Information Technology for Organisational Systems
H.-J. Bullinger et al. (Editors)
Elsevier Science Publishers B.V. (North-Holland)
ECSC-EAEC-EEC, Brussels-Luxembourg, 1988

QUERYING DATABASES IN NATURAL GREEK

Panagiotis Stamatopoulos, Alexis Arvillias, Maria Katzouraki, George Philokyprou (*)

Computer Center, NRCPS Democritos, Aghia Paraskevi, Athens, Greece

(*) University of Athens, Panepistimioupolis, Athens, Greece

A natural language system for understanding sentences stated in Greek is presented in this paper. These sentences are considered to apply to a database environment, in which they act as database queries. The system is not restricted to a specific domain, as it consists of a general part and a special one. Adopting different versions of the latter part makes it possible for the system to work on various applications. The acceptable subset of Greek is described in the metamorphosis grammars formalism. Every sentence belonging to the Greek subset is translated into a semantic structure whose interpretation, given a certain database, yields the appropriate answer to the user. The system has been implemented in Prolog, a programming language based on the principles of logic programming.

1. INTRODUCTION

Existing *data manipulation languages* (SQL, QBE, SEQUEL, SQUARE etc.) provide a formal way for retrieving information from *databases*. While these languages are adequate for programmers, a non specialized person needs a more user friendly interface to interact with databases. Evidently, *natural language understanding systems*, which recognize the user's native language, fulfill the friendliness requirement.

Recently developed natural language understanding systems employ *metamorphosis* [12], *definite clause* [9], *extraposition* [10] or other variations of *logic grammars* [15] to define the subset of the language being processed. Such systems have been created for English [5,7,8,14], French [4], German [11], Spanish [6] and Portuguese [3]. As far as the implementation is concerned the only programming tool used is the language *Prolog* [1,2].

All of the previously mentioned systems translate each natural language sentence into a formal structure, possibly through successive transformations to intermediate representations. This structure is either a Prolog goal [5,14] or a codification of the sentence according to a well defined semantic framework [3,4,6,7,8,11]. In order to yield an answer to the sentence, the surrounding Prolog system executes the Prolog goal in the first case, while in the second case a Prolog program is used to interpret the codification of the sentence.

Some systems [3,4,6] are based on the semantic framework proposed by Alain Colmerauer [13]. In this framework the *quantifiers* (articles, pronouns, numerals etc.) are of special importance. Moreover, a *three valued logic* is defined, where, besides the well known values "true" and "false", the value "undefined" is introduced. This value corresponds to sentences that are neither "true" nor "false" because of the existence of *false presuppositions*.

Following the developments already reported, we designed and implemented a natural language understanding system for database access adapted to Greek. For this purpose we introduced an entirely new semantic framework, to define the translation of every acceptable Greek sentence into a *semantic structure*. The most remarkable characteristic of our semantic framework is the distinction between *definite* and *indefinite* quantifiers, a feature which does not appear in Colmerauer's framework. This distinction leads on the one hand to the correct interpretation of sentences containing false presuppositions and on the other to the efficient implementation of the natural language understanding system.

It is worthwhile noting that the design of the system was also highly affected by the specific natural language being analyzed. We mention the declension system of Greek, the partial freedom related to word order and the semantic differences existing between Greek and other natural languages. For the definition of the acceptable Greek subset we used metamorphosis grammars. There has been no attempt to handle unrestricted Greek language, since, as it is generally admitted for every natural language, such a goal is unfeasible, at least in the foreseeable future.

The system was implemented using a Prolog interpreter developed on an IBM AT (6 MHz) of our laboratory and was functionally tested on a couple of application domains. The first domain is a subset of the Balkan States geography and the second one is related to radioactivity measurements. Given the portability of the system it could be equally readjusted to any other world.

2. THE PROPOSED SEMANTIC FRAMEWORK

2.1. The model of the surrounding world

Natural language understanding is meaningful only when a specific world domain is considered, whose data should be organized according to a given model.

In the model we propose the *atoms* of the world are classified into *groups*. There are also *relations* between atoms belonging to certain groups. A *subgroup* is a subset of a group whose atoms relate to a given atom of another group.

The groups of the surrounding world are organized in a *group tree*. Figure 1 shows the group tree for the Balkan States geography world.

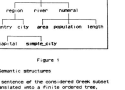

Figure 1

2.2. Semantic structures

Every sentence of the considered Greek subset is translated into a finite ordered tree, called the *semantic structure* of the sentence. The nodes of a semantic structure are special keywords, codifications of atoms, groups or relations and names of variables. The syntactic components of a sentence are mapped into semantic structures, which, when properly arranged, form the semantic structure of the sentence.

In what follows we present informally how different types of syntactic expressions correspond to semantic structures.

a) The conjunction of common or proper nouns corresponding to atoms is mapped into a *collection*.

Example :
... η Ελλάδα, η Βουλγαρία και η Αλβανία ...
(... Greece, Bulgaria and Albania ...)

Semantic structure :
collection([['Ελλαδα'],['Βουλγαρια'],
['Αλβανια']])

b) Common nouns corresponding to groups are mapped into *group structures*.

Example :
... εκτάσεις ...
(... areas ...)

Semantic structure :
group_structure([numeral,area],X)

c) Common nouns which are preceded by definite quantifiers are mapped into sets. The absence of relative sentences designating a common noun is coded as an *empty structure*. As definite quantifiers are considered, the definite articles possibly followed by a cardinal, an ordinal, a comparative adjective, a cardinal and a comparative adjective, the adjective "μέσος" (mean) or the adjective "συνολικός" (total). Moreover, the adjective "όλοι" (all) followed by a definite article forms a definite quantifier.

Example :
... οι 5 πρωτεύουσες ...
(... the 5 capitals ...)

Semantic structure :
set(the(5),X,group_structure([region,city,
capital],X),empty_structure)

The collections and the sets are called *complement structures*.

d) Common nouns designated by anything coded as a complement structure are mapped into *subgroup structures*.

Example :
... πληθυσμός της Αθήνας ...
(... population of Athens ...)

Semantic structure :
subgroup_structure([numeral,population],X,
collection([['Αθηνα']]))

The group and subgroup structures are called *selection structures*.

e) Declarative sentences whose quantifiers (in case they exist) are definite, are mapped into *relational structures*.

Example :
Η μεγαλύτερη χώρα συνορεύει με τη Βουλγαρία.
(The biggest country borders Bulgaria.)

Semantic structure :
relational_structure(all_to_all,yes,borders,
[set(minmax(1,bigger,[numeral,area]),X,
group_structure([region,country],X),
empty_structure),collection(
[['Βουλγαρια']])])

f) Declarative sentences containing at least one indefinite quantifier are mapped into *logical structures*. Indefinite quantifiers are the cardinals, the indefinite pronouns "κάποιος" (a), "μερικοί" (some), "κανένας" (no) and "κάθε" (each) or one of the expressions "τουλάχιστον" (at least), "το πολύ" (at most), "περισσότεροι από" (more than) or "λιγότεροι από" (less than) followed by a cardinal.

Example :
3 ποτάμια διαρρέουν την Τουρκία.
(3 rivers flow through Turkey.)

Semantic structure :

```
logical_structure(exactly(3),X,group_structure(
   [river],X),empty_structure,
   relational_structure(all_to_all,yes,flows,
   [collection([X]),collection(
   [['Τουρκια']])]))
```

In case of more than one indefinite quantifier in a sentence the following rules apply.

1. Every indefinite quantifier belonging to the main part of a sentence dominates all the indefinite quantifiers of possible relative sentences.

2. When a noun preceded by indefinite quantifier is followed by a designator containing another indefinite quantifier, then the designator's quantifier dominates the noun's quantifier.

3. If the subject and the objects of a verb are introduced by indefinite quantifiers, then the quantifiers' hierarchy decreases from left to right, according to the natural order of their appearance.

g) Compound declarative sentences are mapped into *conjunction structures*.

Example :
Η Ελλάδα είναι η τέταρτη χώρα και δε συνορεύει με τη Ρουμανία.
(Greece is the fourth country and does not border Rumania.)

Semantic structure :

```
conjunction_structure(relational_structure(
   one_to_one,yes,is,[collection([['Ελλαδα']]),
   set(order(4,bigger,[numeral,area]),X,
   group_structure([region,country],X),
   empty_structure)]),relational_structure(
   all_to_all,no,borders,[collection(
   [['Ελλαδα']]),collection([['Ρουμανια']])]))
```

The relational, logical and conjunction structures are called *declarative structures*.

h) Relative sentences are mapped into declarative structures.

Example :
... οι πόλεις οι οποίες βρίσκονται στην Αλβανία ...
(... the cities which belong to Albania ...)

Semantic structure :

```
set(all,X,group_structure([region,city|Y],X),
   relational_structure(all_to_all,yes,belongs,
   [collection([X]),collection(
   [['Αλβανια']])]))
```

i) Interrogative sentences introduced with the pronoun "ποιος" (which) are mapped into "*which*" or into "*which_are*" structures.

Example :
Ποια ποτάμια διαρρέουν τη Γιουγκοσλαυία;
(Which rivers flow through Yugoslavia?)

Semantic structure :

```
which(X,group_structure([river],X),
   empty_structure,relational_structure(
   all_to_all,yes,flows,[collection([X]),
   collection([['Γιουγκοσλαυια']])]))
```

j) Interrogative sentences introduced with the pronoun "πόσοι" (howmany) are mapped into "*howmany*" or into "*howmany_are*" structures.

Example :
Πόσες είναι οι πόλεις της Ελλάδας;
(Howmany are the cities of Greece?)

Semantic structure :

```
howmany_are(set(all,X,subgroup_structure(
   [region,city|Y],X,collection([['Ελλαδα']])),
   empty_structure))
```

Every "which", "which_are", "howmany" and "howmany_are" structure is called *interrogative structure*.

2.3. Values of semantic structures

A variable appearing in a semantic structure is *bound* if it is introduced by a selection structure which is a subtree of the semantic structure, otherwise it is *free*.

A semantic structure is *closed* if it contains only bound variables, otherwise it is *open*. Every complete sentence is mapped into a closed semantic structure, while parts of the sentence may be represented by either open or closed semantic structures.

For a given status of the surrounding world a *value* can be assigned to every closed semantic structure. The possible values of a closed semantic structure are the following.

1. truth_value(t)
   where t: true, false or undefined
2. collection(c)
   where c: a list containing lists of atoms
3. number(n)
   where n: a non negative integer number

2.4. On Colmerauer's semantic framework

In the semantic framework proposed by Alain Colmerauer [13] there is no distinction between definite and indefinite quantifiers, although definite quantifiers introduce in most cases independently computable sets. This unified approach of quantifiers fails to interpret correctly sentences belonging to a certain class. For example, consider the sentences :

Η χώρα που είναι μικρότερη από την Αλβανία συνορεύει με 3 χώρες.

(The country which is smaller than Albania borders 3 countries.)

and

3 χώρες συνορεύουν με τη χώρα που είναι μικρότερη από την Αλβανία.
(3 countries border the country which is smaller than Albania)

Given that in the Balkan States geography there is no country smaller than Albania, according to Colmerauer's semantic framework the formal representation of the first sentence is assigned the value "undefined", while that of the second one is assigned the value "false", although the two sentences are semantically equivalent. This phenomenon appears between sentences having the following features. Both have the same main verb representing a symmetric relation. In the first sentence the main quantifier of the subject is definite, introducing a false presupposition, and that of the object is indefinite. The second sentence is derived by mutually interchanging the subject and the object of the first one.

The system we implemented tries to compute for both sentences the set of "the country which is smaller than Albania". As this attempt fails, because of the existence of a false presupposition, both semantic structures of the previous sentences are assigned the value "truth_value(undefined)", thus yielding the correct answer.

Moreover, Colmerauer's semantic framework fails to lead to efficient implementation of natural language understanding systems in terms of evaluating the computable sets introduced by definite quantifiers. Let's examine the sentence :

2 ποτάμια διαρρέουν τη χώρα που συνορεύει με τις χώρες των οποίων οι πρωτεύουσες είναι μεγαλύτερες από την πρωτεύουσα της Βουλγαρίας.
(2 rivers flow through the country which borders the countries whose capitals are bigger than the capital of Bulgaria.)

In order to assign a value to the semantic formula of the above sentence, any natural language understanding system following Colmerauer's proposal has to compute M times the set of "the country which borders the countries whose capitals are bigger than the capital of Bulgaria", M x N times the set of "the countries whose capitals are bigger than the capital of Bulgaria" and M x N x N times the set of "the capital of Bulgaria", where M is the number of rivers and N the number of countries in the surrounding world.

Instead, in the natural language understanding system we implemented, based on the semantic framework we propose, each of the previous sets is computed only once. This is derived from the fact that the assignment of values to semantic structures containing nested sets follows a bottom up procedure. Considering the specific example, the third set is initially computed and the result is used for the computation of the second set. The new value obtained is passed to the first set, whose computation allows for a value to be assigned to the semantic structure of the original sentence.

3. IMPLEMENTATION OF THE SYSTEM

3.1. Organization

Figure 2 shows the general outline of the natural language understanding system we designed and implemented. The system consists of three main parts, the *supervisor*, the *natural language processor* and the *semantic structure processor*.

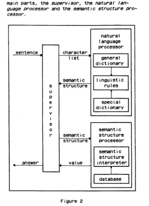

Figure 2

Each sentence posed to the system is transformed by the supervisor to a list containing the characters of the sentence. This list is passed to the natural language processor which creates the corresponding semantic structure, if the sentence belongs to the acceptable subset of Greek. The semantic structure processor analyzes the semantic structure just created and assigns to it a value which is given to the supervisor. Finally, the supervisor formulates the appropriate answer to the user's original sentence.

3.2. Natural language processor

In order to describe the acceptable subset of Greek we defined a set of *metamorphosis grammars rules* [12] which was subsequently transformed to an equivalent set of *metamorphosis grammars rules in normal form*. For each rule belonging to the latter set there exists a corresponding Prolog clause. All these clauses are contained in the natural language processor and are distributed among its three parts, the *general dictionary*, the *special dictionary* and the *linguistic rules*.

The general dictionary consists of Prolog clauses which provide information about Greek letters and phonetics, word themes, word endings and words. All this information is necessary for any world domain to be implemented and for this reason it is application independent. Examples of metamorphosis grammar rules embedded in the general dictionary are the following.

```
1.  verb_to_be  -->
       ['E','I','N','A','I'].
2.  definite_article(feminine,plural,
          genitive,with_initial_t)  -->
       ['T','Ω','N'].
3.  theme_of_mean  -->
       ['M','E','Σ'].
4.  theme_of_which  -->
       ['O','Π','O','I'].
5.  ordinal(8)  -->
       ['O','Γ','Δ','O'].
6.  exactly  -->
       ['A','K','P','I','B','Ω','Σ'].
7.  full_preposition(with)  -->
       ['M','E'].
8.  conjunction_and  -->
       ['K','A','I'].
9.  question_mark  -->
       [';'].
10. vowel,['E']  -->
       ['E'].
11. verb_ending(ei_oyn,singular)  -->
       ['E','I'].
12. neutral_singular_nominative_ending(
          to_as_ta_ata)  -->
       ['A','Σ'].
```

The metamorphosis grammar rules containing information about themes of words and words depending on the specific application considered correspond to Prolog clauses which are included in the special dictionary. Examples of such rules are the following.

```
1.  theme_of_common_verb(ei_oyn,Number1,
          [argument(Gender1,Number1,nominative,
          no_preposition,[river],Complement1),
          argument(Gender2,Number2,accusative,
          no_preposition,[region,country],
          Complement2)],all_to_all,flows,
          [Complement1,Complement2])  -->
       ['Δ','I','A','P','P','E'].
2.  theme_of_proper_noun(to_o_ta_a,plural,
          [region,city,capital],'Τιρανα')  -->
       ['T','I','P','A','N'].
3.  theme_of_group_common_noun(o_os_oi_oi,
          [numeral,population])  -->
       ['Π','Λ','H','Θ','Y','Σ','M'].
4.  measurement_unit(nominative,[numeral,
          area])  -->
       ['T','E','T','P','A','Γ','Ω','N','I',
       'K','A','@','X','I','Λ','I','O','M','E',
       'T','P','A'].
```

The character '@' in the last rule is a special one acting as word delimiter.

The special dictionary includes also Prolog facts which provide semantic information relevant to the application of interest.

The third part of the natural language processor, the linguistic rules, contain Prolog clauses which implement a set of metamorphosis grammar rules. This set of rules describes the acceptable subset of Greek from the syntactic and the semantic points of view. For example, the following couple of rules define the valid cases for a sentence.

```
1.  sentence(Semantic_structure)  -->
       interrogative_sentence(
          Semantic_structure),
       question_mark,
       ['@'].
2.  sentence(Semantic_structure)  -->
       declarative_sentence(
          Semantic_structure),
       fullstop,
       ['@'].
```

Another example is the definition of valid natural language expressions which represent atoms of the surrounding world.

```
1.  atom(Gender,Number,Case,Group,Atom)  -->
       article(Gender,Number,Case,Initial_t),
       proper_noun(Gender,Number,Case,Group,
          Atom),
       ['@'].
2.  atom(Gender,Number,Case,Group,Atom)  -->
       cardinal(Atom),
       ['@'],
       possible_measurement_unit(Case,Group).
3.  atom(Gender,Number,Case,Group,Atom)  -->
       common_noun_phrase(Gender,Number,Case,
          Group,Atom).
4.  atom(Gender,Number,Case,Group,Atom)  -->
       special_atom(Gender,Number,Case,Group,
          Atom),
       ['@'].
```

It is evident, from the examples already presented, that the terminal symbols of the grammar we defined are characters. This choice is a result of the declension system of the Greek language, which would lead to the definition of dictionaries with extremely large sizes if the terminal symbols were defined at the word level.

3.3. Semantic structure processor

The semantic structure processor consists of the *semantic structure interpreter* and the *database*.

The Prolog clauses composing the semantic structure interpreter define logically a set of procedures which are responsible for the evaluation of the semantic structures produced by the natural language processor. A semantic structure may be considered as a program which is processed by the semantic structure interpreter to derive a value as output, while accepting the present status of the surrounding world as input. The procedure which is initially activated when the supervisor triggers the semantic structure processor is defined as follows.

```
1.  value(Semantic_structure,Value)  :-
       interrogative_structure(
          Semantic_structure,Value),
       !.
2.  value(Semantic_structure,Value)  :-
       declarative_structure(
          Semantic_structure,Value),
       !.
3.  value(Semantic_structure,truth_value(
          undefined)).
```

The database is the formal description of the surrounding world. It is in this part of the system that the relations, the groups and the subgroups of the world are defined.

The relations are distinguished into *actual* and *virtual*, depending on the way of their implementation. An actual relation is defined by a set of Prolog facts, while for the definition of a virtual relation a set of Prolog rules is required. For each relation there exists a certain *schema*, which is defined by the name of the relation and the names of the groups participating in the relation. We consider the following relations for the implementation of the Balkan States geography.

```
1.  belongs(<city>,<country>)
2.  is_region(<region>,<population>)
3.  is_city(<city>,<population>,<country>)
4.  borders(<country>,<country>)
5.  flows(<river>,<country>)
6.  is_country(<country>,<area>,<population>)
7.  is_capital(<capital>,<population>,
       <country>)
8.  is_simple_city(<simple_city>,<population>,
       <country>)
9.  is_river(<river>,<length>)
```

Relations 1 to 3 constitute the virtual database and relations 4 to 9 the actual one.

A small sample of Prolog clauses implementing relations of the database is as follows.

```
1.  is_city(City,Population,Country)  :-
       is_capital(City,Population,Country).
2.  is_city(City,Population,Country)  :-
       is_simple_city(City,Population,Country).
3.  borders('Βουλγαρια','Ρουμανια').
4.  flows('Δουναβης','Γιουγκοσλαυια').
5.  is_country('Ελλαδα',131944,8890000).
6.  is_capital('Αγκυρα',1208790,'Τουρκια').
7.  is_simple_city('Δυρραχιο',53160,'Αλβανια').
8.  is_river('Αξιος',380).
```

Every group corresponding to a leaf of the group tree is defined in the database in a way shown by the examples below.

```
1.  group([region,city,capital],[Capital])  :-
       is_capital(Capital,Population,Country).
2.  group([numeral,population],[Population,
          Region])  :-
       is_region(Region,Population).
```

The subgroups of the surrounding world are also defined in the database in a way similar to the definition of groups.

```
1.  subgroup([river],[River],Country)  :-
       flows(River,Country).
2.  subgroup([numeral,area],[Area,Country],
          Country)  :-
       is_country(Country,Area,Population).
```

3.4. Portability of the system

The modular design of the natural language understanding system we presented allows for an easy implementation of various applications. Four parts of the system, i.e. the supervisor, the linguistic rules, the general dictionary and the semantic structure interpreter, constitute the *nucleus* as they are application independent. The sizes of these parts are 49, 255, 265 and 109 Prolog clauses respectively, giving a total of 678 clauses. The only parts of the system that need to be created in case we adopt another domain are the special dictionary and the database. These latter parts form the *surroundings* of the nucleus. In the Balkan States geography domain the special dictionary and the database contain 61 and 66 Prolog clauses respectively. In the radioactivity measurements domain the special dictionary consists of 255 clauses and the database of 543 clauses.

4. SAMPLE QUERIES

Indicative queries posed to the system are presented below along with the answers obtained and the execution times needed for the IBM AT configuration employed.

a) User :
Η Γιουγκοσλαυία συνορεύει με τη μικρότερη σε πληθυσμό χώρα.
(Yugoslavia borders the smallest country in terms of population.)

System :
Η πρόταση είναι σωστή
(The sentence is correct)

Natural language processor time : 3.08 sec
Semantic structure processor time : 0.88 sec
Total time : 3.96 sec

b) User :
Πόσα ποτάμια της Βουλγαρίας διαρρέουν 3 χώρες;
(Howmany rivers of Bulgaria flow through 3 countries?)

System :
Υπάρχουν 2
(There exist 2)

Natural language processor time : 3.90 sec
Semantic structure processor time : 2.03 sec
Total time : 5.93 sec

c) User :
Ποιες πόλεις βρίσκονται στη χώρα της οποίας η πρωτεύουσα είναι μεγαλύτερη από την πρωτεύουσα της Ρουμανίας;
(Which cities belong to the country whose capital is bigger than the capital of Rumania?)

System :
Αθηνα,Θεσσαλονικη
(Athens,Thessaloniki)

Natural language processor time : 9.78 sec
Semantic structure processor time : 5.27 sec
Total time : 15.05 sec

d) User :
Ποια είναι η μέση έκταση των χωρών που συνορεύουν με την τρίτη χώρα;
(Which is the mean area of the countries that border the third country?)

System :
183356

Natural language processor time : 23.68 sec
Semantic structure processor time : 2.91 sec
Total time : 26.59 sec

e) User :
Ποια είναι τα μήκη των ποταμιών τα οποία διαρρέουν κάποια χώρα που είναι μικρότερη σε έκταση από τη Ρουμανία;
(Which are the lengths of the rivers which flow through a country that is smaller than Rumania in terms of area?)

System :
380:Αξιος,2850:Δουναβης,530:Εβρος
(380:Axios,2850:Danube,530:Evros)

Natural language processor time : 26.31 sec
Semantic structure processor time : 4.67 sec
Total time : 30.98 sec

f) User :
Ποιες είναι οι εκτάσεις των χωρών που διαρρέει ο Εβρος και οι οποίες συνορεύουν με τη χώρα της οποίας η πρωτεύουσα είναι η πόλη της οποίας ο πληθυσμός είναι μεγαλύτερος από 600000 κατοίκους και μικρότερος από 1000000 κατοίκους;
(Which are the areas of the countries that Evros flows through and border the country whose capital is the city whose population is greater than 600000 citizens and less than 1000000 citizens?)

System :
131944:Ελλαδα,780576:Τουρκια
(131944:Greece,780576:Turkey)

Natural language processor time : 44.38 sec
Semantic structure processor time : 11.20 sec
Total time : 55.58 sec

5. CONCLUSION

We have described a natural language understanding system which makes it possible for a subset of Greek to act as a query language for databases. This system is based upon a novel semantic framework, which seems to be more advantageous compared to that of Alain Colmerauer in terms of both correctness and implementation. As far as the efficiency of the system is concerned, it is sufficiently satisfactory as even relatively complex queries are answered in less than a minute on an ordinary personal computer using a Prolog interpreter without special performance characteristics. The execution times could be dramatically improved if a Prolog compiler were used instead, which would result in a 2-3 orders of magnitude better performance.

REFERENCES

[1] Clocksin, W. and Mellish, C., Programming in Prolog, Springer Verlag, Berlin, 1981.
[2] Sterling, L. and Shapiro, E., The art of Prolog, MIT Press, Cambridge, 1986.
[3] Coelho, H., A program conversing in Portuguese providing a library service, Ph.D. Thesis, University of Edinburgh, 1979.
[4] Perichaud, L., Consultation en Français d'une banque de données sur fichiers et mise en place du système Prolog nécessaire, Thèse de 3ème cycle, Université Aix-Marseille, 1981.
[5] Warren, D. and Pereira, F., An efficient easily adaptable system for interpreting natural language queries, DAI Research paper, No. 155, University of Edinburgh, 1981.
[6] Dahl, V., Translating Spanish into logic through logic, American journal of computational linguistics, Vol. 7, No. 3, pp. 149-164, 1981.
[7] McCord, M., Using slots and modifiers in logic grammars for natural language, Artificial intelligence, Vol. 18, No. 3, pp. 327-367, 1982.
[8] Hadley, R., Shadow: A natural language query analyzer, Computers and mathematics with applications, Vol. 11, No. 5, pp. 481-504, 1985.
[9] Pereira, F. and Warren, D., Definite

clause grammars for language analysis - A survey of the formalism and a comparison with augmented transition networks, Artificial intelligence, Vol. 13, No. 3, pp. 231-278, 1980.

[10] Pereira, F., Extraposition grammars, American journal of computational linguistics, Vol. 7, No. 4, pp. 243-256, 1981.

[11] Schwind, C., Logic based natural language processing, in: Dahl, V. and Saint-Dizier, P. (eds.), Natural language understanding and logic programming, pp. 207-219, North Holland, 1985.

[12] Colmerauer, A., Metamorphosis grammars, in: Bolc, L. (ed.), Natural language communication with computers, Lecture notes in computer science, No. 63, pp. 133-189, Springer Verlag, Berlin, 1978.

[13] Colmerauer, A., An interesting subset of natural language, in: Clark, K. and Tarnlund S.-A. (eds.), Logic programming, pp. 45-66, Academic Press, London, 1982.

[14] Filgueiras, M., A kernel for a general natural language interface, Proceedings of logic programming workshop, pp. 419-436, Albufeira, Portugal, 1983.

[15] Dahl, V., Current trends in logic grammars, Proceedings of logic programming workshop, pp. 578-607, Albufeira, Portugal, 1983.