QE-III: A FORMAL APPROACH TO NATURAL LANGUAGE QUERING

James Clifford
Graduate School of Business Administration
New York University

ABSTRACT

In this paper we present an overview of QE-III, a language designed for natural-language querying of historical databases. QE-III is defined formally with a Montague Grammar, extended to provide an interpretation for questions and temporal reference. Moreover, in addition to the traditional syntactic and semantic components, a formal pragmatic interpretation for the sentences of QE-III is also defined.

I. INTRODUCTION

Numerous systems for natural language database access have been described in the literature, including [Woods 1972], [Waltz 1976], [Harris 1978], and [Hendrix 1978]. While these systems are dissimilar in a number of different respects, they all share what to us is the same defect, namely the lack of any fundamental formal theory of the semantics of the database or of the English query language.

We view the development of those and other such systems as belonging to the first phase in the development of a formal theory of database semantics and of database querying, much as the early years in the design of computer languages such as FORTRAN were part of the first phase in the development of a theory of programming language semantics. The birth of programming language theory awaited the impact of formal language theory and a theory of syntax-directed translation. An analogous development in the area of natural language querying would require the impact of formal language theory and a theory that coupled the syntax and the semantics of English. Many linguists today believe that Montague's theory of universal grammar [Montague 1970b] is the first successful attempt at formalizing such a uniform syntactic and semantic theory of natural language. We believe that some such formal theory of a query language is an important first step towards the development of provably correct and reliable natural language processing systems. For inherent in the notion of program "correctness" is the concept of a standard against which a program is to be judged.

In [Clifford 1982] we provided a formal definition of the query fragment QE-III as a Montague Grammar. QE-III simplifies the semantic theory of the language presented in [Montague 1973] (known as PTQ), and offers a natural correspondence to the semantics of queries in a database context. It fragment is provided with a formal syntax, semantics, and pragmatics, each component designed with the database application in mind. Among the major extensions to the PTQ fragment embodied in QE-III are the inclusion of time-denoting expressions and temporal operators, an analysis of verb meanings into primitive meaning units derived from the database schema, the inclusion of certain forms of direct questions, and the inclusion of a formal pragmatic component. These extensions, and the interpretation with which they are provided, are motivated by the goal of database access, but they are equally interesting in their own right. The syntactic theory presented is in some cases admittedly naive, for we have been primarily interested in getting the interpretation right. Recent work (e.g. [Gazdar 1981]) indicates that broad syntactic coverage can be coupled with a formal semantics.

This paper provides a brief introduction to the work presented in [Clifford 1982], where a small query fragment (QE-III) is rigorously provided with a complete semiotic theory: syntax, semantics, and pragmatics.

II. HISTORICAL DATABASES

In [Clifford & Warren 1983] we showed that a formal semantics can be given to the concept of an historical relational database. The semantics
given was analogous to the semantics of
the relational database model viewed as
an applied first order theory; extending the relational model to an
historical database prompted a move to
the higher-order language IL-s [Clifford
1982], (with its built-in concept of
denotation with respect to an index) in
order to provide a formal semantics for
such databases in a natural way.

Briefly, each "ordinary" relation
was extended with a special attribute,
"STATE," which served to index the facts
recorded by tuples in the relations.
The values for this attribute, drawn
from a domain of times, effectively
time-stamped each tuple in a manner
analogous to the notion of "denotation
with respect to an index" which is the
principle underlying the "possible world
semantics" of formalized intensional
logic (see Dowty [1981]).

With the notion of a historical
database comes the burden of providing
an interpretation for queries and
commands that make reference (explicit
or implicit) to the notion of time. QE-III
was designed to provide such an
interpretation for such database
queries. The interpretation of queries
expressed in English is defined formally
in terms of the formal semantics of the
HDBM. The correlation between the HDBM
semantics and this query language is
made explicit by interpreting the query
fragment via an indirect translation
into the same intensional logic IL-s
that was used to formalize the HDBM.
Through these translations, the model
for IL-s that "corresponds" to a
particular HDB (in a sense formalized in
[Clifford & Warren 1983] also serves as
the model for a formal definition of the
model-theoretic interpretation of the
English queries. In addition to
providing a semantic interpretation,
which in model-theoretic terms is called
its denotation, we also provide for each
expression a pragmatic interpretation in
a manner to be explained.

III. CRITERIA FOR THE THEORY

In developing the theory of QE-III
we were guided by two basic principles.
First was that the interpretation or
"meaning" of a natural language database
query be as close as possible to the
interpretation of database queries in,
say, the relational algebra or calculus.
This meant that the interpretation of a
query should somehow encompass its
answer as represented in the underlying
database. Second was the issue of
computational tractability. This meant
taking into account what was known about

IV. OVERVIEW OF THE LANGUAGE QE-III

A. Individual Concepts vs. Entities

Most recent research in the field
of Montague Semantics has incorporated
the suggestion, first made by Bennett
[1974], that Montague's treatment of
Grammars, as well as what database
theory had to say about the semantics of
the modelled enterprise. This led to
the adoption of systematic
simplifications to the PTQ translations
from English to logic wherever these
were suggested by the simplified view of
the semantics of the enterprise provided
by the database model. Moreover, since
we were not attempting to develop a
semantic theory of questions for English
in general, these simplifications are
introduced into the translation process
as early as possible. This has the dual
effect of making some of the PTQ theory
a little more accessible, and
eliminating the need to resort to the
less computationally attractive

In addition, the following criteria
have guided some of our decisions. (1)
The theory should fall within the
general confines of Montague's
framework, i.e., syntax and semantics
defined in parallel, with the semantics of
a phrase defined compositionally in
terms of the semantics of its
components. (2) Proper treatment of the
interaction of questions and
quantifiers; as PTQ successfully
accounts for multiple readings of
sentences with interacting quantifiers
("A woman loves every man"), our
solution allows for all of the readings
of questions involving quantified terms
("Who manages every employee?"). (3)
Provision for Y/N questions,
WH-questions, temporal questions
("when"), and multiple WH-questions
("Who sells what to whom?"). We have
made little attempt to develop a
sophisticated syntax for our fragment.
Since our primary concern has been
"getting the meaning right," we felt
that a too broad syntactic coverage
might obscure our major points. We
believe that the QE-III theory of
questions, particularly our proposal to

provided a pragmatic interpretation in
a manner to be explained.
common nouns (CNs) and intransitive verbs (IVs) as denoting sets of individual concepts (ICs) is unduly complicated. Under Bennett's suggestion both CNs and IVs denote sets of simple individuals; this simplifies the typing scheme of English categories in these fragments. In [Clifford & Warren 1983] the database concepts of key attributes and role attributes are identified, respectively, with "ordinary" CNs (which reduce to sets of entities in PTQ by means of MP-1) and "extraordinary" CNs (which denote sets of ICs). Accordingly we have not adopted the Bennett type system, but have instead maintained the PTQ treatment.

B. Verbs

Montague's semantic treatment of verbs leave them completely unanalyzed; thus, for example, the English verb "walk" translates into the constant "walk" in IL, "love" into "love", etc. Because we use a database as a representation of the logical model, we can provide an analysis of English verbs that takes into account the meaning of verbs as encoded in the database. As an example, the translation of "manage" in our fragment is given as:

\[ \lambda \text{W} \lambda x \lambda z \text{W}(i)(\lambda y \text{ASSOC}(y(i), x) \& \text{EMP}(i)(y(i)) \& \text{MGR}(i)(x)) \]

This expression is of the same logical type as manage in a PTQ-like treatment, and combines with Terms in the same way, but it does not leave "managing" unanalyzed. Instead it specifies that its subject x must be an IC that is a MGR, and its object must be an entity that is an EMP, and these two must be ASSOCiated in the database schema. In general the translation of any verb in QE-III specifies the attribute of its subject (or the disjunction of alternatives, if any). The translation of a TV further specifies the attribute(s) of its direct object, and a DT of its indirect object. Moreover any relationship(s) among these attributes are specified.

C. Tenses

Extensions to PTQ have had to handle the issue of tense and its interaction with other components of a sentence. We agree with Dowty's [1979] premise that tense is a property of the clause as a whole, and not merely of the verb. This is particularly important when, as in QE-III, there are different kinds of sentences: declaratives, WH questions, Y/N questions, and WHEN questions. For under a straightforward extension of the treatment of tense in PTQ, the number of rules would proliferate alarmingly, since separate rules would be needed for each kind and tense of sentence formed by conjoining a Term and a VP. For this reason we incorporated into QE-III the additional syntactic categories of tensed sentences of each variety, and have modified the Subject + Predicate rule (S4 in PTQ) to create an untensed sentence. Additional rules for each tense create the final, tensed version of any sentence.

D. Database Questions

Numerous researchers have examined the question, "What is an appropriate formal treatment of the semantics of questions?" ([Hamblin 1973], [Karttunen 1977], [Bennett 1977 & 1979], [Belnap 1982], [Hausser & Zaefferer 1979], are among the many who have tried to formulate an answer within a Montague Grammar framework.) We propose in our theory that the proper place for considering the answer(s) to a question is in a separate theory of pragmatics for the language. We have not yet proposed a completely general theory of pragmatics. But we believe that incorporating a formal pragmatic component to our fragment that treats the notion of a response to a question is defensible as at least one component of a theory of language use. Our formalization of a pragmatic component to the theory of QE-III accords well with what Stalnaker [1972] sees as the goals of "a formal semiotics no less rigorous than present day logical syntax and semantics." Those goals, he goes on to say, include an analysis of such linguistic acts as "assertions, commands,..., requests... to find necessary and sufficient conditions for the successful (or perhaps in some cases normal) completion of the act."

In its technical details our approach is both simple and elegant. It removes from the semantics the burden of providing an account of the response to a question, and allows it to do what semantics has always done best, account for reference. Then, just as the semantics of a language is based upon its syntax, the pragmatics is based upon both the syntactic and semantic analyses (in Hamblin's [1973] phrase, it "complements syntax and semantics." ) The simplicity with which we can state the formal pragmatic rules for our fragment, to capture the notion of the answer to a question, is based upon this ability to use both the syntax and the semantics to build the pragmatic theory. Two examples must suffice here to illustrate these ideas.
A pragmatic interpretation of YNQs that meets the criteria set forth in section III is not difficult to obtain. Since we want to interpret YNQs as either "yes" or "no", they can be defined to denote objects in \{0,1\}. But this is just the denotation set of the corresponding declarative sentence expressing the proposition that the YNQ asks. Thus we easily meet our criteria by providing that a YNQ denote the same proposition as that denoted by the declarative sentence from which it was derived. For example, "John manages the shoe department" would roughly be translated as:

manage'(i)(John, Shoe Dept.)

This formula is true with respect to a state \(i\) just in case John manages the shoe department in that state. Our analysis of the corresponding question "Does John manage the shoe department?" provides that it is derived syntactically from "John manages the shoe department" and that semantically and pragmatically it denotes the same object in the model. Under this view, then, a formula in the logic essentially "questions" the model as to its truth or falsity in the same way that a YNQ questions the database for the response "yes" or "no."

WH-questions in QE-III denote (a semantic concept) just as declarative sentences do. Thus the WH-Question "Who manages whom?" and the declarative sentence "He manages him" both receive the same semantic analysis:

\[ \exists x \{ x(i)=u-2 \& EMP(i)(u-1) \& MGR(i)(x) \& ASSOC(u-1,x) \} \]

Both are treated as denoting the same object with respect to an index, a variable assignment, and a model. But they are interpreted differently in the pragmatics. The pragmatics is defined as a function that, given a derivation for an expression of QE-III together with its syntactic category and its denotation (semantics), returns a (possibly) new object in the same model. Thus, although we view pragmatics as a separate component of a language theory, it is closely allied to the semantics -- both provide interpretations of linguistic expressions within the context of the same logical model. The formal definition of the pragmatic component provides that these two sentences, interpreted pragmatically, denote what the following expressions of IL-s denote:

who manages whom? ----> \[ \exists u-2 \& EMP(now)(u-1) \& MGR(now)(x) \& ASSOC(u-1,x) \]

he manages him ----> \[ 3 x \{ x(i)=u-2 \& EMP(i)(u-1) \& MGR(i)(x) \& ASSOC(u-1,x) \} \]

The pragmatic interpretation of the question is the set of n-tuples that answer it, while of the declarative sentence it is the same as its denotation. The pragmatics for QE-III is thus a simple theory of the effects of producing an expression in that language within the assumed context of a question-answering environment. That is, we assume that a user of QE-III is using the language to produce some effect within this context, and it is this effect (a representation of the answer to the question) which we formalize as the pragmatic component of the language definition.

V. CONCLUSIONS

QE-III is a formal English query language for historical databases whose definition is provided in three distinct parts. First we define the syntactic component: the categories of the language, the basic expressions of these categories, and the rules of formation. Together these constitute an inductive definition of the set of meaningful expressions of QE-III. The semantics of language is presented next, following Montague's general procedure in PTQ. This consists of giving, for each syntactic rule, a corresponding rule of translation into the logic IL-s, for which a direct semantic interpretation has already been specified. Finally, we provide a pragmatics for the language when used in the assumed context of a question-answering system. The pragmatics consists of a set of rules that together define a function which, for any derivation tree of an expression in the language, provides what we call its pragmatic interpretation.

QE-III is an attempt to demonstrate that a successful formal treatment can be given to a Natural Language database querying facility, through the medium of a formal intensional logic. We view this work as important for two reasons. First, it represents the first attempt to adapt the ideas of Montague Grammar to a practical problem. Most research since the PTQ paper has either been in the form of extensions or modifications to its linguistic or logical theory, or of computer implementations of the theory. Our work tries to show that this theory of language can serve as the
Second, it represents a change in emphasis in approaching the NLQ problem from the engineering approach -- get as much coverage as possible and get the system to work -- to a more formal approach -- proceed in small steps and develop a formal theory of what you do with each step that you take. This work represents only a first step in this direction within a Montague Semantics framework. The QE-III fragment is certainly not adequate to express all of the queries that one would want to present to an HDB. We hope, however, that it will lay the groundwork for a formal theory of database querying that is both extendible and implementable.

ACKNOWLEDGMENTS

This research is part of the author's Ph.D. thesis done at SUNY Stony Brook under the direction of David S. Warren; his encouragement and support are gratefully acknowledged.

REFERENCES


