"--selects the paths in "New York" as a State and as a County "containing" counties and cities. This modifying phrase is used in turn to select a path from the term "city." The resulting interpretations are: (1) a city "in" New York County, and (2) a city "in" New York State.

These four examples of amalgamation supply us with a total of nine possible interpretations: three for examples (1) and (2), two for example (4) and one for example (3). The amalgamation process differs somewhat, for example (5)--"the population of New York City." In this case the sense of the term "population" corresponds to a concept associated with members of the city set, and a single derived reading specifying the population of the City of New York will result.

There is an important distinction between the selection of sense and concepts for derived readings. Once a subset of senses for a lexical or derived reading is chosen, further access to the nonselected senses is not permitted. This is not the case for concepts. Recurrent access to a list of concepts may be essential. For instance, the interpretation of question 4 in Figure 5 required access to the concepts associated with "city" twice--once for "Public Welfare" and once for "Education." In general, concepts are only accessed during the amalgamation process when a meaning of one term overlaps the meanings of a concept associated with another term.

These examples indicate some of the requirements which semantic rules fulfill. In general, semantic rules map the paths corresponding to two sets of lexical or derived readings into zero or more paths that provide a semantic interpretation for the lexical string (sequence of words) associated with the two original sets of readings. The path selection process is a function of the conditions spelled out in selection restrictions, while the marker sequences constituting paths in derived readings are dependent upon conditions specified in semantic rules. Each derived reading forming a part of a semantic interpretation consists of (a) the lexical string (e.g., "New York, New York"), (b) the dominating grammatical marker (e.g., "NAME"), and (c) some function of the semantic markers and referents from the selected paths. A derived reading then can represent the following:

- the amalgamated sense of a lexical string,
- the specification of a particular object, e.g., "the city = New York,"
- the determination of interset relations, e.g., "Northeast contains New York,"
- the association of an object with a property.

The general observation that the grammar (or meaning) of a sentence is a function of the grammar (or meaning) of its parts can be replaced within the scope of the present framework with the following statements:

- The markers associated with a lexical string are a function of the markers associated with its constituent lexical strings.
- The derived readings assigned to a lexical string are a function of the lexical and derived readings assigned to its constituent lexical strings.

Now the questions presented in Figure 5 can be interpreted and discussed. Belnap (2) considers that a question presents "a series of alternatives"[1] from which a subset may be selected to form a "direct answer" to the question. This subset is selected in accord with the kind of "request" that the question poses. The types of requests and alternatives can be used to categorize questions. Belnap specifically discusses three types of requests--"unique alternative," "complete list," and "non-exclusive"--and two ways of presenting alternatives--"fill in the blank questions,"

[1] Terms in quotation marks are those used with special meaning in (2).

and "multiple choice" questions.

Six categories can be derived from the several combinations of the above-mentioned requests and alternatives: The questions from Figure 5 are repeated with their original numbers underneath the appropriate categories.

Category 1:

Fill in the blank/unique alternative questions.

2. What is the population of Los Angeles?

5. How many restaurants and bars are there in the City of Los Angeles?

7. How many government employees are there in New York State?

8. How many people are there in California?

These questions, discussed below, may all be rephrased to the form "the population of Los Angeles is _____." They delimit alternatives by specifying a condition for their choice and a set of names. In the above questions the conditions vary, but the set of names is the same. Each question is answered by the name for some number--a numeral. Further, each of these questions is answered by exactly one (unique) alternative.

Question 2 has a straightforward translation into a request for the "datum" associated with the predicate "population" for the objects denoted by "Los Angeles" (a city and a county). The ambiguity of "Los Angeles" leads us to consider Question 2 as two separate questions of the same type with different conditions for the selection of an answer.

The computer response to Question 2 is the output of fact assertions answering these implicit requests, e.g.,

Population (Los Angeles City) is 2,479,015

Population (Los Angeles County) is 6,038,771

Question 5 is interesting because it illustrates both the limitations and the advantages of the question answering technique proposed in this paper. Nowhere in the data file do we actually have a count of the number of "restaurants" and the number of "bars" in a city. What we do have is data associated with the predicate: "number of eating and drinking places." This data undoubtedly includes restaurants and bars. However, it may also include other entities: "cocktail lounges," "cafeterias," etc. Therefore, a precise answer to the question cannot be found in the data base.

Lacking the precise answer, a relevant answer is desired. Since "restaurants" and "bars" are connected by a coordinating conjunction, it is possible to combine the senses of these terms if: (a) they share identical semantic markers, and (b) they point to the same referent. We have already indicated that such an overlap of senses exists in Figure 8. "Restaurant" is marked with the semantic markers (eating place) and (drinking place) and "bar" is marked with the marker (drinking place).

The steps in interpreting question 5 (indicated by numbers in parenthesis) may be outlined as follows: (1) The phrase "the City of Los Angeles" is interpreted as in the previous discussion of New York City. (2) In the phrase "in the City of Los Angeles" the semantic marker "(located in)" is added to the interpretation. (3) In the phrase "there in the City of Los Angeles," the adverb signifies "(located in)" and does not add to the length of the interpretation. (4) In the expression: "are there in the City of Los Angeles" the verb "are" has three senses, including the equational sense as in "what are cities" and two forms of an associative sense. In one associative sense, the verb "are" provides a link between "city" in the verb phrase and the referent of the subject of the sentence. A semantic rule must pick the correct sense of "are" and must indicate the type of subject which may be associated with this type of verb phrase, namely, a subject which references one or more properties. (5) In the expression "restaurants and bars," both terms being coordinated specify properties, have a semantic marker in common, and point to the same predicate; therefore, coordination is permitted. (6) The noun and verb phrases are amalgamated according to a rule which matches the subject argument type and the verb phrase argument type and finds them compatible, i.e., the property specified in the subject of the question applies (with the appropriate associative sense of "are") to "Los Angeles." (7) In this final step one sense of "how many" is selected which is compatible with the previously selected concept signifying eating and drinking places. Since the number of such places is directly available as a

value and hence need not be computed, the '(find)' '(quantity)' sense is selected. (This sense of "how many" indicates that the quantity desired already exists as a value. No tabulation is necessary, as would be required in questions such as "how many cities have a larger population than that of Beverly Hills?" In this second case the actual tabulation of the number of cities meeting the condition in the comparative phrase would have to be carried out.) The answer for this question will be a fact assertion of the form: 'Number of eating and drinking places (Los Angeles City) is 5,551.'

Fact assertions as answers in these two examples should probably be obligatory because they supply an essential context for determining the relevance of an answer. For some questions, a simple "yes," "no," or numerical response may be sufficient. In general, fact assertions should be optional or obligatory answers depending on the question type and the questioner's needs.

Questions 7 and 8 utilize the same interpretation of "how many" as Question 5. No calculation is called for. Question 8 could also be expressed as 'What is the population of California?" since the phrases "how many people" and "the population" would lead to the selection of the same predicate.

Category 2:

<u>Fill in the blank/complete list questions</u>.

1. What Los Angeles cities have a population larger than 000?

9. What states in the northwest region of the United States with a farm land area of less than 100,000 acres have a median income for families in excess of 6,000 dollars?

These questions specify as answers a listing of all alternatives (in this case the names of cities and states) that meet the request. Question 1 must receive the interpretation "cities in the county of Los Angeles" since the interpretation "Los Angeles (as a city) Cities" is semantically anomalous. Question 9 is similar to Question 1 except for its greater length due to several qualifying prepositional phrases.

Category 3:

Fill in the blank/non-exclusive question.

3. What is a city with a population over 100,000 people?

A non-exclusive question differs from a unique alternative question in its request for an exemplar or typical answer from the file. In Question 3, the article "a" expresses this exemplar condition. The determiners "any" and "some" have similar interpretations.

Category 4:

Multiple choice/unique alternative questions.

4. Does the City of Los Angeles spend more for public welfare than for education?

10. Which one of the cities of Los Angeles and San Francisco has the larger population?

These questions specify four alternatives in accord with Belnap's analysis:

- The City of Los Angeles spends more for public welfare than for education.
- The City of Los Angeles does not spend more for public welfare than for education.
- Los Angeles has the larger population.
- San Francisco has the larger population.

Question 4 illustrates how a verb in the source language can be related semantically to a predicate applying to the data file. The verb "spend" is marked as referring to the expenditure of money. Figure 8 also shows that the concepts associated with cities have a path through "government," "finances," and "spend money" to "public welfare," and "education." The comparative phrase which forms a part of the verb phrase including "spend" leads to merging the links to "public welfare" and "education" with the sense of "spend" and to the two predicates: "expenditures for public welfare in the city government" and "expenditures for education in the city government."

Multiple choice questions are distinguished from fill-in-the-blank question types by their explicit listing of alternatives or explicit asking of a "yes" or "no" question that can be interpreted as representing two explicit alternatives. Question 10 defines a subset of the city set

to be searched for the member with the larger "datum" associated with the predicate "population."

Category 5:

Multiple choice/complete list questions.

11. Does Los Angeles, or San Francisco, or Chicago have a population density greater than that of the City of New York?

Category 6:

Multiple choice/non-exclusive questions.

12. Does some one of the cities of Los Angeles or San Francisco or Chicago have a population over that of New York City?

Because of the similarities between these categories, they are discussed together. Question 11 is a multiple choice/complete list question since the alternatives are directly specified in the question and the "or"'s are to be interpreted in the inclusive sense. An answer to this question will therefore list the names of all specified cities that meet the conditions of the request. Question 12 is subject to a similar interpretation with the exception that at most one city will meet the specified conditions. Both questions pose a set of six alternatives of the form:

(a) "Los Angeles has a $\left\{\begin{matrix}\text{population density}\\ \text{population}\end{matrix}\right\}$ greater than that of New York City.

(b) "Los Angeles doesn't have a $\left\{\begin{matrix}\text{population density}\\ \text{population}\end{matrix}\right\}$ greater than that of New York City.

(c) "San Francisco has a {population density / population} greater than that of New York City.

(d) "San Francisco doesn't have a {population density / population} greater than that of New York City.

(e) "Chicago has a {population density / population} greater than that of New York City.

(f) "Chicago doesn't have a {population density / population} greater than that of New York City.

Question 11 can also be considered as comprising three unique alternative questions: i.e., questions with alternatives (a)(b), (c)(d), and (e)(f). Therefore, this question is of a complex type--a combination of multiple choice unique alternative and complete list questions and is answered either by "no" or by a list of cities.

Similarly, question 12 overlaps two of the six question categories since it is both yes/no ("Does") and non-exclusive ("some one of the cities").

Ten of the sample questions appear to be easily classifiable according to four of Belnap's question types. The last two questions indicate a need to extend the analysis to cover more complex questions. Further work on the logic of questions can be expected to shed additional light on question answering as a process of interest in its own right, above and beyond its intimate association with grammar and semantics. It may

yield a set of categories that will be useful in testing an SFQA system's question-answering power.

d. Generating File Searching Procedures

Two target languages were studied with respect to their suitability for specifying file searching procedures. One of these languages, QUUP, is the source language for the LUCID Data Management System (7). The census data is available for use in the LUCID system in such a form that it can be searched and manipulated by means of QUUP expressions formulated either directly by a user or indirectly as a result of computer translation of a question into a LUCID query.

A comparison of natural language questions and their corresponding paraphrases in QUUP language should reveal some of the linguistic devices employed in natural and artificial languages for expressing the content and structure of retrieval requests. Such a comparison will also enable contrasting user performance in phrasing requests in both languages.

As interesting and useful as these comparisons may be, this form of question processing is inherently inefficient. The second form of target language being studied comprises a file searching language at a lower, more machine-oriented level. This target language will ultimately be composed of direct calls on subroutines which will form a part of

TDMS (Time-Shared Data Management System).[1] Directly synthesizing file searching instructions or a file searching program will allow circumvention of many of the steps involved in translating from a question to QUUP and then to a file searching program. In order to study the feasibility of this kind of question translation, a target language was constructed that was largely based on the procedures for accessing information stored in matrix form developed by Iverson (11), and Walker and Bartlett (22). Grammars for a subset of both target languages are illustrated in Figure 9. The file searching language requires the data to be organized into matrixes composed of "object" rows and "property" or "item" columns where a value is located at the intersection of a row and column. A matrix is required for each grouping of cities by state, of counties by state, and of states by region. Matrix operations (e.g., "SELECT," "PSRCHEQ"), when combined with the operands M, I, and VV, form Instruction I_1 to Instruction I_4. Instruction I_1 is a selection operation which selects a column of values from a matrix. Instructions I_2 and I_3 specify searching operations which select those values in a column equal to or greater than specified values. Instruction I_4 simply prints out the list of found values.

Instructions I_1, I_2, and I_3 reduce a matrix to only those columns, values, and rows that meet the specified conditions. These reduced matrixes are referred to in succeeding instructions by using one or more asterisks,

[1] This system is being implemented with ARPA support on the IBM S/360 Computer.

```
(a)  (Matrix) M = 'M--CI'/'M--CO'/'MREG'/PREV

(Identifier) I = 'CINM'/'CONM'/'STNM'/'STCO'/'POP'

(Value) V = 'LA'/'CAL'/1$(DIGIT)

        X = V/PREV

        PREV = 1$('*')

        I1 = 'SELECT('+M+I+')'

        I2 = 'PSRCHEQ('+M+I+X+')'

        I3 = 'PSRCHGR('+M+I+X+')'

        I4 = 'PRINT('+ PREV + ')'

        REQUEST = 1$(I1/I2/I3) + I4

(b) (Relations) RL = 'EQ'/'GR'

(Connectives) CN = 'OR'/"AND'

(Comparative) CP = I + RL + V

QUUP QUERY = 'PRINT'+I+'WHERE'+CP+$(CN+CP)
```

Figure 9. Partial Specifications for Two Target Languages

depending upon whether the reduced matrix was generated by the prior instruction, the second prior instruction, etc.

In the first rule in Figure 9, for example, PREV refers to a previously specified reduced matrix. "MREG," "M--CI," and "M--CO" refer to vectors of matrices (e.g., four state-by-region matrices, 50 city-by-state matrices, and 50 county-by-state matrices). The matrix of California City information would be specified as:

PSRCHEQ (M--CI, STNM, CAL)

A request for any city in the country with a population greater than 26,000 would be formulated:

PSRCHGR (M--CI, POP, 26000)

Instructions call for the selection of a specific matrix, for the selection of a particular column in that matrix, and for those values which are equal to or greater than a specified value. A request is simply a sequence of one or more type I_2, I_2, or I_3 instructions followed by one I_4 instruction.

Queries in the subset of the LUCID language are formed (Figure 9) from: (1) the same set of identifiers and values; (2) the relations "EQ" and "GR" or the connectives "OR" and "AND"; and (3) the comparative (an identifier relation-value combination). A query in QUUP comprises the word "print" followed by a combination of an identifier, the word "where," a comparative, and a sequence of zero or more connectives and comparatives.

A more extensive set of instructions and LUCID query terms than those shown in Figure 9 were implemented in the META5 program. This program provided an experimental capability for translating from English questions into both the matrix operation language and the LUCID language. In addition, this rudimentary capability was implemented to learn more about the precise nature of the tasks involved in writing a translator.

The six questions which are shown in Figure 10 (and their parallel translations into the two target languages) have one thing in common--they are all paraphrases of each other in the sense that each will result in the selection of the same answer. However, Questions 1, 2, and 3 are under-specified, Question 4 is completely specified, and Questions 5 and 6 are over-specified. Question 4 is completely specified since it references a particular state (California) in a particular county (Los Angeles), and the cities in this county. This formulation of the question should require minimal processing and search time. Question 2 translates into requests and queries which are identical with those of Question 1. The differences between these two questions is simply the explicit occurrence of "county" in Question 2 versus its elliptical expression in Question 1. Questions 3 and 4 both specify the State of California and have identical translations. Again, the difference is one of an explicit versus an elliptical use of the term "county." Questions 5 and 6 result in the selection of the correct answers from the file even though they are "overl--" or "redundantly" specified.

```
 1. WHAT LOS ANGELES CITIES HAVE A POPULATION LARGER THAN 100000  .
1.
                                        PRINT
                                        CINM WHERE
                                        CONM EQ
                                        LA
PSRCHEQ(M--CO,CONM,LA)
SELECT(*,STCO)
PSRCHEQ(M--CI,STCO,*)
                                        AND
PSRCHGR(*,POP,100000)
                                        POPGR100000
SELECT(*,CINM)
PRINT(*)
TRANSLATION COMPLETED

 2. WHAT LOS ANGELES COUNTY CITIES HAVE A POPULATION LARGER THAN

 100000  .
2.
                                        PRINT
                                        CINM WHERE
                                        CONM EQ
                                        LA
PSRCHEQ(M--CO,CONM,LA)
SELECT(*,STCO)
PSRCHEQ(M--CI,STCO,*)
                                        AND
PSRCHGR(*,POP,100000)
                                        POPGR100000
SELECT(*,CINM)
PRINT(*)
TRANSLATION COMPLETED

 3. WHAT LA CAL CITIES HAVE A POP LARGER THAN 100000  .
3.
                                        PRINT
                                        CINM WHERE
                                        CONM EQ
                                        LA
PSRCHEQ(M--CO,STNM,CAL)
PSRCHEQ(*,CONM,LA)
SELECT(*,STCO)
                                        AND
                                        STNM EQ
                                        CAL
PSRCHEQ(M--CI,STNM,CAL)
PSRCHEQ(*,STCO,**)
                                        AND
PSRCHGR(*,POP,100000)
                                        POPGR100000
SELECT(*,CINM)
PRINT(*)
TRANSLATION COMPLETED
```

Figure 10. Examples of Question Translation
(Sheet 1 of 3)

```
 4. WHAT LA COUNTY CAL CITIES HAVE A POP LARGER THAN 100000  .
4.
                                        PRINT
                                        CINM WHERE
                                        CONM EQ
                                        LA
PSRCHEQ(M--CO,STNM,CAL)
PSRCHEQ(*,CONM,LA)
SELECT(*,STCO)
                                        AND
                                        STNM EQ
                                        CAL
PSRCHEQ(M--CI,STNM,CAL)
PSRCHEQ(*,STCO,**)
                                        AND
PSRCHGR(*,POP,100000)
                                        POPGR100000
SELECT(*,CINM)
PRINT(*)
TRANSLATION COMPLETED
```

```
 5. WHAT WESTERN REGION LOS ANGELES CITIES HAVE A POPULATION

 LARGER THAN 100000  .
5.
                                        PRINT
                                        CINM WHERE
PSRCHEQ(MREG,RENM,WESTERN)
SELECT(*,STNM)
                                        RENM EQ
                                        WESTERN
PSRCHEQ(M--CO,STNM,*)
PSRCHEQ(*,CONM,LA)
                                        AND
                                        CONM EQ
                                        LA
SELECT(*,STCO)
PSRCHEQ(M--CI,STCO,*)
                                        AND
PSRCHGR(*,POP,100000)
                                        POPGR100000
SELECT(*,CINM)
PRINT(*)
TRANSLATION COMPLETED
```

Figure 10.
(Sheet 2 of 3)

```
 6. WHAT WESTERN REGION LOS ANGELES COUNTY CALIFORNIA CITIES

HAVE A POPULATION LARGER THAN 100000  .
6.
                                                PRINT
                                                CINM WHERE

PSRCHEQ(MREG,RENM,WESTERN)
SELECT(*,STNM)
                                                RENM EQ
                                                WESTERN

PSRCHEQ(M--CO,STNM,CAL)
PSRCHEQ(*,CONM,LA)
                                                AND
                                                STNM EQ
                                                CAL
                                                AND
                                                CONM EQ
                                                LA

SELECT(*,STCO)
PSRCHEQ(M--CI,STNM,CAL)
PSRCHEQ(*,STCO,**)
                                                AND

PSRCHGR(*,POP,100000)
                                                POPGR100000

SELECT(*,CINM)
PRINT(*)
TRANSLATION COMPLETED
```

Figure 10.
(Sheet 3 of 3)

Both questions specify a region of the United States as well as a particular county, and then refer to cities in that county. In Question 6, where the state, county, and region are all specified, the redundancy is more apparent. The sequence of file searching instructions for this question leads to the following selections:

(a) the names of all states in the West

(b) the California county matrix

(c) Los Angeles county

(d) the state and county code number for the county of Los Angeles

(e) the California city matrix

(f) the cities of California which have the state and county code number given in (d), e.g., the cities of Los Angeles, Long Beach, Beverly Hills, etc.

(g) those cities selected in (f) with a population over 100,000

(h) the names of the cities selected in (g).

A very elementary but basic mechanism for effecting informal inference[1] is evident here. Sequences of instruction, which in parameterized form represent general facts such as "every region contains some state," and "every state contains some county," lead to searches which establish the truth of explicit facts such as "West contains California," "California

[1]The determination of a relevant predicate through the association of semantic markers in the semiotic network may be considered as another form of informal inference. (See Example 5 in Figure 5.)

contains Los Angeles County." In turn, the implicit fact "West contains Los Angeles County cities" is determined.

The form of request construction rules, in contradistinction to semantic rules, is similar to the form of the rewrite rules used to specify the grammar of source and target language expressions. Basically, request construction utilizes a generative grammar to synthesize particular expressions subject to the constraints imposed by the preceeding two phrases of question processing.

5. SUMMARY AND CONCLUSIONS

This approach to question answering and fact retrieval is an attempt to steer a feasible path between small-scale, experimental, natural language systems and large-scale, operational, data management systems. Consideration of theoretical and practical factors has led to a framework for translating English questions into data management system file searching procedures. Three basic factors in the framework are (1) the subset of English available for phrasing questions (SL), (2) the intermediate form for search procedures (TL), and (3) the structure and content of the data base (DB).

The practical utility of a system based on this framework will be dependent upon efficient implementation of a particular SL/TL/DB triple, and upon its consequent effectiveness in answering user questions. Generality will be primarily dependent upon the scope of the English subset admissible as an SL, the richness of the TL, and the ability to handle data bases of varying subject content.

Initial results with data structures on the level of complexity found in the Census data are most encouraging. Present restrictions have to do with limitations to context-free rewriting rules, to the lack of a multiple path syntactic analysis procedure, and to explicit, stored facts that can be characterized as object/property/value and object/relation/object triples. The theories used in constructing this framework illustrate how both linguistic and logical constructs may be fruitfully integrated into a fact-retrieval system. The methods for derivation of structural descriptions and semantic interpretations for questions stem from work by linguists on generative grammars and semantic theory. At this level the complex problems of anomaly, ambiguity, and paraphrase must be dealt with. The constructs employed are: constituent, marker, sense, concept, lexical reading, and derived reading. A transition to logical constructs such as set, object, predicate, fact, and the logical connectives at the request construction level proved useful and natural for the description and characterization of structured data and associated file searching languages. Work on the logic of questions and the logic of facts is helpful in characterizing the request contained in a question and in relating it to structured information and associated retrieval operations. An appeal to logic will become even more essential as more complex forms of inference are pursued. It appears that the designer of fact retrieval systems is faced, not with a choice between the tools of linquistics and the tools of logic, but rather with the problem of employing and using constructs from both fields in an effective manner.

The Census data base reveals both relatively simple structural and rather complex semantic features. The structure comprises simple combinations of a small number of components. Several sets and several hundred predicates enable characterization of the fact functions which have the explicit facts composed of objects, properties, relations, and values as members.

Semantic complexity is reflected both by the fact that different objects may share the same name; and more importantly, by the fact that the meaning behind many of the predicates is complex. This complexity leads to the establishment of many relevant associations between predicates and English terms whose meaning they share or overlap. These features help to account for the fact that although an English question and its corresponding representation in target language form may appear to have a straightforward structure, the translation from one form to the other may be quite complex.

The initial question-to-request translation program demonstrates several significant aspects of question processing. Among these are: the detection of anomalous terms and questions, the disambiguation of word meaning through the use of contextual clues, and the means for dealing with simple forms of paraphrase and ellipsis.

Another feature demonstrated by this program--an essential for any fact retrieval program--is a means for effecting inference. Two kinds of inference (referred to as "informal inference") may occur. The first kind is effected while synthesizing derived readings from lexical readings. This is

illustrated by Questions 4 and 5 in Figure 5, where predicates referencing "government expenditures" and "eating and drinking places" are determined to be relevant to questions concerning "Los Angeles spending" and "restaurants and bars."

The second kind of informal inference occurs as a byproduct of data file searching. It differs from formal deduction only in the lack of use of explicit inference rules and the conventional symbolism of logic. Implicit facts are derived from a series of explicit and general facts.

Upon completion of the programming presently in process, the census data question-answering facility will be available for detailed exploration of the power, feasibility, and potential utility of this approach. A critical question concerns the required scope and complexity in order for the English microgrammar to be more useful for a community of users than existing query languages. At present, initial evidence suggests that many questions can be more conveniently and succinctly phrased in the English subset than in a query language. On the other hand, we do not yet know how many restrictions in available syntax patterns and vocabulary a user will be willing to endure, provided that he is informed of the nature of unacceptable questions and can then reformulate his request into simpler terms.

Once a viable system is achieved for the initial data base, the question of the transferability of the framework to other data bases of similar structure and the extension of the framework toward more complex capabilities can be attacked. One measure of effectiveness for the framework is its suitability

for being generalized or extended. There are several possible extensions of the framework. The scope of the source language may be widened either by incorporating results from ongoing projects in generative phrase structure grammars or by converting the source language to a more adequate model of grammar, such as the transformational grammar being developed at the Mitre Corporation (24). Employment of this type of grammar could greatly extend the grammatical capabilities of the system, and would provide a more satisfactory basis upon which to pursue semantic interpretation (see references 5 and 13).

Fact storage, retrieval, and deductive inference capabilities could be significantly increased by enriching the target language to include the predicate calculus. By so doing, general facts could be expressed in this form, associated with appropriate paths in the semiotic network, and utilized in conjunction with a decision procedure and explicit facts to effect deduction in a more rigorous manner. The structure of explicit facts can be extended along the lines prescribed by Travis (21) to include reference to time and to second-order information such as indicators of source and reliability.

BLANK PAGE

Bibliography

(1) Bar Hillel, Y., Language and Information. Addison - Wesley Publishing Company, 1964.

(2) Belnap, N. D. Jr., An Analysis of questions, TM-1287. System Development Corporation, Santa Monica, California, June 1963.

(3) Bureau of the Census, County and City Data Book 1962. U. S. Department of Commerce, Washington, D. C., 1962.

(4) Chomsky, N., Syntactic Structures. Mouton and Co. - S - Gravenhage, 1957.

(5) Chomsky, N., Aspects of the Theory of Syntax. Cambridge: The M.I.T. Press, 1965.

(6) Church, A., Definition of "Intension" and "Extension," in D. D. Runes, etc., Dictionary of Philosophy. Littlefield, Adams and Co., 1961, pp 147, 148.

(7) Cozier, W. A. and Dennis, W. C., QUUP Users Manual, TM-2711/000/01. System Development Corporation, Santa Monica, California, February 1966.

(8) Elliot, R. W., A Model for a Fact Retrieval System, TNN 42. University of Texas Computation Center, May 1965.

(9) Francis, W. N., The Structure of American English. Ronald Press Co., New York, 1958, Chapter 6.

(10) Giuliano, V. E., Comments (on reference (20)), Comm. of ACM. January 1965, pp 69-70.

(11) Iverson, K., A Programming Language. New York: John Wiley & Sons, 1962.

(12) Katz, J. and Fodor, J., The structure of a semantic theory, Language. No. 39, No. 2, April - June 1963, pp 170-210.

(13) Katz, J. and Postal, P., An Integrated Theory of Linguistic Descriptions. Cambridge: M.I.T. Press, 1964.

(14) Kellogg, C., The fact compiler, a system for the extraction, storage and retrieval of information, Proc. of the AFIPS Western Joint Computer Conf. Spring 1960.

(15) Kellogg, C., et al, ALERT-1 Reference Series (four volumes). Ramo Wooldridge Division of Thompson Ramo Wooldridge, Canoga Park, California 1961.

(16) Kellogg, C., Designing artificial languages for information storage and retrieval in Automated Language Processing. H. Borko, ed., John Wiley and Sons, in press.

(17) Oettinger, A. G., Automatic processing of natural and formal languages in: Proceedings of IFIP Congress 1965. Washington, D. C., Spartan Books, Vol. 1, pp 9-16.

(18) Oppenheim, D. K., The META5 language and system, TM-2396. System Development Corporation, Santa Monica, California, July 1965.

(19) Raphael, B., SIR: a computer program for semantic information retrieval, AFIPS Fall Joint Computer Conf. 1964.

(20) Simmons, R., Answering English Questions by Computer: A survey, Comm. of ACM. January 1965, pp 49-68.

(21) Travis, L., Analytic information retrieval. In Natural Language and the Computer. P. Garvin, ed., McGraw Hill, New York, 1963, pp 310-353.

(22) Walker, D., and Bartlett, J., The structure of language for man and Computer: problems in formalization, First Congress on the Information Sciences. November 1962.

(23) Watt, W., A prerequisite on the utility of microgrammars, Technical Note 258. National Bureau of Standards, Washington, D. C., April 1965.

(24) Zwicky, A. M., Friedman, J., Hall, B. C., Walker, D. E., The Mitre syntactic analysis procedure for transformational grammars, AFIPS Proceedings of the Fall Joint Computer Conference. Spartan Books, 1965.

APPENDIX A: A PARTIAL GRAMMAR

1. Q = QWD1 + ICL
2. Q = QWD1 + PP + VP
3. Q = QWD1 + NP + DCL + VP
4. Q = QWD2 + VP
5. Q = QWD2 + VP + DCL
6. Q = VERB + QN + COMPP
7. Q = VERB + QN + NP
8. Q = VERB + QN + PP
9. Q = PP + VP
10. Q = ICL
11. NP = CORD1 + NP1 + 1$(CORD2 + NP1) + NP1
12. NP = CORD1 + NP1 + 1$(CORD2 + NP1)
13. NP = NP1
14. NP1 = QN + $(PP)
15. NP = PREL + $(PP)
16. NP1 = QN + PART + QN
17. NP1 = PREL + PART + QN
18. QN = CORD1 + QN1 + 1$(CORD2 + QN1) + QN1
19. QN = CORD1 + QN1 + 1$(CORD2 + QN1)
20. QN = QN1
21. QN1 = DET + ADJ + NM
22. QN1 = DET + NM
23. QN1 = ADJ + NM
24. QN1 = NM
25. NM = CORD1 + NM2 + 1$(NM2) + NM1
26. NM = CORD1 + NM2 + 1$(NM2)
27. NM = NM1
28. NM1 = 1$(NM2)
29. NM2 = CN
30. NM2 = PN

```
31.  NM2 = PROPN
32.  NM2 = UN
33.  NM2 = ON
34.  VP = VERB + COMPP
35.  VP = VERB + NP
36.  VP = VERB + PP
37.  VP = VERB + ADJP
38.  VERB = VLNK
39.  VERB = VTR
40.  PP = CORD1 + PP1 + 1$(CORD2 + PP1)
41.  PP = PP2
42.  PP1 = PREP + QN + RELP
43.  PP1 = PREP + QN
44.  PP2 = PREP + CORD1 + QN1 + RELP + 1$(CORD2 + QN1 + RELP)
45.  PP2 = PREP + CORD1 + QN1 + RELP + 1$(CORD2 + QN1)
46.  PP2 = PP1
47.  COMPP = CORD1 + COMPP1 + $(CORD 2 + COMPP1)
48.  COMPP1 = NP + RELP
49.  COMPP1 = RELP
50.  RELP = REL + NP
51.  RELP = REL + PP
52.  RELP = REL + VALUE
53.  RELP = REL + VALUE + NP
54.  ADJ = ADJS
55.  ADJ = ADJC
56.  ADJP = ART + ADJ + PREP + QN
57.  ADJP = ADJ + PREP + QN
58.  ADVP = ADV + COMPP
59.  ADVP = ADV + NP
60.  ADVP = ADV + PP
61.  DET = DETS + 'OF THE' + DETN
62.  DET = ART + DETI'
```

(Page 82 Blank)

```
63.  DET = DETS
64.  DET = 'NO' + DETS
65.  DET = 'NO' + DETS + 'OF THE'
66.  DET = DETS
67.  DET = ART
68.  ICL = NP + VP
69.  DCL = CORD1 + PREL + VP + $(CORD2 + PREL + VP)
70.  QWD1 = 'WHAT'---
71.  QWD2 = 'WHERE'---
72.  DETS = 'EVERY'---
73.  DETN = 'ONE'---
74.  ART = 'THE'---
75.  PREL = 'THAT'--
76.  PREP = 'OF'--
77.  PART = 'HAVING'--
78.  CORD1 = 'EITHER'--
79.  CORD2 = 'OR'---
80.  REL = 'EXCEED'--
81.  NEG = 'NOT'
82.  VALUE = 1$(DIGIT)
83.  PN = 'CHICAGO'--
84.  CN = 'CITY'--
85.  ON = 'PER ENT'--
86.  UN = 'DOLLARS'--
87.  PROPN = 'POPULATION'--
88.  VTR = 'HAS'--
89.  VLNK = 'IS'--
90.  ADJS = 'LARGEST'--
91.  ADJC = 'LARGER'--
92.  ADV = 'ALSO'--
```

APPENDIX B: EXAMPLES OF SEMANTIC MARKERS FOR LEXICAL READINGS

Node Number	Semantic Marker	Successor Nodes
1.	City/S	2
2.	members	3,6
3.	contained in	4,5
4.	county/M	T1
5.	state/M	T2
6.	properties	7,10,48,54,63
7.	area	8
8.	land	9
9.	square miles	T3
10.	population	11
11.	people	12,14,16,24,34,42,45
12.	quantity	13
13.	total	T4
14.	density	15
15.	per square mile	T5
16.	age	17,19
17.	median	18
18.	years	T6
19.	distribution	20
20.	percent	21,22,23
21.	≦ 5 years	T7
22.	≧ 21 years	T8
23.	≦ 65 years	T9
24.	family	25,27
25.	quantity	26
26.	total	T10
27.	income	28,30
28.	median	29

Node Number	Semantic Marker	Successor Nodes
29.	dollars	T11
30.	distribution	31
31.	percent	32,33
32.	<3000 dollars	T12
33.	>10000 dollars	T13
34.	education	35
35.	people of age	36
36.	≥ 25 years	37
37.	school years completed	38,39
38.	median	T14
39.	percent	40,41
40.	< 5 years	T15
41.	≥ 16 years	T16
42.	nativity	43
43.	percent	44
44.	foreign born	T17
45.	race	46
46.	percent	47
47.	non white	T18
48.	housing units	49
49.	percent	50
50.	with	51,52
51.	TV set	T19
52.	automobile	53
53.	≥ 2	T20
54.	trade	55
55.	retail	56
56.	sales	57,60
57.	gas	58

Node Number	Semantic Marker	Successor Nodes
58.	service	59
59.	station	T21
60.	eating	61
61.	drinking	62
62.	places	T22
63.	government	64
64.	finances	65
65.	spend money	66,67,68,69
66.	total	T23
67.	police	T24
68.	public welfare	T25
69.	education	T26
70.	county/S	71,73,75
71.	members contained in	72
72.	state/M	T27
73.	members contain	74
74.	city/M	T28
75.	properties	7,10,48,76,80,84
76.	government	77
77.	finance	78
78.	spend money	79
79.	total	T29
80.	industry	81
81.	mineral	82
82.	establishments	83
83.	quantity	T30
84.	agriculture	85,89,92
85.	farm	86
86.	land	87
87.	acres	88

Node Number	Semantic Marker	Successor Nodes
88.	in thousands	T31
89.	farms	90
90.	quantity	91
91.	< 10 acres	T32
92.	commercial	93
93.	fertilizer	94
94.	used	95
95.	in tons	T33

Unclassified

Security Classification

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)

System Development Corporation
Santa Monica, California

2a. REPORT SECURITY CLASSIFICATION

Unclassified

2b. GROUP

3. REPORT TITLE

An Approach to the On-Line Interrogation of Structured Files of Facts Using Natural Languages.

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

5. AUTHOR(S) (Last name, first name, initial)

Kellogg, Charles H.

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
29 April 1966	86	24

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
AF 19 (628), ARPA Order 773	SP-2431/000/00
b. PROJECT NO	
c.	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d	

10 AVAILABILITY/LIMITATION NOTICES

This document has been cleared for open publication and may be disseminated by the Clearing House for Federal Scientific & Technical Information.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY

13. ABSTRACT Two principal objectives provide the main focus for development of an on-line capability for fact retrieval: (1) formulation of a conceptual framework within which certain issues and problems of fact retrieval may be viewed and clarified; and (2) achievement of a practical and useful fact retrieval capability in the reasonably near future. A balance should be obtained between these two partially interactive, partially conflicting objectives. Recent insights and work in the theories of linguistic description and in logical explication have aided in pursuit of the first objective. Although much of this theory is tentative in nature, a core of concepts now exists which can be brought to bear on the problem of computer retrieval of facts. The second objective was approached by selecting a large existing collection of descriptive information as a 'data base' and then, as much as possible, using experimental data management procedures that have been developed for the SDC/ARPA-TSS. The conceptual framework was shaped and given substance by introducing and using current conceptions concerning descriptions of the grammar and semantics of natural languages and formalizations of the notions of question and of fact. This information structure is linked to English Question terms through an associative lexical structure--a semiotic network--which reflects grammatic, semantic, and pragmatic features of lexical items. The principal dichotomy in the framework is between an algorithm for processing questions and the information store. The algorithm can be specialized to deal with varying subsets of English terms, their syntactic patterns, and their associated semantic interpretations. The information store consists of an interpretive information file and a representative information file.

DD FORM 1 JAN 64 1473

Unclassified

Security Classification

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
On-Line Interrogation Structured Files Fact Retrieval Natural Language						

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2a. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2b. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year; or month, year. If more than one date appears on the report, use date of publication.

7a. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7b. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8a. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8b, 8c, & 8d. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9a. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9b. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through

_____."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through

_____."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through

_____."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS)*, *(S)*, *(C)*, or *(U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, rules, and weights is optional.