Janusz Stanislaw Bień

TOWARDS COMPUTER SYSTEMS FOR CONVERSING IN POLISH

1. PHILOSOPHY OF THE MARYSIA PROJECT

1.1. *Natural language communication trends in computer software.*

The progress in computer hardware in recent years has been enormous. Computers are now extremely fast and relatively cheap, the capacity of their storage has also been multiplied. These factors influence both the range of computer applications and the complexity of software. Computers are now used directly not only by mathematicians, physicists and data processing departments, but also by scientific workers of almost all domains of knowledge (including philology, philosophy, archaeology, etc.), managers and even sometimes laymen such as patients in hospitals. On the other hand, the great computational power of existing hardware allows us to develop very sophisticated systems for solving complicated problems, in a fully automatic manner or by means of interaction with man. There is no reason to doubt this is a steady trend in the computer world. We have to realize now that it means that man-machine communication will become more and more crucial in computer usage. First, if we cannot make communication with computers easier, then the greater number of computer users requires the total cost of training to rise considerably. Secondly, even an excellent problem solver can be of no use if we do not develop the means for stating a problem correctly. The aim of research in proving the correctness of programs and automatic program synthesis is to solve the software crisis by making the work of programmers easier.

Acknowledgedment. The work described here has been done at the mathematical department of Warsaw University by the team consisting of S. Szpakowicz, W. Łukaszewicz and the author; in the early stage of development it was supervised by Prof. S. Waligòsski.

It is not yet clearly realized that any result in the domain may only shift the burden from expressing ideas in programming language to doing the same but in another formalism. The following should prove the above statement. Let us consider the man-machine interaction presented in the R. W. FLOYD (1971) paper and try to design a formalism for it. It will become evident that if such a formalism exists, then because of its complexity it is not easier to express the ideas in it than just to program the problem.

Our assumption is that the only long-term solution of these problems is man-machine communication in natural language. It is, of course, an old idea. This has appeared in the COBOL design, a questionnaire method of inquiries of men by computers and vice-versa, and in some question-answering and information retrieval systems, etc.. C. MEADOW (1970, p. 141) has stated the following: " the lure of natural language communication is with us, and we may expect to see a continuing trend towards its use, or its approximation, in all forms of man-machine communication ". We strongly believe this and this is the genesis of the project aiming at developing the MARYSIA Polish language conversational system.

1.2. *What does " conversational system " mean.*

When the idea of time sharing was brand new, every session with any time sharing system was called a " conversation with a computer ". It still happens that we meet the word " conversation " in this meaning, but it is better to distinguish interactive systems (and languages) and conversational systems. By the latter we mean a system which allows interaction in natural language, usually a limited language. This state ment requires some clarification. It can be understood in a broad sense as the following; every system you can communicate with in natural language is a conversational system. However, its narrow sense is more appealing. Let us consider for the moment the structure of a conversational system. I claim that for designing and debugging purposes such a large system is to be split into some modules with clearly established functions of modules and interfaces between them. One of the modules is to be a " brain " of the system, it determines the behavior of the system by controlling its slave modules. As a rule, for the purpose of portability and adaptability it should not have contact directly with the external environment. One (and often the only) method of interaction between the " brain " and the external world is to use a special conversational module. The module has as an input utterances of a natural language and translates them into a " brain " formalism or vice-versa (as proved by T. WINOGRAD, 1971, during the analysis of an utterance a feedback from the " brain " is desirable for efficiency). In most cases the " brain " can be just an already existing interactive system, in other cases the rest of the system may be especially developed for making full utilization of the module possibilities, e.g. automatic resolution of ambiguities during text preparation for statistical computations or preprocessing of utterances before semantic and pragmatic analysis in an artificial intelligence system. Such a module is fairly complex and relatively independent of other parts of the system it is embedded in; therefore we prefer to consider it as a separate system and to refer to it as a conversational system. The MARYSIA system is conversational in this narrower sense.

1.3. *The aim of the* MARYSIA *project.*

We consider the rich inflexion of the Polish language as the main difficulty in developing systems for man-machine communication in Polish. For example, it makes the questionnaire method inconvenient, because for psychological reasons we have to choose between two possibilities: either to allow only " yes "-" no " answers or to accept a considerable number of mismatches, caused by impermissible inflexional forms. It is also not possible to develop any more sophisticated language processing system for Polish without implementing (or simulating) algorithms of inflexional analysis and synthesis. Therefore the primary aim of the MARYSIA project was to break the barrier of inflexion. This means solving two problems. First, we had to design a formalism to talk about inflexion with sufficient precision. Secondly, we had to develop a general purpose system of practical use, with the ability to perform inflexional analysis and synthesis. These attributes of a system seem to be contradictory, but we found a way out. We have split the system into two parts with different functions. One part of it has to cover the morphological level of the language. This is an open ended part of the system, because there are only two restrictions on its adequacy: one is the computer storage available and the second is the necessity to describe the morphology by means of the notions designed by us for this purpose. It is important that the adequacy is not fixed at the moment of system generation but can be increased step by step, mainly by putting new items into the MARYSIA dictionaries. The second part of the system has to serve temporarily as a means for "jumping over" the higher levels of language, such as syntax and semantics, and eventually pragmatics. At the moment it is rather primitive. It has been patterned after J. WEIZENBAUM's ELIZA systems (1966) which were interpreters for exchangeable scripts, consisting mainly of decomposition and reassembly rules; the difference is the rules of MARYSIA's scripts can refer to morphological descriptions of a word. The rich inflexion of Polish is here of some help, because many syntactic relations and some semantic facts are clearly reflected by morphology, and therefore even simple means can cover some parts of syntax and semantics. We do not know yet how large is the domain of syntax and semantics, which is reducible to morphology (and also to MARYSIA's script rules). To find this out as well as to recognize the practical applicability of a morphology based conversational system are the secondary aims of the MARYSIA project.

2. MARYSIA'S LINGUISTIC PROBLEMS

2.1. *General assumptions.*

Automatic text processing of any kind forces us to face many linguistic problems of great importance. If a working system is required as a result of a project, then as a rule it is impossible to spend much time on working out solutions to all problems; in most cases we take for granted, sometimes even unconsciously, existing opinions. It is my feeling that we should not take for granted all our linguistic background, because almost every project can verify or reject some linguistic statements, e.g. the work on a frequency dictionary can clarify some problems of homonymy, etc. The main theoretical point of the MARYSIA project was the concept of " word ".

There are many definitions of " word " in the linguistic literature. Why do we not want to use any of them? The reason is that all of them (at least all I know) are of no use when we want to decide whether a given object is a word or not. Such a situation is fairly common in linguistics, let us take for example the well known definition of

" morpheme ": " the minimal meaningful unit of an utterance " and try to check that a given text is an utterance, that a given unit is meaningful and that it is minimal. After all, we do not have to accept the situation. How to avoid it then? It is necessary to find a basis for linguistic researches, serving as the only criterion for evaluating the theories. If we look for such a basis, we realize more than ever that language has no clear boundary in any aspect. There is no border between languages in space and time, there is no border between using language and other types of behavior (e.g. between understanding utterances and reasoning based on knowledge of reality); the opinions concerning perception of speech and handwritten letters have recently changed very much, so the concept of spoken or written utterance is no less vague than the " meaning ". What way out is there? Let us draw attention to the fact that a printed or typed text is quite different from any other kind of utterance, because it is in fact a string of characters from a finite, well defined alphabet. A new page, a change of type font, etc. can be considered as special letters in the alphabet, as is the case in computer composition systems. Therefore we can decide that every well printed text is equivalent to a computer-readable text of any form (paper or magnetic tape, text prepared for OCR readers, etc.). In the present state of the art the computer readable text is, in my opinion, the only basis for all linguistic research. In other words, if we consider Babbage as the father of computers, it is Gutenberg who is the father of linguistics (at least computational linguistics).

A unit which can be defined strictly on the basis of a computer readable text is a " word ", i.e. a string of characters between two delimiters, e.g. punctuation marks. Such words are of different kinds, they can constitute numbers, abbreviations, mathematical formulae, etc. We will consider now only those words which are composed exclusively of letters (or have been substituted by such a word, e.g. the word *5* in English can be substituted by *five*). Of course, the division of text into words is of little interest for a linguist for two reasons. First, spelling rules are often rather loose, therefore the same text can be segmented in different ways, and " a word can have different spellings " (I put this in quotation marks because " word " obviously has a different meaning in this context). Secondly, a word – again because of spelling rules – is sometimes too long for our purposes. I refer to cases when a word is obtained by concatenation of two or more different words (which happens in Polish and is very frequent in German for example). The second difficulty is more important and we solve it first by introducing the notion of a " lex ", which is a word or a substring of a word. Let us distinguish now word-types, word-tokens, lex-types and lex-tokens. The lexes are defined mainly by enumerating their lex-types. For practical purposes this is quite enough, and from the theoretical point of view the finite list of lex-types can be supplemented by a device for generating potential lexes from a finite dictionary of morphemes. I would like to stress our point that all lexes of practical significance come from a finite (although large) dictionary. It is also important that as a rule lexes are quite different from morphemes; they can be described loosely as " words, which because of spelling rules can be sometimes written together ".

2.2. *Hierarchy of linguistic units.*

In this paragraph I present the hierarchy of linguistic units as implemented in the MARYSIA system, i.e. as it was designed in 1970-1971. It has in general stood the test of time and the only changes it is subject to are of an aesthetic kind. The terminology I use here is consistent with English summaries of my papers (J. St. Bień, 1971; 1972 *a*; 1972 *b*).

Lexes exhibit different features, which are not equally relevant to us. It is natural then to consider them as variants of higher-level units. Therefore we introduce the notion of a " lexeme "; a lexeme-type consists of an ordered set of its allolexes and a choice function describing what allolex is to be used in a given context. All allolexes of a lexeme should be fully equivalent from the linguistic point of view although they may have different spelling or pronunciation. Examples of allolexes in Polish are *niego*, *jego*, *go* (all mean " him ", the first is used after a preposition, the second at the beginning of an utterance, and the third in all other contexts), in German *neue* and *neuer* (strong and weak declensions of an adjective), in English *a* and *an*, etc. The choice functions are implemented as a special kind of finite automaton, which has as an input the lex-tokens which are in the neighbourhood of the lexeme-token under consideration. It follows from this that allolexes are never stylistic variants. We think that distinguishing stylistic differences in texts can be very useful in some applications, e.g. computer aided language teaching. Therefore every lexeme, apart from its strictly grammatical properties, has its stylistic features which I call frequency evocation and quality evocation. At present frequency evocation can have as a value one of five grades: proper, acceptable, rare, wrong, non-existent; the quality evocation is described by means of qualifying labels. All lexemes with the same grammatical properties fall into one " form ". We insist that for every form it is one lexeme which is the " best " one, i.e. it has the highest frequency evocation. This is a way to obtain a normative dictionary together with an adequate enough description of the real vocabulary.

Until now we have introduced four linguistic units (word, lex, lexeme, form), but none of them is equivalent or even similar to the most popular meaning of *word* (in Polish *słowo*, *wyraz*), i.e. word in the sense of e.g. A. PENTILLÄ (1972). We call such a unit a "formeme "; it is an ordered set of forms. The ordering is necessary, because with every position in the set some syntactic features are connected. Now the problem is: what syntactic features can be put together into one formeme, in other words, how to establish borders between formemes. Our answer is that it can be done only arbitrarily by trading off the complexity of dictionary entries and the grammar which uses them. In the MARYSIA system forms of a formeme can exhibit only features of number, case, gender and person; all other features are assigned to a formeme as a whole.

There is also one more notion, it is the " group ". The group was designed for strictly technical purposes, i.e. for making dictionaries more compact by collapsing the descriptions of similar formemes into single entries. On account of the lack of a semantic component in the MARYSIA system it is used now in a different way. We put some formemes into one group if and only if there are enough regular differences between them from the semantic point of view. For example, an adjectival class of groups contains in every entry the positive and comparative degrees of the adjective, its adjectival adverb and the adjectival noun; the verbal class of groups contains the Present Tense and simple forms of the Imperative Mood, the Past Participle, the Passive and the adjectival Simultaneous Participle, etc.

As far as I know, the notions introduced for the MARYSIA system have no counterparts in linguistic theories, mainly because MARYSIA notions account for stylistic variations. Another important difference is that they are based on text words and therefore they do not describe phrases (e.g. verb forms which are spelled separately). It may seem strange that so many notions have to be introduced to clarify the notion of word (at least for Polish), but it seems to me that a convenient and elegant description of a vocabulary still requires some additional notions.

2.3. *Morphological coordinates.*

It should be noted now that of the five notions introduced in the preceding paragraph, only one of them refers to an observable and printable object, i.e. the lex (strictly, the lex-token). The problem is then how to refer to any concrete object of another type, e.g. a lexeme, a group, etc. The solution we use (J. St. Bień, 1970; 1972 *a*) is the following. Every item of a vocabulary possesses its paradigm, i.e. the set of all lexes which are included (directly or not) in the item. When the full paradigm designates the object we can refer to it by enumerating lexes of the paradigm; if this is not the case, we have to mark the level of the item (e.g. the word *pod*, meaning " under ", can label the prepositional formeme, the only inflexional form of the formeme, or the only lexeme of the form, etc.). This method is safe, but rather inconvenient when the paradigm of an item is numerous. In this case we can use an abbreviated method of reference, i.e. we may describe the paradigm instead of enumerating it. In most cases it is enough to give only one lex of the paradigm to describe it exactly, but in some situations it may be necessary to give two or more of them. It is worth noting that any lex (or set of lexes) can be used to label a paradigm, although we may prefer the traditional convention of using the Nominative Singular for nouns, the Infinitive for verbs, etc. For distinguishing different levels of vocabulary it was suggested in J. St. Bień (1972) that we use different type fonts (or underlining and quotation marks in manuscripts), but the more traditional " labelled bracketing ", e.g. $[\mathrm{dom}]^{\mathrm{FM}}$ for the given formeme, can also serve for this purpose very well.

The above mentioned method is very good for a human, but it is inconvenient for internal representation of vocabulary items in computer programs. Especially for this purpose we have designed " morphological coordinates ". We have noticed that every item can be referenced by means of giving the address of the biggest vocabulary item it is included in, and specifying some of its particular features. The features which are used for the purpose in the MARYSIA system were also influenced by technical considerations. As the result of the trade-off the MARYSIA's morphological coordinates are the following:

1. The dictionary item address. The item may represent a group or a formeme.

2. The morphological type of the formeme under consideration.

For formeme items it serves mainly for checking purposes, but for groups it describes a subset of all formemes belonging to the given group.

3. Serial number of the formeme in the formeme subset of the given morphological type. For formeme items it is equal to zero and serves only for checking. For group items it describes together with the second coordinate exactly one formeme of the given group.

4,5,6,7. The values of, respectively, number, case, gender and person categories. A value equals zero if a category does not concern the formeme.

8. Serial number of the allolex in the given lexeme.

If all eight coordinates are specified, then as a rule we refer to exactly one lex. In some cases there are some stylistic variants of the given lex; they have the same morphological coordinates. Then we can specify qualifying labels for evocations which are of interest to us; if we do not do this, it is assumed we refer to the " best " lex of the given form.

The most important property of the morphological coordinates is that they can serve as a convenient tool for handling useful sets of lexes. We obtain the result by leaving some coordinates unassigned. In this way we can reference e.g. the whole paradigm of a traditional verb by specifying only its dictionary address. We can refer to any form of the Present Tense of the given verb by specifying the first three coordinates. If we want to refer to any form of the Present Tense of any verb, we just have to leave the address coordinate unassigned. For checking agreement in an utterance we are interested in an object such as any noun (no matter whether a " normal " noun or the Gerund etc.), we can specify it by assigning the respective value to the second coordinate. There are many other possibilities, but the examples given above should be sufficient to prove that the morphological coordinates are a convenient means for handling different vocabulary items.

3. MARYSIA SYSTEM FROM THE USER'S POINT OF VIEW

3.1. *Script.*

For every application of the MARYSIA system at least one script should be prepared. The primary purpose of a script is to establish a way of classifying all possible utterances into some kinds of required reaction types; this is obtained by listing " decomposition rules " which should be applied to an utterance for every phase of the man-machine dialog. The secondary purpose of the script is to allow generation of a computer response by means of " composition rules ". At the moment scripts are coded in a formalism oriented towards its internal representation in the computer, because a planned preprocessor has not yet been implemented. Therefore I will not give any concrete example of a script, but I will describe it verbally.

A decomposition rule is a basic item of a script, it describes a class of utterances, which are formally similar. It is composed of three parts: a list of lex schemata, a list of allowed permutations and the list of required relations between lexes. A lex schema consists of eight slots for morphological coordinates, the slots can be filled by coordinate values or left unassigned. In this way a schema designates some sets of lexes, which can range from exactly one lex to the set of all lexes described in the system dictionary. There are also some special schemata, e.g. " short general schema " means any lex from the dictionary or an empty lex (i.e. no lex at all), " long general schema " means a a string, possibly empty, of lexes from the dictionary, separated by non-final punctuation marks (e.g. spaces, commas). There is also a very important schema called " word schema ", which matches every word not recognized by the morphological analysis of the system.

Lists of permutations were introduced because of the fairly free word order in the Polish language. A permutation is a string of references to schemata, described in the first part of a rule, separated by descriptions of required punctuation marks (including an empty punctuation mark for lexes which are to compose words). For every permutation there is a " reaction ", which is not set by the system, but can be arbitrarily defined by a user (e.g. it can cause some computation or just point to a composition rule for preparing an answer).

When a decomposition rule is applied to an utterance, the following actions are taken. First, the utterance is preprocessed to remove superfluous spaces, change upper case letters to lower case equivalents, etc. Then the words are split into lexes when necessary, and lexes are classified according to the schemata of the rule. Next, permutations are checked sequentially until one of them matches; now is the moment when the relational part of the rule becomes important. There are two types of relations. One of them is called " agreement " and really serves for checking agreement of given coordinates (usually number or case) of instances of lexes described by specified schemata. The second one is called " government " and is used to compare a specified coordinate against a constant or against one of eight " phraseological numbers " provided for every formeme by the dictionary. The phraseological numbers describe some syntactic features of a formeme, e.g. the rection of a verb, the gender of a noun, etc. If the specified relation holds, the match is successful and the reaction associated with the permutation is passed as the result; otherwise the next permutation or the next rule is applied.

The structure of composition rules is very similar to decomposition rules. The main differences are that for obvious reasons there is only one permutation and that the permutation can refer not only to its own schemata, but also to instances of schemata of the most recently applied decomposition rule. The other difference is that the relations are not checked but realized, i.e. the value of a coordinate of one instance of a lex is assigned to a specified coordinate slot of another schema (" agreement "), or the value of a constant or a phraseological number is passed to a specified slot (" government "). After this process all coordinate slots of all schemata should be filled, then the lexes specified by coordinates are generated and printed as a computer utterance.

Scripts contain all rules which are to be applied in a conversation. In different moments of a discourse it is necessary to use different subsets of decomposition rules or to apply a different order for matching them. For the purpose decomposition rules can be grouped into " expectation sets ". This is not required for composition rules as they are pointed explicitly by reaction in the matched permutation or by the " brain " of a user's system.

3.2. *Dictionary.*

From the users' point of view the MARYSIA system should have only one dictionary; this is not the case at the moment because we have not yet implemented some necessary utility programs and a user who wants to update the MARYSIA's vocabulary is involved with three dictionaries. It is only a temporary situation and therefore I will describe now exclusively the main dictionary, which is to be the " only " one.

The main dictionary contains items, which are composed of three divisions: morphological, syntactic and pragmatic. The last one is not used in practice. The syntactic division is rather primitive, it constitutes just a set of eight phraseological numbers per formeme. The morphological division is of most interest to us and it is the most complicated. First, it is split into four parts, according to the four grades of frequency evocation. The reason for this is the following. In some applications it may be necessary to reduce the adequacy of the dictionary because of constraints on dictionary size or because it just will not be needed; then we are able to remove easily the parts of items which are of no interest. Next, every part is a list of morphological segments (in most cases it contains only one segment). Every segment has its quality evocation, which is stored as a string constituting a qualifying label from W. DOROSZEWSKI's dictionary (1958), and is often empty. As in the case of frequency evocation, we can easily get rid of segments with no empty labels if we do not need them. Segments are of three types, which serve different purposes. The simplest one is a quotational segment, which contains a list with explicitly coded lexes, together with their morphological coordinates and their quality evocations. This segment is used separately only for uninflected items; it serves more often as a supplement to other types of segments and thus contains the " variant " or " exception " forms of a paradigm. The second type of segment is a generation segment, which is used to describe some irregular items by means of an algorithm for generating their lexes. The third and the most important one is a parametric type of segment.

A parametric segment contains three ordered sets of parameters, which are called morphological evocations, morphological numbers and morphological bases. The latter two of them can be considered as a generalization of traditional concepts respectively of pattern of inflexion and of a stem. The difference is that a base can be constituted by an arbitrarily defined string of letters, and the number of bases for describing the given type of item can also be arbitrarily defined. Similarly, inflexional patterns traditionally classify whole paradigms, but we can arbitrarily split the paradigm into some subparadigms (which may consist even of single forms) and then we can independently assign a description to every subparadigm. The morphological evocation has no counterpart: it decides whether a slot in a paradigm is filled by a given item or not.

There are different levels of parametric segments. If a segment describes a formeme, then it belongs to the inflexional level. In the first stage of dictionary development all items can belong to this level, but if we want to make full use of script possibilities, we have to provide also the derivational level. The segments of this level describe groups, i.e. sets of formemes. This is done by simulating the inflexional segments which had to be put in the dictionary if the derivational level was absent. We can introduce also a third level, the extractional one. An extractional segment provides parameters for a special type of group; by taking into account idiosyncrasies of a given type of group, the extraction segment can use less parameters (especially bases, which are very space consuming) than an equivalent derivational segment. The multilevel lex generation allows us to trade off between the size of a dictionary and the time of lex generation. Together with possibilities of other trade-offs, e.g. adequacy versus dictionary capacity, it should make the dictionary system easy to adapt to different applications.

3.3. *System tables.*

Developing a good algorithm of inflexional analysis and synthesis is not an easy task. Instead of trying to obtain it in the first attempt, we decided to design our system as a set of table-driven programs. Therefore we may improve the system performance by exchanging step by step its tables; we can also easily change our previous decisions concerning, for example, borders of formemes, etc. It even seems possible to change the MARYSIA system into another language version, the results of the first attempt to do this (L. Kwieciński, 1972) are encouraging. Now we will review the system tables in the order of their application for system response.

The input utterance is at first preprocessed and coded in special PF code; these are the only non-table-driven parts of the system. Then the words are divided into lexes. This is the task of two finite automata (all system automata are, of course, driven by exchangeable tables), which scan a word in both directions and establish probable lex borders. Now the lexes are to be transformed into keys for searching in a backing dictionary called the index. The transformation consists of cutting some letters from the ends of the lexes; the place for the cut is indicated by another set of automata. Every automaton of the set is working on the assumption that the lex belongs to a given formeme type, then the key (or keys) suggested by the automaton is searched (by means of hash coding) in the segment of the index which is devoted mainly to keys of the given formeme type. It has some advantages.

First, it is a way of solving some cases of homonymy, next, the automata are small and therefore easy to design and to debug. The keys are not matched exactly but owing to the PF code (J. ST. BIEŃ, 1971) and special formats of the index entries, stem alternation is not taken into account during the matching. It should be noted that at this moment some false hypotheses concerning lex borders are rejected because respective keys are not found in the index. The index contains pointers to the linkage dictionary, which was designed as a separate part because of storage constraints. The linkage dictionary yields for every lex its first three morphological coordinates, i.e. a formeme specification including formeme type. We have noted that the latter information together with the lex itself is usually enough to establish the rest of the morphological coordinates with high probability, therefore now the lex is inspected by one of the special automata, which outputs possible coordinates. If we do not require 100 percent probability that the coordinates are correct, we can stop the analysis at this moment; otherwise we reconstruct the lexes by synthesis and reject false hypotheses.

The tables for the synthesis are more differentiated. First, there is a table of formatives, i.e. strings of letters used to compose lexes. Next, there is a table of choice functions. Choice functions are finite automata (they can also compose a choice function segment in a dictionary item). Then there is a large table called the inflexional partition. Besides some technical information it contains algorithms transforming parameters of an inflexional segment into lexes; algorithms are expressed by means of extremely primitive " morphological description language " consisting of about ten instructions. The other two partitions are optional. They contain algorithms in the morphological description language to transform parameters of one level into parameters of another level, i.e. extractional ones into derivational ones or derivational parameters into inflexional ones.

For all types of information needed by the MARYSIA system there is a computer-independent (although at the moment rather awkward) external form. Its syntax is given in the BNF notation and its semantics is described in Polish in BIEŃ *et al.* (1973).

4. *Present state of the project and the future development.*

Because of the delay in installing a new computer for Warsaw University, we have decided to implement the system in the first instance on the GIER computer, the only available one when the project was started. It was decided to write the programs in GIER ALGOL 4 and to split the analysis and synthesis parts of the system into passes because of fast storage constraints. At the moment all parts of the system have been implemented, the tables of the system have been debugged and thoroughly tested; small dictionaries for testing purposes have been prepared. Still before us is checking the system as a whole, working according to some testing scripts.

In the future we want to rewrite the MARYSIA system for a bigger and faster computer (it will probably be the IBM 360) and to develop some utility programs to facilitate loading the backing dictionaries and script writing. We will also check the generality of the system tables by preparing a German language version of the MARYSIA system.

As far as the long-term plans are concerned, the following tasks are to be solved. First, it will be necessary to improve the adequacy of the MARYSIA morphological component by increasing the number of entries in the dictionaries. Secondly, it will be necessary to develop systems which will cover the higher levels of the language; because of our " bottom-up " approach to language description it will be the syntax that will be elaborated next. The third direction of the research can be called developing text-world interfaces; I mean by this accepting texts prepared for typesetting devices, optical character recognition, and voice input and output. For technical reasons, the OCR will probably be excluded; speech processing by computer is the interest of another group at Warsaw University and we hope to join together at a suitable moment, which should not be before developing at least a good syntactic parser (following the recent ideas of e.g. D. R. HILL, 1972). Therefore in the near future we will be interested only in input of text coded on different kinds of media used in the printing industry.

Sample dictionary entries

```
-S_LO_NCU-
-S_LO_NCU-
-S_LO_NCA-
-S_LO_NC-
0
]
0,0,0
1
[1,
4,1,0,5,1,1,0,0
]
0
>
1,
<
1,2
0
1,3
[
2,7,6
2,2
1,5,1,3,1,5,4
-P_LUCO-
-P_LUCA-
-P_LUCU-
-P_LUCU-
-P_LUCA-
-P_LUC-
0
]
0,0,0
1
[1
4,1,0,5,1,1,0,0
]
0
>
1,
<
1,2
0
1,3
[
2,7,6
2,2
1,6,1,3,1,5,4
-_LYKO-
```

```
-_LYKA-
-_LYKU-
-_LYKU-
-_LYKA-
-_LYK-
0
]
0,0,0
1
[1
4,1,0,5,1,1,0,0
]
0
>
1,
<
1,2
0
1,3
[
2,7,6
2,2
1,5,1,3,1,5,4
-D_LUTO-
-D_LUTA-
-D_LUTU-
-D_LUCIE-
-D_LUTA-
-D_LUT-
0
]
0,0,0
1
[1
4,1,0,5,1,1,0,0
]
0
>

1,
<
1,2
0
1,3
[2,7,6
2,2
1,5,1,6,1,8,7
-PISKL_E-
```

SAMPLE PARADIGM LISTING, USED FOR CHECKING THE DICTIONARY

```
[BIEN] CKL2
[BIEN] CKL2
ITEM 1596
LEVEL  1  TYPE  6


  FORMEME TYPE  6
     FORM CATEGORIES    0   1   0   0
        PROPER LEXEME

           LEX    JA
           LABEL


     FORM CATEGORIES    0   2   0   0
        PROPER LEXEME
           CHOICE FUNCTION ADDRESS     73
           ALLOLEX NUMBER  1

              LEX     MNIE
              LABEL
           ALLOLEX NUMBER  2

              LEX     MNIE
              LABEL
           ALLOLEX NUMBER  3

              LEX     MI_E
              LABEL


     FORM CATEGORIES    0   3   0   0
        PROPER LEXEME
           CHOICE FUNCTION ADDRESS     73
           ALLOLEX NUMBER  1

              LEX     MNIE
              LABEL
           ALLOLEX NUMBER  2

              LEX     MNIE
              LABEL
           ALLOLEX NUMBER  3

              LEX     MI
              LABEL


     FORM CATEGORIES    0   4   0   0
        PROPER LEXEME
           CHOICE FUNCTION ADDRESS     73
           ALLOLEX NUMBER  1
```

```
              LEX    MNIE
              LABEL
           ALLOLEX NUMBER  2
              LEX    MNIE
              LABEL
           ALLOLEX NUMBER  3
              LEX    MI_E
              LABEL

     FORM CATEGORIES   0   5   0   0
        PROPER LEXEME
           LEX    MN_A
           LABEL

     FORM CATEGORIES   0   6   0   0
        PROPER LEXEME
           LEX    MNIE
           LABEL

     FORM CATEGORIES   0   7   0   0

        NON-EXISTENT

ITEM 1615
LEVEL   1  TYPE   6

  FORMEME TYPE  6
     FORM CATEGORIES   0   1   0   0
        PROPER LEXEME
           LEX    TY
           LABEL

     FORM CATEGORIES   0   2   0   0
        PROPER LEXEME
           CHOICE FUNCTION ADDRESS      73
           ALLOLEX NUMBER  1
              LEX    CIEBIE
              LABEL
           ALLOLEX NUMBER  2
              LEX    CIEBIE
              LABEL
           ALLOLEX NUMBER  3
              LEX    CI_E
              LABEL
```

```
FORM CATEGORIES    0   3   0   0
   PROPER LEXEME
      CHOICE FUNCTION ADDRESS      73
      ALLOLEX NUMBER  1
         LEX     TOBIE
         LABEL
      ALLOLEX NUMBER  2
         LEX     TOBIE
         LABEL
      ALLOLEX NUMBER  3
         LEX     CI
         LABEL

FORM CATEGORIES    0   4   0   0
   PROPER LEXEME
      CHOICE FUNCTION ADDRESS      73
      ALLOLEX NUMBER  1
         LEX     CIEBIE
         LABEL
      ALLOLEX NUMBER  2
         LEX     CIEBIE
         LABEL
      ALLOLEX NUMBER  3
         LEX     CI_E
         LABEL
```

REFERENCES

J. St. Bień, *Prowizoryczna terminologia czasownikowa* (unpublished paper), 1970.

J. St. Bień, *An Alphabetic Code for the Inflexional Analysis of Polish Texts,* in « Algorytmy », VIII (1971), 14.

J. St. Bień, *O pewnych problemach przetwarzania jezyków fleksyjnych na maszynach cyfrowych,* in « Prace Filologiczne », XXIII (1972^{a}).

J. St. Bień, *O dwóch pojęciach pożytecznych przy automatycznym przetwarzaniu tekstów,* in *Z polskich studiów slawistycznych,* Seria 4, Językoznawstwo, Warszawa, 1972^{b}.

J. St. Bień, W. Łukaszewicz, S. Szpakowicz, *Opis systemu* MARYSIA, in « Sprawozdania IMM i ZON UW » (Reports of the Warsaw University Computational Centre), n. 41, 42, 43 (1973).

W. Doroszewski (ed.), *Słownik języka polskiego,* 11 voll., Warszawa 1958-1969.

R. W. Floyd, *Towards Interactive Design of Correct Programs,* IFIP Congress 1971, Invited Papers 1971.

D. R. Hill, *An Abbreviated Guide to Planning for Speech Interaction with Machines: the State of the Art,* in « International Journal of Man-Machine Studies », IV (1972) 4.

L. Kwieciński, *Die deutschsprachige Variante des Konversationssystems* MARYSIA (M. A. thesis), Warsaw 1972.

C. Meadow, *Man-Machine Communication,* New York 1970.

A. Pentillä, *The Word,* in « Linguistics », LXXXVIII (1972), pp. 32-37.

J. Weizenbaum, ELIZA: *a Computer Program for the Study of Natural Language Communication between Man and Machine,* in « Communications of the ACM », IX (1966) 1.

T. Winograd, *Procedures as a Representation for Data in a Computer Program for Understanding Natural Language,* Cambridge (Mass.) 1971.