*G. V. Lezin, K. K. Boyarsky,
E. A. Kanevsky, A. I. Popova*

PROGRAMMING OF TEXTS CONCEPTUAL TREATMENT

Introduction

Computer representation of knowledge is an actual but not yet solved problem of historical investigations methodology. The huge volume of knowledge in the humanities is presented, as a rule, in the form of texts. In historical investigations the texts of the literary monuments written in ancient languages as well as pieces of ancient scripts serve as the principal source of information. The problem of scientific knowledge formalisation is thus connected with the problem of computer representation of semantics of natural language texts and their conceptual modelling.

Within a wide scope of the problems relevant to the modelling of communications employing a natural language (NL), we are taking interest, first of all, in a special case of the NL communications, which aims at the accumulation, systematisation, and the conceptual modelling of the texts in a concrete objects area. This process is meant to create a sublanguage with a formalised semantics, providing computer "understanding" of the texts in a given sphere of communications. It is the process of creating a problem-oriented sublanguage where a user of the knowledge base is to determine main practical and communicative objects of information accumulation.

The studied approach to the use of the model of understanding of a NL-text requires realisation and investigation of two level-interrelations of a user with the knowledge base created by him. One of them, conventionally denoted as the "level of formation of communication sphere", is used in those cases when the base should be upgraded by a set of primary objects, in order to ascertain their properties or initial states, to complete a conventional list of semantic linkages, to introduce certain scripts or initial rules for the performing of plausible reasoning necessary for the text understanding. This is the level for communications employing formal means of interaction with the knowledge base. The next one (the NL-level) assumes exchange with the base using a problem-oriented sublanguage as well as automatic — that is, performed by the base analyser, extraction of conceptual information from the text.

In this paper we consider the formal means of interaction with the knowledge base on the level of creation a communication sphere. At the base of determining of the means under consideration lies a view on the knowledge base of a humanitarian investigation as on the system of hypertext formatted as an electronic card catalogue. In a number of works [1] were published the backgrounds of the general approach to creation of such a system and were considered certain elements of the technique of semantic encoding of NL-texts (with the example of employing the V. V. Martynov's semantic primitives [2]). The means of exchange with the knowledge base considered in this work are determined as a language for recording the texts content and for programming their conceptual treatment (a formal language). Using a definition "conceptual treatment" we follow R. Shank [3] and assume the procedures of the text analysis and synthesis, which are based on the scope of definitions and their interrelations created beforehand and upgraded in the process of the analysis.

1. The Formal Communication Language

In our approach to the definition of the formal language we were guided by the following requirements to the practical representation of the means of interaction with the knowledge base:

a) the formal language is a part of the general set of means for the description of a linguistic processor, the "carrier" for the problem-oriented language;

b) in the framework of this language, the means of hypertext (formation and requests at the reading), the philological toolkit of a humanitarian investigation (creation of thesauruses and lexical pointers, determination of semiotic juxtapositions and synonyms series), the means of control over factographical data bases, and the means of definition of a problem-oriented sublanguage (description of infor-

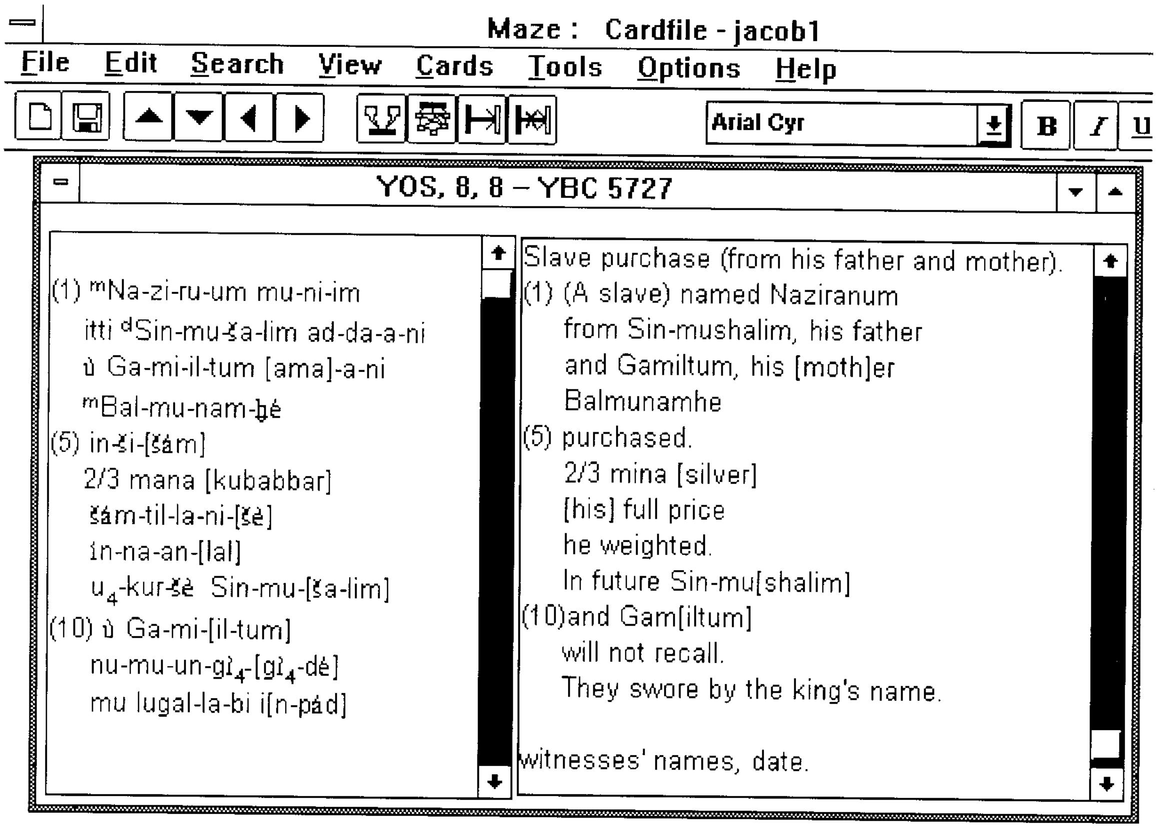

Fig. 1

mational structures of "conceptual memory" and of algorithms of message analysis) are integrated in a common complex.

The information in the card catalogue is represented by a scope of cards [4]. The structure of a card, the route for dividing information contained in it into separate elements (windows) are determined by the card pattern. For any element of the pattern one points its label, position in the card, and the type of information written in it (text or image). Further we consider only the elements possessing a text information. One card may keep several different texts. In *fig. 1* is given an example of the card with two texts (reproduced from the V. A. Yacobson card catalogue [5]).

The text in a fragment of a card may be considered as:

a) "inscription" — that is, a text where all features of its graphic representation are essential,

b) "message" — that is, a chain of characters possessing an informational content regardless the font used for any certain character.

The following types of relations between the cards are possible [6]:

a) to a given card a set of other cards may be juxtaposed — that is, it may be defined as an entry;

b) a given card may be included into various entries;

c) hypertext may be constructed to include a card into an entry (the hypertext means a tree-like net with a given inclusion of the card as its root and the branches of the card being named — a branch corresponds to a transition name in the hypertext.

A message is extracted from an inscription by transformation of its code into an alphabetical representation [7]. Such a transformation is defined by a special informational structure — the card catalogue alphabet. This structure assigns a list of images to a given character in various fonts.

For a given message may be determined a pathway for the dividing the parent mark chain into sets of marks with emphasising some of them (key words) and their interpretation. The sets of marks may be emphasised by either a user (by means of special graphic means for the emphasising, e. g., combining the images colour and attributes), or computationally (by determination of the text syntax). The emphasised mark sets, or precisely, their inclusions into a given text, may be saved in the system dictionaries.

The texts written in the formal language belong to a certain type of messages for computational treatment by the system. Further, we consider structure and methods for the treating of the formalised messages.

2. Objects and Relations

The formalised messages contain factocraphic information and instructions for its treatment. All acquired information is accumulated in the conceptual memory system, where it is integrated into a unified informational complex of the knowledge base. Treatment of the accumulated information according to the instructions given is initialised by inquiries to the conceptual memory. The latter may be considered as a united hypertext where, along with the data extracted from the messages, location of the messages in the hypertext and the routes between the messages are also fixed. Information is represented in the conceptual memory in a normalised (canonical) form.

Correspondingly, with this concept we differ external and canonical representations of messages. In this section we consider general structure and elements of external messages.

The messages are recorded as a sequence of sentences of the formal language. A sentence may either introduce a certain object or group sentences into a new object (conceptualisation), or be an instruction determining relations between various classes of sentences. The sentences are separated by the mark ";".

The sentences may be denoted by numerals or alphabetical characters. The sentences dealing with numbers are recorded in the syntax form identical with the commonly accepted (we are not going into details of these records here). The chains of alphabetical characters considered as an object (not as designation of an object) are separated by the mark "`". Further we consider the sentences concerning individual objects, conceptualisation, and instructions.

The *individual objects* are the essences *a priori* supposed as:

a) always having a designation (one, or several synonyms);

b) capable to be considered either as an element of a certain ensemble or as an ensemble comprising a scope of elements.

In addition, an object may be considered as relationship or as one of the elements of this relationship, all relations being regarded as asymmetrical.

The objects designations are recorded as a tentative sequence of alphabetical characters separated by the language spacers or brackets "[" or "]". The record S1 := S2 denotes that S1 and S2 are equivalent, S1 and S2 being different designations of the same object.

We consider the sentences as fragments of hypertext with the common contents. Therefore there is a possibility of a local and general designation of an object, the same general designation being used in various messages corresponds to one and the same object.

A local designation of an object is inherent in the message where it is used. Within a message one local designation corresponds to one object. In different messages similar local designations corresponds to different objects. The local designations have complex structure represented by a pair ⟨designation⟩ ⟨message name⟩ — along with the object designation a message is given, where the character is localised. In its turn, the message name joints designation of a card and, possibly, designation of the text on the card (when the card contains several texts). The following convention is accepted: a card designation is always prefaced with the prefix "#". A message name in local notation may be given by the prefix "#" alone if the current message is only assumed.

A record of type E1 : E2 establishes that the elemental object E2 (E2 is considered to be an element in this relation) belongs to the ensemble object E1 (E1 is considered to be an ensemble). We will further assume that E2 object concretises E1 object (that is E2 presents an example, or a concretisation of E1).

A record E1(E3) defines that the ensemble object E1 has an attribute E3.

More complex record, E4 (E3.E5), establishes that E5 is value of the E3 attribute of the elemental object E4, and that possibly there is an ensemble object E not indicated in the record and therefore unknown. In respect of the E, one may state that E4 specifies it, while E3 is its attribute. In other words, E4 specifies an unknown object E with the attribute E3, the value of the latter is E5.

Finally, a record E1 : E4 (E3.E5), being a statement concerning E4, evidently points to the existence of an object E1 with a specimen E4 and with the value E5 of its attribute E3.

The pairs:

⟨object-ensemble⟩ : ⟨object-element⟩, (1)
⟨object-ensemble⟩ (⟨object-attribute⟩), (2)

and triad:

⟨object-element⟩ (⟨object-attribute⟩.⟨value⟩) (3)

are suggested to be *terminal statements*. The object-element in the terminal statement (1), object-ensemble in (2) and object-element in (3) are *main objects* of these statements. The convention on the main object allows us to record the sets of terminal statements in a compact form of the language sentences. In particular, the sentence E1 : E2 : E3 is equal both to (E1 : E2) : E3 and to the set E1 : E2; E2 : E3, and the sentence E1 : (E2 : E3) is equivalent to E1 : E3; E2 : E3. With the traditional convention on the use of parentheses, we are free enough to combine statements when making a sentence of them in language is necessary.

EXAMPLE 1. The content of the card represented in *fig. 1* may be given by the following formal message:

```
# [YOS8,8—YBC 5727] :=
{legal_document : agreement : D# (object_
of_agreement.purchase-sale : P#,
obligation.(oath : K#));
family : F# (son. Naziranum#, dad. Sin-mushalim#,
mother. Gamiltum#);
P# (seller. (S# := {Sin-mushalim#, Gamiltum#}),
buyer.Balmunamhe#, merchandise.Naziranum#,
price. (silver : C# (unit.mina, amount.2/3));
K# (obliger.S#, to_the_name.king (name),
oath_object.P#))}.
```

Note that interpretation of the message is invariant in respect of the order of sentences.

It is often necessary (or convenient) to consider a set of sentences as an integral object. In the above example the Sin-mushalim# and Gamiltum# pair is one and the same subject of the agreement. When the set of statements taken into brackets { }, it indicates that this set should be accepted (and fixed) as an integral object. Following R. C. Shank [8], we designate such an object as *conceptualisation*.

Conceptualisations may be evident or differing by the form of its record. An evident conceptualisation is designated as

⟨designation⟩ := {⟨conceptualisation⟩}

(the symbol sequence ":= " means "this is").

Conceptualisations may be joined by links, e. g., of "cause-result" type. Thus in common cases a conceptualisation record has the following syntax structure:

⟨conceptualisation⟩ := {⟨set of statements⟩} | {⟨conceptualisation⟩ ⟨link⟩ ⟨conceptualization⟩}.

The links designations are not determined and their number is not evidently restricted. The fact that an object is designated at the junction of two conceptualisations indicates that the given designation corresponds to a link.

EXAMPLE 2. By designating relations of mutual determination (causal) as "R ⇔ R" we are capable to describe the fact of sale as mutually causal facts of change in owning:

{{owning_change : Y1# (from_whom.S#, to_whom.Balmunamhe#, ownership : Naziranum#)} R ⇔ R {owning_change : Y2# (from_whom.Balmunamhe#, to_whom.S#, ownership : silver : C# (unit.mina, amount.2/3))}}.

The *inquiries to the conceptual memory* have a form of sentences with mark "?". For example, the sentence ? ownership is considered as an inquiry to the conceptual memory: is there an object in it designated as ownership. The answers allowed are "yes" or "not". More interesting is the inquiry which has a form of logical function (predicate). For example, the sentence ? ownership : !X assumes that from the conceptual memory will be obtained all concretisations of the object ownership. The possible answers are more essential in their positive parts: "yes, !X = Naziranum, !X = silver..., end" or "not".

In the example of inquiry we used !X as the function variable. *Variables* are special objects of the conceptual memory. Their designations include prefix "!". The range of their definitions comprises the assembly of objects in the conceptual memory in its current state. The variable designations are localised within a sentence where they used. The possible statement are !X = E (the value of variable !X is the object E), !X ≠ E (not E), !X = !Y, !X ≠ !Y (the values of the variables are equal or unequal).

3. Semantemes and Guide bar

The *semantemes* [9] is a next type of statements of the language, which is used when a class of specimens should be determined and the new class structure should be defined through what was defined earlier. A semanteme construction has the form:

⊢⟨semanteme designation⟩ : ⟨variable⟩ (⟨attribute_1⟩ ⟨variable_1⟩, ..., ⟨attribute_k⟩. ⟨variable_k⟩) := [⟨a sequence of sentences⟩].

In this construction, the ⟨semanteme designation⟩, ⟨attribute_1⟩, ..., ⟨attribute_k⟩ denotes objects, and ⟨variable⟩, ⟨variable_1⟩, ..., ⟨variable_k⟩ denotes variables.

EXAMPLE 3. Using the semanteme construction, we may define the class of the objects purchase-sale on the basis of definition of the class of objects ownership_change:

⊢purchase-sale : !X1 (seller.!X2, buyer.!X3, merchandise.!X4, price.!X5) := [{ownership_change : !Y1 (from_whom.!X2, to_whom.!X3, ownership : !X4)} R ⇔ R {ownership_change : !Y2 (from_whom.!X3, to_whom.!X2, ownership : !X5)}];

The semanteme essence becomes clear if one consider the inquiries to conceptual memory connected with it. In response to the inquiry

? purchase-sale!Z1 (seler.!Z2, buyer.!Z3, merchandise.!Z4)

the conceptual memory neglects the corresponding semanteme. However, when the inquiry

? ownership_change : !X (from_whom.!Y1, to_whom.!Y2, ownership.!Y3)

is obtained, the conceptual memory will take into account the semanteme containing the object ownership_change and to response will consider the facts of purchase-sale attempting to extract an information on the ownership_change.

A semanteme points to the conceptual memory how to attempt achieve a target of the proposed search by analysing the facts integrating this goal. Semanteme controls an "ascending" search of the target.

Guide bar (procedures), unlike semantemes, control "descending" search of target: use of a guide is initiated by turning to its left part which is to be determined. The guide structure:

⊢⟨guide designation⟩ : ⟨variable⟩ (⟨attribute_1⟩. ⟨variable_1⟩, ..., ⟨attribute_k⟩. ⟨variable_k⟩) ⇐ [⟨sequence of sentences⟩], ..., [⟨sequence of sentences⟩];

The notation "⇐" means "follows from". In a concrete guide the attributes of determined object may be omitted.

EXAMPLE 4. Conceptual memory will establish the fact of the slave selling from the facts given in Example 1 if the following guide bar will be inputted in it:

man : !Z ⇐ [[man : !Y1; family : !Y2 (!Y3. !Y1, !Y4. !Z)]; [purchase-sale (seller. !Z]; [purchase-sale (seller. !Y := {!Z}]; [purchase-sale (buyer. !Z]];
slave_sale : !X (merchandise. !Y) ⇐ [purchase-sale : !X (merchandise. !Y); man : !Y];

The first guide of the guide bar establish that when even one of members of a family is man, hence, other members are men too. A seller and a buyer also are men. According to the second guide, the purchase of a slave corresponds to the event when the object of purchase-sale is a man.

Recursion in guide bar is allowed.

4. Canonical form of message representation

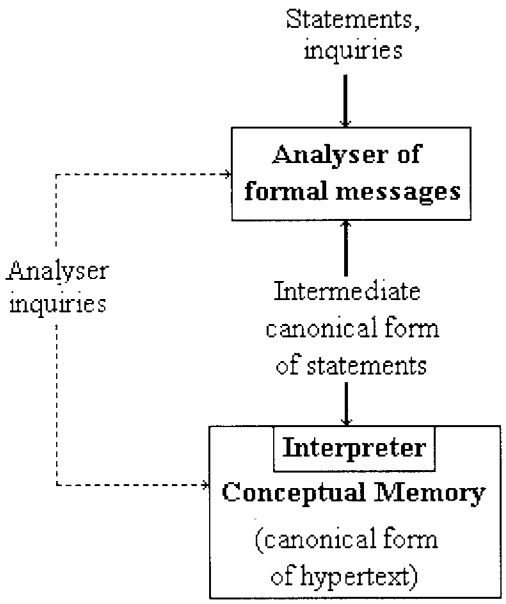

Fig. 2

Hypertext is formed in the conceptual memory from the formal messages extracted from the catalogue cards. Above we considered the external (input) format of the messages. The scheme of treatment of a message written in a catalogue card is shown in *fig. 2*. The input information goes to the analyser input, which tests its syntax and transforms it into the intermediate *canonical form*.

The result of analysis is treated by the interpreter integrated into the conceptual memory. It is convenient to consider the input information for the interpreter as a continuous flow of statements and inquiries. A statement points to the interpreter to include its content into the conceptual memory hypertext as supplementing information. An inquiry points to the interpreter to examine whether the inquiry content true and to complete it by examples from the conceptual memory.

The canonical form of messages is obtained by transformation of input statements into the sets of terminal statements. Some of these transformations were considered in the preceding section, another will be given below. Note that the intermediate canonical form saves the results of analysis in the terminal statements keeping unchanged external notations to an object. Unlike this, in the conceptual memory the external names inherent in the objects are transformed into unique internal records. To them are added relations which define the correspondence between the external and the internal notations. The defining of such relations is one of principal tasks of the interpreter.

The following constructions of terminal statements are allowed:

a) simple terminal: t1 : t2 is concretisation, t1 (t2) is attribute, t1 (t2. t3) is the attribute value, t1 = t2 is equal, t1 ≠ t2 is not equal, and 't1' := t2 denotes an object;

b) conceptualisation: t1 {t2}, t1 {⟨simple terminal⟩}, that is, the conceptualisation contains either an object, or a simple terminal construction;

c) link: t1 {t2, t3}.

Here t1, t2, t3 labels the places which in the construction examples are replaced either by internal notations to the object, or by variables.

When the internal notation to the object is essential, we will record it as ˜⟨notation⟩.

A few words concerning the canonical form structure. The message card being a conceptualisation is simultaneously a "moveable" unit of the hypertext. Records of all other conceptualisations are limited by the card in which they are recorded (regardless of information related to the objects defined as the general).

EXAMPLE 5. Suppose the formal message of the card #C contains the conceptualisation

{{C1 : C2 (C3. (C4 : C5))} ⇔ {C6 : C4}};

The canonical form of this message in the conceptual memory will be written as:

\`#C\` := ˜#c; ˜#c {\`C1\` := ˜c1}; ˜#c {\`C2\` := ˜c2}; ...;
˜#c {\`C6\` := ˜c6}; ˜#c {\`⇔\` := ˜c7}; ˜c7 {˜c8, ˜c9};
˜c8 {˜c1 : ˜c2}; ˜c8 {˜c2 (˜c3. ˜c5)}; ˜c8 {˜c4 : ˜c5};
˜c9 {˜c6 : ˜c4}.

Semantemes and guides are represented in the canonical form by the corresponding conceptualisations, which terminal elements (in the defining part of a semanteme and defined part of a guide) are labelled to indicate belonging to a guide or to a semanteme.

5. Conceptual memory

The conceptual memory is a triad

$$\mathbf{CM} = \{\mathbf{G}, \mathbf{F}, \mathbf{I}\},$$

where:

G denotes sets of terminal statements in canonical form, semantemes, and guide bar, forming together the hypertext general information,

F denotes the hypertext fragments (messages and conceptualisations),

I is the interpreter, represented by a set of functors.

The interpreter functors transform statements inputted into the conceptual memory into allowed values of parameters of the predicates of one of three types:

T-predicate points to relations between the individual objects,

C-predicates points to belonging of a **T**-predicate to a conceptualisation,

PC-predicate scans over conceptualisations.

The predicates are saved in the form of terminal statements. They assign "true" or "false" to a set of values of their own parameters. The predicate is true if the conceptual memory contains a given set of parameters.

T-predicates:

`⟨name⟩` := !X means that the !X variable may take as its value all objects with a defined external name; to each of external names its own predicate is assigned.

!`X` := ˜e means all external names of a given ˜e object; to each of the ˜e objects its own predicate is assigned. The latter is supported for the case when the predicates defined by structures are constructed for each of the objects designated as ˜e in the structures.

˜e : !X is concretisation of the ˜e object.

!X : ˜e means the objects concretized by ˜e.

˜e (!X) means attributes of a given object. !X(˜e) means all object with a given attribute.

˜e (!X. !Y) is combination of ⟨attribute⟩.⟨value⟩ for a given object.

C-predicates:

˜e {!X}, ˜e {!X: !Y}, ˜e {!X(!Y)}, ˜e {!X(!Y.!Z)}, ˜e {!X, !Y}

The essence is clear from the records.

PC-predicates:

!X {˜e }, !X {˜e 1 : ˜e 2}, !X {˜e 1 (˜e 2)}, !X {˜e 1 (˜e 2. ˜e 3}, !X {˜e 1, ˜e 2}

mean conceptualisations containing given object or relation.

The general information **G** of the hypertext is represented by an ensemble of **T**- and **PC**-predicates in the current state of conceptual memory, the **F**-fragments are represented by **C**-predicates.

The aim of the search performed by the interpreter has canonical form and is represented by a set ˜Q $(t_1; ...; t_k)$, where t_i $(1 \leqslant i \leqslant k)$ is either a predicate or a constant in a form of terminal element. An inquiry is considered as a conjunction of predicates, and equality of the latter to true or false is determined from the conceptual memory in the current state. The target of the search is changed every time when upon calculation of the next t_i predicate it obtains the value labelled as belonging to a guide or to a semanteme. In such a situation, the interpreter interrupts calculations initialised by current target, saves the state of the search of this target to be able return to the search in future, and constructs a new target.

Suppose ˜R $\{r_1; ...; r_m\} \Leftarrow$ ˜D $[d_1; ...; d_n]$ is a guide to which belongs t_i predicate, that is, in the ˜R set there is the r_j coinciding with the t_i within precision up to designation of variables. The use of a guide in correspondence with a known algorithm (see, e. g., [10]) precedes by a procedure of formation of a call to the guide of selecting the t_j elements $(i \leqslant j \leqslant k)$ in ˜Q, which current values will be obtained as a result of application of ˜R guide. While performing this procedure, the interpreter makes the following:

a) unitises variable in the ˜Q and ˜R sets and changes variables by values if the latter are known [8],

b) labels all t_j in ˜Q as capable to be calculated by the ˜R guide if there is r_w $(1 \leqslant w \leqslant m)$ in ˜R, with the internal name identical t_j.

As a new target is selected one of disjunctions of ˜D set not yet calculated.

For the semanteme ˜S $(s_1, ..., s_m)$:= ˜D $[d_1, ..., d_n]$ the procedure of formation of call is analogous, except selection of t_i is performed over the ˜D set, and that the new target is a ˜S set.

Conclusion

We are sure that integration of hypertext and means for relevant description of its fragments in a common conceptual memory is appropriate for the natural representation of analytical work with texts.

In this communication we paid attention mainly to the description of the formal tools for the encoding a text essence and the functional memory operating. Out of this framework remained important problems of automatic (that is, according to the guide bar) formation of the hypertext elements and treatment of nonformal messages (analysis of NL-texts and collection of dictionaries). The problem of automatic formation of pathways in the hypertext undoubtedly may be solved. The questions connected with the NL-texts analysis, extraction of formal messages from the latter, form a body of the following step of this research.

The approach to formal description of a text essence employed in this work is based on the use of clausal logic form. The constructions "concretisation-attribute" included into the language to a great extent are treated as form of syntax representation of relations between objects. Consequently, the structure of relations between objects described by these constructions is not rigid and may be supplemented by new elements of description.

Notes

1. G. V. Lezin, K. K. Boiarskiĭ, A. I. Popova, "Sistematizatsiia informatsii: semanticheskoe kodirovanie tekstov" ("Systematisation of information: semantic coding of texts"), *Trudy mezhdunarodnogo seminara Dialog'96 po komp'iuternoĭ lingvistike i eë prilozheniiam* (Moscow, 1996), pp. 131—6; K. K. Boiarskiĭ, G. V. Lezin, A. I. Popova, V. V. Sokol'skaia, "Sistema predstavleniia znaniĭ MAZE: ėlektronnaia kartoteka" ("A system MAZE for representation of knowledge: electronic card catalogue"), *Informatsionnye tekhnologii v gumanitarnykh i obshchestvennykh naukakh. Sistema MAZE: predstavlenie znaniĭ v gumanitarnykh issledovaniiakh* (St. Petersburg, 1995), pp. 13—22.

2. V. V. Martynov, *Universal'nyĭ semanticheskiĭ kod: USK-4. Preprint* (Universal semantic code: USC-4. Preprint) (Minsk, 1988).

3. R. Shenk, *Obrabotka kontseptual'noĭ informatsii* (Conceptual Information Processing) (Moscow, 1980). The publication represents the Russian translation of R. C. Shank's work published in Amsterdam—Oxford in 1975.

4. Boiarskiĭ, Lezin, Popova, Sokol'skaia, *op. cit.*

5. V. A. Iakobson, "Komp'iuternaia assiriologiia (k postanovke problemy)" ("Computer Assyriology: the problem definition"), *Informatsionnye tekhnologii v gumanitarnykh i obshchestvennykh naukakh*, ed. B. L. Ovsiyevich, fasc. 3 (St. Petersburg, 1996), pp. 3—9.

6. Boiarskiĭ, Lezin, Popova, Sokol'skaia, *op. cit.*

7. E. A. Kanevskiĭ, E. N. Klimenko, "Slovar' kak sredstvo analiza teksta" ("A dictionary as a tool for the text analysis"), *Informatsionnye tekhnologii v gumanitarnykh i obshchestvennykh naukakh*, ed. B. L. Ovsiyevich, fasc. 3, pp. 28—34.

8. Shenk, *op. cit.*

9. I. A. Mel'chuk, *Opyt teorii lingvisticheskikh modeleĭ "smysl ⇔ tekst". Semantika, sintaksis* (Experience in the Theory of the "Essence ⇔ Text" Linguistic Models. Semantics, Syntax) (Moscow, 1974).

10. M. Branokhe, "Upravlenie pamiat'iu v realizatsiiakh Prologa" ("The memory management of PROLOG Implementations"), *Logicheskoe programmirovanie*, ed. V. N. Agafonov (Moscow, 1988). The publication represents the Russian translation of M. Bruynooghe's writing edited in *APIC Studies in Data Processing*, vol. 16, eds. K. L. Clark and S.-A. Tarnlund (London, 1982).

Illustrations

Fig. 1. A card example.
Fig. 2. A scheme of messages treatment.

RUSSIAN ACADEMY OF SCIENCES
THE INSTITUTE OF ORIENTAL STUDIES
ST.PETERSBURG BRANCH

Manuscripta Orientalia

International Journal for Oriental Manuscript Research

Vol. 3 No. 2 June 1997

THESA
St.Petersburg-Helsinki

CONTENTS

Front cover:

"A Ship Among the Blocks of Ice", a colour drawing from the book 2 of the manuscript *Kankai Ibun* preserved in the collection of the St. Petersburg Branch of the Institute of Oriental Studies (C 191), fol. 14a, 14.0 × 20.5 cm.

Back cover:

"Theatre in the Capital of the Russian Empire", a colour drawing from the book 11 of the manuscript *Kankai Ibun* preserved in the collection of the St. Petersburg Branch of the Institute of Oriental Studies (C 191), fols. 11b—12a, 32.5 × 26.5 cm.