KYBERNETIKA — VOLUME 14 (1978), NUMBER 5

Theory of Types and Data Description

PAVEL MATERNA

The intensional character of natural languages is contrasted with the extensional character of artificial languages. A modification of Church's simple theory of types is shown to be able to serve as a tool for analysis of both the natural languages and those artificial languages that are used when data (e.g., in data bases) are to be fixed and retrieved.

INTRODUCTION

The theories of data structures suffer (at least at the logical level) from numerous ambiguities, vaguenesses, etc. The purpose of the present paper is, firstly, a clarification of some notions that are frequently used in the area mentioned above. (The notions in question are those of a class, of a property, of a relation, of an attribute.) Secondly, a modified version of Church's simple theory of types is shown to be a good tool for dealing with entities relevant for data description. Thirdly, natural languages are semantically confronted with artificial languages and this confrontation is shown to be useful for the theory of data description languages.

Remark. Where the terms "set" or "class" are used in the paper they are conceived of in their intuitive sense. This means that no specific conception of sets is presupposed. Therefore, no existing distinctions between various set theories are relevant for the following considerations. The more neutral term "collection" (cf. D. 2) might perhaps better serve our purposes.

1. SETS (CLASSES) AND PROPERTIES

E. Stenius in [9] calls our attention to misusing the term "set" (or "class"). The pack of wolves is a bad example of a set, he says, because "what makes a pack of wolves into a 'whole' is precisely the fact that this whole is not completely determined by the wolves it contains".

Well, the notion of a set is, of course, not the same notion as the notion of a structured whole. However, the former is frequently being confused not only with the latter but – even more frequently – with the notion of a property.

Theoretically, everyone "feels" that there is a difference between the two notions. Nevertheless, the extensionalist traditions of formal logic survive in the form of the opinion that the difference in question can be neglected because properties are intensional entities (the nearly full unanimity concerning this true part of the implicit argumentation is promising) and intensional entities are to be taken into consideration in some ("oblique") contexts only (even Church and Montague who are considered to be "intensionalists" share this false part of the argumentation).

From the above point of view a logical analysis of our language may represent properties by classes. Therefore, the 1st order predicate logic is being considered to be an adequate – or at least a "nearly adequate" – tool of such a logical analysis.

As has been shown in [11] this extensionalist view-point is misleading. The ideas from [11] (as well as from other papers by Tichý) are not only critical but also positive in that they enable to make logical analyses that systematically take into account intensional character of the entities that we talk about. Therefore, these ideas are the main source of the present considerations.

Before we introduce the formal tool of intensional analysis we shall informally explain the intuitive distinction between classes and properties and add some further remarks.

We often speak about "classes" of whales, houses, stars, etc. Whereas, however, there is, clearly, such an entity as the class of prime numbers, we could hardly claim that there is something like "the class of whales".

Firstly, whether a number is or is not a prime number depends exactly on what number it is; the only fact that this number is such and such number fully determines its being or not being a member of the class of prime numbers. On the other hand, let A be an individual; A's being or not being a whale depends not on its being just A; the mere fact of being A does not fully determine A's being or not being a member of "the class of whales". Thus the class of prime numbers is what is meant by "class" in mathematics and it can be conceived of as a function from natural numbers to truth-values. On the other hand, "the class of whales" is no class; it is no function from individuals to truth-values, because we cannot neglect that what does depend upon whether some individual is or is not a whale is "the state of world". A state of world may be conceived of as some distribution of some elementary features over the objects whose features they might be. Any such conceivable distribution is what is called in the contemporary semantic literature "possible world". Thus the first approximation of our explication of the term "property" (for to be a whale, a house, a star, etc. are examples of properties) would be as follows:

A1. A property is a function from the set of possible worlds into the set of classes.

According to A1., to be a whale in a world *W* is to be a member of the class which is the value the property 'whaleness' in *W*.

Notice that this conception is in good accordance with our intuitions. To know whether an individual is a whale we must apply empirical methods, i.e., we have to observe this individual and apply some criteria. Thus some amount of our knowledge of the world is necessary for our claim that the given individual is (is not) a whale. No such empirical knowledge and no empirical criteria are necessary for our answering the question whether the given number is a prime number.

Remark. Calling "trivial property" any property whose value is the same in all the possible worlds one could say, of course, that to be a prime number is a property, too, namely a trivial property.

Secondly, an important correction of A1. is necessary if one wants to take into account time. In this case it does not hold that the values of properties in possible worlds are classes. Take, e.g., the whales in our actual world. It is clear that the number of whales in the actual world is variable, that whales die and new whales are born. This means, however, that even in a given world the values of a property is, generally speaking, not a class, because a class has "always" the same members. Thus we can formulate the second approximation:

A2. A property is a function from the set of possible worlds into the set of functions from time moments to classes.

According to A2., to be a whale in a world *W* at the time moment *I* is to be a member of the class which is at the time moment *I* the value of the property 'whaleness' in *W*.

Unfortunately, neither A2. is a satisfactory explication of the term "property". It is important to see that whereas, e.g., the value of 'whaleness' in *W* at the time moment *I* is a class of individuals, the value of, say, 'being a nation' is (in a world *W*, of course, at the time moment *I*) a class of classes of individuals and the value of 'being believed by A. Einstein' is a class of propositions. The correction of our approximations that will take into account these distinctions will be made as soon as we cease being as informal as we are now.

2. A GENERALIZATION. INTENSIONS

Neglecting the important difference between classes and properties the 1st order predicate logic represents properties by one place predicates. (Even the authors who explicitly stress this difference "support" their views by means of the 1st order calculus + comments made in natural language – cf. Hintikka.) An analogous simplification due to the extensionalist character of the 1st order logic may be seen in the fact that the n-placed predicates are believed to represent n-ary relations, whereas they represent only what is usually called "relations-in-extension".

The difference between relations-in-extension and relations-in-intension corresponds with the difference between classes and properties; this is easy to see because classes are n-ary relations-in-extension and properties are n-ary relations-in-intension for $n = 1$.

Thus whereas $>$ is a good example of a relation-in-extension (on the set of, say, rational numbers), the relation of being older (than) is an example of a relation-in-intension (on a set of individuals). It is, therefore, a function that takes any possible world to at most one function from time moments to relations-in-extension (of individuals), i.e., to sets of ordered pairs (of individuals).

One of the most unpopular consequents of the extensionalist representation of properties and relations is that a (closed) sentence of the 1st order predicate logic inevitably denotes a truth-value. To say about a one place predicate, e.g., P, that it can represent a property ('whaleness') by reducing it to a class ('the class of whales') may at first sight seem not wholly unacceptable. However, nobody likes saying that a sentence names (is about) a truth-value.

According to the conception just referred to a sentence names a proposition. Not going into details one can say that a proposition is a function from possible worlds into the set of functions from time moments to truth-values. (Some propositions are simply functions from possible worlds to truth-values.)

Thus the sentence

The present American President is J. Carter

denotes the function that takes in the actual world as its value the function that takes the time moments, say, between 1. 4. 1977 and 3. 4. 1977 to TRUTH, the time moments before 1. 1. 1977 to FALSITY, etc. (Notice that this function is undefined, e.g., for the worlds where the U.S.A. are a monarchy or – in the actual world – for the time moments at which the U.S.A. did not exist.)

In logical semantics there is a term which covers any functions from the set of possible worlds. We define:

D1. An *intension* is a function the domain of which is the set of possible worlds.

3. A FORMAL TOOL: THEORY OF TYPES

Clarifying and making more precise the previous considerations presupposes using a logical apparatus that would be adequate for the present conception: It is by now obvious that the 1st order predicate logic cannot be such an apparatus. We accept (cf. [11], ([6]) that a modified version of Church's simple theory of types (TT, see [2]) is what we want to have. Therefore we define:

D2. A *base* is any set of mutually disjoint non-empty collections.

D3. 1. Any member of the base B is a *type over* B;

2. Let $\alpha, \beta_1, \ldots, \beta_n$ be *types over* B. Then $\alpha(\beta_1, \ldots, \beta_n)$, i.e., the set of (total and partial) functions from $\beta_1 \times \ldots \times \beta_n$ into α, is a *type over* B;

3. Nothing other is a *type over* B.

In what follows we shall omit the phrase "over B" presupposing that a base B is already given.

D4. Let α be a type. Any member of α is an *α-object*.

Example. Let

$$B = \{\{A, B, C\}, \ \{1, 2\}, \ \{a, b, c, d\}\}.$$

Let the type of $\{A, B, C\}$ be ξ, of $\{1, 2\}$ – φ, of $\{a, b, c, d\}$ – χ.
Then the type $\xi(\varphi, \varphi)$ is the set of 256 functions:

	f_1	f_2	f_3	f_4	$f_5 \cdots f_{64}$		$f_{65} \cdots f_{255}$		f_{256}
$\langle 1, 1\rangle$	A	A	A	A	A	A	B	–	–
$\langle 1, 2\rangle$	A	A	A	A	A	–	A	–	–
$\langle 2, 1\rangle$	A	A	A	A	B	–	A	–	–
$\langle 2, 2\rangle$	A	B	C	–	A	–	A	C	–

An example of an $\xi(\chi)(\varphi, \varphi)$-object:
Let g_i, g_j, g_k be the following functions:

	g_i	g_j	g_k
a	A	A	C
b	A	–	–
c	–	B	B
d	–	C	A

(Clearly, g_i, g_j, g_k are $\xi(\chi)$-objects.)
Then the function h:

	h
$\langle 1, 1\rangle$	g_i
$\langle 1, 2\rangle$	g_j
$\langle 2, 1\rangle$	g_k
$\langle 2, 2\rangle$	–

is such an example.

After having introduced the notion of a *variable* (an α-variable being an abstract representative of α-objects) one could inductively define the notion of a *construction*. (Atomic constructions are objects and variables, molecular constructions are applications and abstractions.) This definition, however, has been published in [5] and we prefer to call the reader's attention to that paper, where also the relevant bibliography is given. Here we adduce only some examples: (Cf. the preceding example!)

Let x, y be a φ-variable and a χ-variable, respectively.

Examples of atomic constructions:

$$x, y, g_i, h, f_5, \ldots .$$

Examples of applications:

the application of h to $\langle 1, 2 \rangle$: this is a $\xi(\chi)$-construction which constructs g_j. We write

$$h(1, 2) ;$$

the application of h to $\langle 2, 2 \rangle$: this is an "improper" $\xi(\chi)$-construction which constructs no object (see the table for h!).

Example of an $(x, y\text{-})$ abstraction:

$$\lambda xy(h(x, 1)(y)) .$$

This is an $\xi(\varphi, \chi)$-construction which constructs the following function, say k:

	k
$\langle 1, a \rangle$	A
$\langle 1, b \rangle$	A
$\langle 1, c \rangle$	–
$\langle 1, d \rangle$	–
$\langle 2, a \rangle$	C
$\langle 2, b \rangle$	–
$\langle 2, c \rangle$	B
$\langle 2, d \rangle$	A

To apply TT to mathematics, Church needed a very simple base. His base consists of two types: the type o is the set of truth-values, the type ι is the set of (natural) numbers. For applying TT to the logical analysis of natural languages it is necessary to start from a more complex base. Before introducing such a base we shall formulate some preliminary intuitions.

A language user has at his disposal what has been called in [11] intensional basis This is a collection of elementary empirical features ("traits") the distribution of which characterizes the world that has to be described by the language. To know the actual distribution (of these traits) within some time interval I is the same as to be omniscient with respect to I. The actual distribution is a matter of empirical research and, therefore, is not given a priori. What is, of course, given a priori with respect to the intensional basis, is the set of all conceivable (thinkable, possible) distributions. This set we call the *logical space* (with respect to the language in question) and its members are the *(possible) worlds.*

If we do not take into account some complications connected with actual natural languages we may suppose that there is a set – let us call it the *universe* – the members of which are elementary objects of our predications. The members of the universe are called *individuals* and they are given a priori (with respect to the given language).

Now we can introduce a base that enables rather deep logical analyses of natural language expressions.

D5. A *minimal natural-language-base (MNL-base)* consists of types o, ι, τ, ω, where $o = \{T, F\}$ is the set of truth-values,

ι = the universe,

τ = the set of time moments (may be identified with the set of real numbers),

ω = the logical space.

D5. makes it possible to define more exactly the notion of an intension:

D6. Let α be a type. Any $\alpha(\omega)$-object is an *intension over* α. Any $\alpha(\tau)(\omega)$-object is a *t-intension over* α.

Thus a property of individuals is a (t-)intension over $o(\iota)$, i.e., an $o(\iota)(\omega)$- or $o(\iota)(\tau)(\omega)$-object, a property of classes of individuals is a (t-)intension over $o(o(\iota))$, i.e., an $o(o(\iota))(\omega)$- or $o(o(\iota))(\tau)(\omega)$-object; generally speaking, a property of α-objects is a (t-)intension over $o(\alpha)$, i.e., an $o(\alpha)(\omega)$- or $o(\alpha)(\tau)(\omega)$-object; a proposition is a (t-)intension over o, i.e., an $o(\omega)$- or $o(\tau)(\omega)$-object; an n-ary relation-in-intension between β_1-, ..., β_n-objects is a (t-)intension over $o(\beta_1, ..., \beta_n)$, i.e., an $o(\beta_1, ..., \beta_n)(\omega)$- or $o(\beta_1, ..., \beta_n)(\tau)(\omega)$-object, etc.

The extensions are just those objects that are not intensions. (An alternative terminological convention may be based on the inductive definition of the notion "an intension of n-th degree over α" where extensions would be conceived of as intensions of degree 0.)

For the sake of brevity we now shall not expound the theoretical principles of the logical analysis of natural language expressions which is based on TT applied to a MNL-base. (See, however, some elements of such an exposition in [11] and in [5].) Instead we shall adduce some examples confronting a natural language (English) expression with the construction that is – according to the present conception – the logical analysis of that expression.

Example One. One can see that adjectives usually denote such functions that in any world and any time moment associate with a property of α-objects a class of

 α-objects. Take the English expression

(1) red house

"Red" obviously denotes an $o(\iota)(o(\iota)(\tau)(\omega))(\tau)(\omega)$-object, say, R; "house" is a name of an ι-property, i.e., of an $o(\iota)(\tau)(\omega)$-object, say H. Therefore, (1) denotes the $o(\iota)(\tau)(\omega)$-object which is constructed by

(1′) $$\lambda w(\lambda t(R(w)(t)(H))),$$

w, t variables of types ω, τ, respectively.

Equivalently, one can say that (1) *expresses* the construction (1′).

Applying (1′) to some world W and time moment I one gets

$$R(W)(I)(H),$$

which constructs a class of individuals, namely of what one could talk about as "red houses at time moment I in the world W".

Example Two. Let A, B, C be three individuals labeled in the given fragment of English as Adam, Bob, Charles, respectively. Take the sentence

(2) *Adam and Bob are older than Charles.*

There are constructible some rules that associate grammatically correct expressions of a language with the constructions expressed by them. According to a possible part of such a rule (2) is transformable to

Adam is older than Charles and Bob is older than Charles.

The atoms that are the builiding stones of the desired construction are

$$A/\iota,\quad B/\iota,\quad C/\iota,\quad \wedge(\text{"and"})/o(o,o),\quad O(\text{"older"})/o(\iota,\iota)(\tau)(\omega).$$

(2) expresses the following $o(\tau)(\omega)$-construction:

(2′) $$\lambda w(\lambda t(\wedge(O(w)(t)(A,C),\quad O(w)(t)(B,C)))).$$

Example Three.

(3) *Some employees of the company X own a house.*

Let us suppose that "employee of the company X" denotes a property $E/o(\iota)(\tau)(\omega)$. "own" clearly names a relation $Ow/o(\iota,\iota)(\tau)(\omega)$ and "house" denotes the property from Example One.

(3) is transformable to

There are x and y such that x is an employee of the company X,

y is a house and x owns y.

The expressions like "there are", "some" etc. denote functions, say, Σ_α of the types $o(o(\alpha))$, where α is a type. In our case $\alpha = \iota$. Thus one gets

$$(3') \qquad \lambda w(\lambda t(\Sigma_\iota(\lambda x(\Sigma_\iota(\lambda y(\wedge(\wedge(E(w)(t)(x),\ H(w)(t)(y)),\ Ow(w)(t)(x, y)))))))$$

from which one gets after applying some abbreviation rules the more readable version

$$\lambda w(\lambda t(\exists xy(E(w)(t)(x) \wedge H(w)(t)(y) \wedge Ow(w)(t)(x, y)))) .$$

Many interesting problems from the area of logical analyses of language expressions, that cannot be solved within the framework of the 1st order predicate logic, are solvable within the framework of the apparatus just referred to. The exposition of these results is, however, not the aim pursued by the present paper.

4. ARTIFICIAL LANGUAGES

What is – among other things – characteristic of natural languages is that they are associated with a MNL-base and that this MNL-base contains the logical space. This logical space ensures it to be possible for a natural language to talk about intensions. The intensions, however, would be unnecessary entities from the viewpoint of a language if they were only trivial functions, i.e., if the value of any of them were the same in all possible worlds. Why one has to take into account the logical space is because, typically, one talks about intensions that are not trivial functions. Any empirical claim concerns some such intensions. If one says, e.g.,

(4) *The color of my cigarette is white,*

there are five intensions one talks about: a) the color; b) my cigarette; c) the color of my cigarette; d) white; e) that the color of my cigarette is white. The first of them is a function that could be called an attribute (see later), the second is what is called "individual concept" (an entity of the type $\iota(\tau)(\omega)$); the third and fourth are properties of individuals and the fifth is a proposition. To verify (4) presupposes that one acquires some amount of the knowledge of the world, i.e., that one applies a) to the actual world (and to the time moment in question, of course), that one does the same for b) and d), confronts the result of applying the first result to the second (thus one gets what is actually the color of the given cigarette) with the third and obtains in this way the truth-value of e) at the given time moment in the actual world. Notice, however, firstly, that this result is not given a priori and that, secondly, any user of English understands (4) independently of knowing the truth-value of e) in the actual world.

The natural languages are thus our tools for dealing with empirical matters.

The term "language" is, on the other hand, used in such phrases as "the language of mathematics", "formal languages", "programming languages", "data description languages", "users' languages" etc. The term one could use to cover what is meant by the phrases of this sort is "artificial language".

We would like to stress here that one of the fundamental features distinguishing the artificial languages from the natural languages is that the former need not deal with intensions. (We do not wish to comment some attempts to apply intensional logic to programming languages – see [3].)

We turn now our attention to one kind of artificial language, namely to the data description languages (DDL).

5. SEMANTICS OF DDL

Let us consider Ahrens-Walter's definition of data (in [1]):

"Daten sind allgemein Buchstaben, Ziffern oder Symbole, die sich auf ein Objekt, eine Realität, eine Bedingung oder andere Faktoren beziehen oder solche beschreiben." (p. 14).

We shall not try to analyze and correct some vague terms contained in this definition. What is important for us is that any attempt to build up a semantics for DDL has obviously to start from the fact that the "data" in a data base (DB) concern some entities that are of "ontological" character, i.e., that are outside the DB in question.

It is obvious, also, that a semantic description of data structures should be based on a semantic analysis of the way how to compose complex data from elementary data. It seems that TT is a good logical tool for such a description. At the same time, TT might serve us as a sort of a link between DDL and natural languages.

Let us observe a simplified example.

Suppose that one would like to have at his disposal a DB from which he could extract informations concerning a class of organizations, their hierarchy, their employees and the birth data and skills of these employees.

Observe the way of acquiring the desired data. Using the expressions of the given natural language ("organization", "employee", "birthday", "birth-place", "skill") and applying them to the actual world at the given time moment (and, of course, to a fragment of the given universe) one gets some values of the *t*-intensions in question in the actual world at the given time moment. These values are what the data are based upon.

Let be given a MNL-base B. Let the type of entity O denoted by "organization" be $o(\alpha)(\tau)(\omega)$ (we are here not interested in the concrete form of the type α). The entity P denoted by "place" be of the type $o(\beta)(\tau)(\omega)$, the entity S denoted by "skill" be of the type $o(\gamma)(\tau)(\omega)$. Then "the organization superior to", "the sub-organizations of", "the employee of", "the birthday of", "the birth-place of" and "the skills of" denote what is usually named "attributes". As this term is frequently used and repeatedly defined in the theories of data structures we shall do our best to clarify it.

Narasimhan in [7] stresses the importance of the knowledge – in acquiring "language behaviour" – of "(3) which attributes are applicable to which Objects" and of "(4) which properties are values of which attributes". What he means by (4). is obvious from his attempt to define attribute (p. 275): according to him "attributes are (partially) computable functions defined over the domain of Objects and assume values from well-defined ranges. The values of attributes are called properties". Narasimhan's terminology is not intuitive enough; he himself says, e.g., that length is an attribute; one would, however, hardly call values of this attribute properties. Nevertheless, one can retain the essential point of the above definition (which is in accordance with many other proposals of an explication of the term "attribute" and with the practice of using it) and modify it so that it fitted to our semantic conception.

D7. Let δ, η, ξ be types, $\xi \neq o$, let A be an $o(\eta)(\tau)(\omega)$-object, A' – an $o(\delta)$-object, B – a) an $o(\xi)$-object, b) an $o(\xi)(\tau)(\omega)$-object. A *B-attribute over A (over A')* is any $\xi(\eta)(\tau)(\omega)$-object (any $\xi(\delta)(\tau)(\omega)$-object) such that associates in any world W and time moment I any member of $A(W)(I)$ (of A') with at most one member a) of B, b) of $B(W)(I)$, and at least one member of $A(W)(I)$ (of A') with just one member a) of B, b) of $B(W)(I)$.

For example, "cigarette" denotes a property of individuals, i.e., an $o(\iota)(\tau)(\omega)$-object. Let B be the set of colors, i.e., an $o(o(\iota)(\tau)(\omega))$-object. "Color of" denotes a function associating every world and every time moment with a function that takes individuals to colors, i.e., an $o(\iota)(\tau)(\omega)(\iota)(\tau)(\omega)$-object. One can easily see that this object is – in accordance with D7. – a B-attribute over cigarettes (but it is also a B-attribute over tables, stars, eyes, etc.).

In our example, "the organization superior to" denotes an O-attribute, say, $Os/\alpha(\alpha)(\tau)(\omega)$, which in the world W at the time moment I associates any organization with at most one organization, namely that one, which is superior to the given organization. Similarly, where Og is an entity of the type $o(o(\alpha))(\tau)(\omega)$ such that associates any world W and time moment I with the class of classes of entities that are at I in W organizations, "the suborganizations of" denotes the Og-attribute So (over organizations), thus we have $So/o(\alpha)(\alpha)(\tau)(\omega)$. Similarly:

"the employee of" denotes the attribute $Em/o(\iota)(\alpha)(\tau)(\omega)$,

"the birthday of" denotes the attribute $Bd/o(\tau)(\iota)(\tau)(\omega)$, "the birth-place of" denotes the attribute $Bp/\beta(\iota)(\tau)(\omega)$,

"the skill(s) of" denotes the attribute $Sk/o(\gamma)(\iota)(\tau)(\omega)$.

Bd, e.g., associates a world *W* and time moment *I* with a function that maps the set of individuals into the set of time intervals (the birthday is a time interval).

Now, imagine that after having applied *t*-intensions denoted by the above expressions to the given (actual) world and to the given time moment one has completed *the stage of applying natural language to acquiring data.*

The stage of using artificial language starts with associating every original natural language expression from the preceding stage with a name (we shall say: *a label*) of its value in the actual world and the given time moment, say *I*. The following table makes this clear on our example.

natural language expression	the corresponding intension + type	the label of the value + type
organization	$O/o(\alpha)(\tau)(\omega)$	$ORG/o(\alpha)$
place	$P/o(\beta)(\tau)(\omega)$	$PL/o(\beta)$
skill	$S/o(\gamma)(\tau)(\omega)$	$SK/o(\gamma)$
the organization superior to	$Os/\alpha(\alpha)(\tau)(\omega)$	$SUPORG/\alpha(\alpha)$
the suborganization of	$So/o(\alpha)(\alpha)(\tau)(\omega)$	$SUBORG/o(\alpha)(\alpha)$
the employee of	$Em/o(\iota)(\alpha)(\tau)(\omega)$	$EMPE/o(\iota)(\alpha)$
the birthday of	$Bd/o(\tau)(\iota)(\tau)(\omega)$	$BIRTD/o(\tau)(\iota)$
the birth-place of	$Bp/\beta(\iota)(\tau)(\omega)$	$BIRTP/\beta(\iota)$
the skill(s) of	$Sk/o(\gamma)(\iota)(\tau)(\omega)$	$SKILL/o(\gamma)(\iota)$

From the types of labels one can see that a) these labels denote extensions, b) the labels one has got from attributes denote functions. Furthermore, *Os*, *So*, *Em* are attributes over *O*, *Bd*, *Bp* and *Sk* are attributes over the given fragment of the universe *U*. Finally, the fact that the labels concern some time moment *I* could be marked by the subscript "*I*" at each label.

The labels denote results of our empirical investigation. These results might be represented by tables which associate i) any member of ORG, ii) any member of (the fragment of) *U* with the value of one of the functions denoted by labels. (These functions might be called "pseudoattributes".)

Let the members of ORG be $O_1, \ldots, O_k$ and the members of (the fragment of) U be $U_1, \ldots, U_m$. The time intervals let be given by dates, the members of PL let be $P_1, \ldots, P_t$, the members of SKILL let be $S_1, \ldots, S_y$. The tables just mentioned would have following forms:

SUPORG

O_1 O_{i_1}
O_2 O_{i_2}
$\vdots$ $\vdots$
O_j – (i.e., O_j has no organization superior to it.)
$\vdots$ $\vdots$
O_k O_{i_k}

SUBORG

O_1 $\{O_{j_1}, \ldots, O_{j_m}\}$
O_2 $\{O_{j'_1}, \ldots, O_{j'_n}\}$
$\vdots$ $\vdots$
O_p $\{\ \}$
$\vdots$ $\vdots$ (O_p, O_q are the lowest-level organizations.)
O_q $\{\ \}$
$\vdots$ $\vdots$
O_k $\{O_{j''_1}, \ldots, O_{j''_l}\}$

EMPE

O_1 $\{U_{k_1}, \ldots, U_{k_s}\}$
O_2 $\{U_{k'_1}, \ldots, U_{k'_v}\}$
$\vdots$ $\vdots$
O_k $\{U_{k''_1}, \ldots, U_{k''_w}\}$

BIRTD

U_1 21041930
$\vdots$ $\vdots$
U_m 02101927

BIRTP

U_1 P_{i_1}
$\vdots$ $\vdots$
U_m P_{i_m}

SKILLS

U_1 $\{S_{j_1}, \ldots, S_{j_a}\}$
$\vdots$ $\vdots$
U_m $\{S_{j'_1}, \ldots, S_{j'_b}\}$.

Now, data correspond – on the logical level – to sentences. Simple data from our example may correspond to sentences such as:

(5) U_{k_1} *is an employee of the organization* O_1 .

(6) U'_1*s birthday is April* 21*st*, 1930 .

(7) O_{j_1} *is a suborganization of* O_1 .

etc.

Complex data correspond to the sentences the basic components of which are the sentences corresponding to simple data. So, e.g., the following sentence corresponds to a complex datum:

(8) *The organizations whose some employees have skills*
S_{j_1} or S_{j_2} are O_1, O_3 and O_4.

One can now represent the sentences like (5) – (8) in an artificial language L_{DB} the atomic expressions of which are the labels of the values of the relevant natural language expressions in the actual world – as fixed in the given DB – at the time moment I:

(5′) $$\text{EMPE}\,(O_1)\,(U_{k_1}),$$

(6′) $$\text{BIRTD}\,(U_1) = 21041930 \text{ (an abbreviation of } = (\text{BIRTD}\,(U_1), 21041930)),$$

(7′) $$\text{SUBORG}\,(O_1)\,(O_{j_1}),$$

(8′) $$\lambda o(\exists x(\text{EMPE}\,(o)\,(x) \wedge (\text{SKILLS}\,(x)\,(S_{j_1}) \vee \text{SKILLS}\,(x)\,(S_{j_2})))) = \{O_1, O_3, O_4\}.$$

(o, x are variables of types α, ι, respectively.)

Let us confront L_{DB} a) with natural languages (NL), b) with the 1st order (predicate) languages (PL1L).

Ad a). There are two main features distinguishing L_{DB} from NL: i) NL-expressions denote intensions and express constructions, whereas L_{DB}-expressions are names of (i.e., they denote) constructions.

ii) Constructions expressed by NL-expressions construct mostly intensions, whereas constructions named by L_{DB}-expressions construct extensions.

Ad b). L_{DB} is a TT-based language and, therefore, two operations only are used by it: application and abstraction. Thus – unlike the typical instances of PL1L – L_{DB} considers, e.g., logical connectives to be names of some functions (the types of these functions are $o(o)$ for unary connectives and $o(o, o)$ for binary connectives). (Therefore, (8′) is also an abbreviation of a more cumbersome expression.)

Trying to represent, e.g., (8) by means of some PL1L one can immediately see the advantages of the TT-based languages. Besides, the L_{DB}-representation is not one constructed ad hoc; it is a clear logic-based representation.

It is obvious that such a set of tables as we have introduced can be represented in some of the ways frequently used in DB systems. One can see, e.g., that in a "record" there could be integrated all the data concerning one organization. From this view-point the values of SUBORG, EMPE, SKILLS can be considered to be repeating groups, etc. These details do not concern us in the present paper.

The updating activity does not change the types of the particular L_{DB}-expressions

What is changed only, are the values constructed by the pseudoattributes in question. Thus $SUPORG_I$ may differ from $SUPORG_{I'}$ for $I \neq I'$, but the type remains to be $\alpha(\alpha)$.*)

Considering the semantics of both the NL in question and L_{DB} in question to be based on TT enables us to see a natural link connecting a (fragment of) natural language with a L_{DB}. This may be useful when one wants to build up a query language for the users of DB (see below).

The possible importance of the type-theoretical point of view for those specialists who are engaged in designing data bases is, however, more fundamental. As has been convincingly shown by Bo Sundgren (see, e.g., [10]), many troubles with designing effective data bases are due to an unsatisfactory theoretical background. Sundgren's infological theory has been built up in order to make the character of a data base more conceivable (and in this way also more perspicuous for a user). To Sundgren's opinion it should be possible to start with a good theoretical ("infological") analysis of the data base concept and gradually to pass to the "datalogical" stage including the problems of implementation. The type-theoretical approach differs from the particular concrete features of Sundgren's theory but the idea is the same. It seems that the conception of a data base as a set of functions could make it possible to create in a rather easy way e.g. a query language where relevant types would be assigned to the key words and simultaneously to the algorithms that construct the objects denoted by these key words.

(Received November 22, 1977.)

REFERENCES

[1] Fr. Ahrens, H. Walter: Datenbanksysteme. Gruyter, Berlin—New York 1971.

[2] A. Church: A formulation of the simple theory of types. Journal of Symbolic Logic *5* (1940), 1, 56—68.

[3] T. M. V. Janssen, P. van Emde Boas: On the proper treatment of referencing, dereferencing and assignment. Stichting mathematisch centrum, ZW94/77.

[4] S. Kaneff, ed.: Picture language machines. Academic Press 1970.

[5] P. Materna, K. Pala: Theoretical framework for syntax and semantics. Celostátní konference o kybernetice, Praha 1976, Sborník prací, 233—253.

[6] R. Montague: Universal Grammar. Theoria *36* (1970), 373—398.

[7] R. Narasimhan: Natural Language Behaviour. In [4].

[8] A. Scheber, J. Šturc: Relačný model báz dát. Informačné systémy 1976/3, 281—294.

[9] E. Stenius: Sets. Synthese *27*/1, 2 (1974), 161—188.

[10] Bo Sundgren: An Infological Approach to Data Bases. URVAL Nr. 7, Stockholm 1973, 3—478.

[11] P. Tichý: An Approach to Intensional Analysis. Noûs *V* (1971), 3, 273—297.

Doc. PhDr. Pavel Materna, CSc., Projektový ústav dopravních a inženýrských staveb (Design Institute PÚDIS), Legerova 69, 110 00 Praha 1. Czechoslovakia.

*) From this view-point a DB cannot, of course, be conceived of as a "set of relations variable in time" (cf. [8]) — after all, no relation is "variable in time" — but as a function linking selected time moments with sets of functions.