Preprint No. 25 Classification: CA 2.2

Organization and Programming of the Multistore Parser

- P. P. Pisani -

INTERNATIONAL CONFERENCE

ON

COMPUTATIONAL LINGUISTICS

COLING

1969

KVAL
Forskningsgruppen för KVANTITATIV LINGVISTIK

RESEARCH GROUP FOR QUANTITATIVE LINGUISTICS

Address: Fack Stockholm 40, SWEDEN

International Conference on
COMPUTATIONAL LINGUISTICS
1-4 September 1969
Sånga-Säby, Sweden

ORGANIZATION AND PROGRAMMING OF THE MULTISTORE PARSER

Pier Paolo Pisani

(Georgia Institute for Research and University of Georgia)

The research reported in this paper was sponsored by the U.S. Air Force Office of Scientific Research (OAR), Information Sciences Directorate (Grant AFOSR 1319-67) and actively supported by the University of Georgia, Athens, Ga. U.S.A.

The Multistore system was developed in order to recognize and explain structural patterns in natural-language sentences (specifically English) and eventually yield an output in which the relations between the various items of the sentence are hierarchically displayed.

The recognition of these structural patterns is made by means of a system of rules which operate on a sequence of words, i.e. a sentence, whose individual characteristics are pre-established. By individual characteristics are meant the possibilities a word has to correlate (i.e. to form a syntactic combination) with another item; these possibilities are represented by 'correlators', that is, by syntactic elements which link two items in a correlation.

Each word is characterized by a set of pre-established data:

a) the S-code, which distinguishes between the various senses of a homograph. For instance, a word like "READ" will have four different S's to distinguish between:

READ = supine	e.g.	I CAN READ
READ = past tense		YESTERDAY I READ
READ = past part.		I HAVE READ
READ = noun		A LONG READ

These distinctions are essential, since whenever a homograph occurs, one and only one of its meanings can be taken into consideration to make the final pattern, unless, of course, the sentence is ambiguous and more than one final pattern is to be recognized, as in:

```
                     i) present tense
I READ THE BOOK
                     ii) past tense
```

b) the sequence of correlational indices (Ic's), that is, the string of potential links that each word-sense has.

Each Ic represents a possible syntactic connection between two items and is identified by:

1) the code number of the relation it establishes between two items;

2) the 'type' of correlation. There are six different types of correlation which split into two groups:'explicit' correlators and 'implicit' correlators.

By 'explicit' correlator we mean a linking element which is represented by a linguistic item; prepositions and conjunctions are explicit correlators; by 'implicit' correlator we mean a relation between two items, which is not expressed by any linguistic item but is indicated by the relative position of the two items (which we call their correlational function).

IMPLICIT CORRELATOR

```
Type N                I      AM
                      |      |
                      N1-----N2

Type M                AM     I
                      |      |
                      M2-----M1

Type V                SERIOUSLY, HE LEFT
                          |        └V2┘
                          V1---------┘
```

EXPLICIT CORRELATOR

```
Type  E          DUCKS  IN  ATHENS
                   |    |     |
                   E1---E3----E2

Type  F          BY  CAR  THEY  TRAVELLED
                 |    |    └─F1──┘
                 F3---F2─────┘

Type  H          DOLLS  SHE  PLAYS  WITH
                   |     └─H1─┘       |
                   H2───────┴─────────H3
```

For each type there are different correlational functions which determine the position a word has in a correlation. When two adjacent words have complementary functions of the same Ic - for instance, word A has 5050 N<u>1</u> and word B has 5050 N<u>2</u> - a 'product' is made and recorded in the form:

Word A 5050 N Word B

This product is considered as one piece and can become first or second correlatum in a wider correlation and is therefore treated as though it were a single word, i.e., it is assigned strings of Ic's which indicate its correlational possibilities both with adjacent words and with adjacent products already made. Single words, however, being vocabulary items, have their strings of Ic's assigned <u>a priori</u>; products, since they arise during the procedure, have to be assigned their Ic-strings dynamically. The assignation of specific Ic's to a product depends on:

a) the correlator responsible for the particular correlation;
b) the characteristics (Ic's) of the word (or product) which

makes up the first or the second correlatum.

The operational cycle that assigns Ic's to a product we call 'reclassification'.

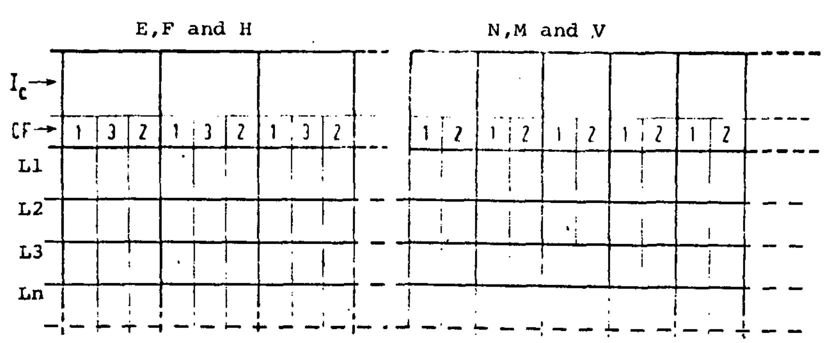
The amount of data involved in an analysis of this kind is really enormous. Let us consider a sentence consisting of ten words, each of which has two different senses (S's). On an average 50 correlational indices are assigned to each sense of a word. Now, just to check the correlational compatibility of two adjacent words about 10,000 matching operations would be necessary; the matching procedure for all the words of the sentence would involve about 90,000 operations. On an average five products would result from the first 10,000 matching operations; each of them would be assigned about 50 correlational indices that represent the product's correlational possibilities to correlate with a another adjacent piece - either a word or a product. The procedure to match these five products with another piece would involve about 637,000 operations. If to this figures we add the number of operations necessary starting from level 3 (see p. 7) with all the products made in the immediately preceding levels (200,000), the total number of operations involved would come to 927,000.

The reclassification routine also involves a great number of operations of this kind: about half of the correlational indices a product is assigned depends on the correlator responsible for that correlation; the other half depends on the strings of indices which the two correlata of the product have. According to the presence or absence of specific indices in the strings of the first or second correlatum, pre-established sets of indices are assigned to the product; or sets of indices are assigned to the product only if they are present in the strings of its two correlata. The reclassification of each product would require about 2,000 operations, which means 100,000 for the average of 50 products in a sentence of 10 words, bringing the total of operations to over a million only for the matching procedure. This would imply - for this part of the program alone - processing times of the order of some seconds of machine time if the most modern computer is available, or of about an hour - at best - if the work is done with an older model.

The amount of work and money involved in a procedure of this kind made us try to find a quicker and more economical way of handling correlational indices: as a result of our efforts the Multistore system was developed.(Bibl. 1)

The basic idea of the Multistore consists in pre-establishing in a given area of the machine's central core as many separate positions as there are correlators in the system. The arrangement of these positions representing the correlators orders them according to type. This assures that at any point of the procedure each Ic is not used several times and in different ways according to the diverse data it contains, but only as one single item which by its positional coordinates implies its various significations. Moreover, the Ic's do not have to be compared one by one with the Ic's of other adjacent words or products, but are simply addressed to one and only one pre-established position. Thus the mass of operations of comparison is avoided and also the necessity to ascertain, after every successful match, which items the matched Ic's represent is eliminated, because the very position of the matched Ic's immediately implies what they stand for. To establish whether two Ic's are complementary and represent a correlation thus becomes the simple task of checking already present information according to the rules of sequence, of correlational function, and of correlator type, all of which are implicit in the location of the markers which are being handled.

The Multistore can be represented as a rectangular area divided into lines and columns. (see Fig. 1 below)



Every column is dedicated to one Ic and subdivided into two subcolumns, if the Ic is of type N, M, or V (implicit); if the Ic is of type E, F, or H (explicit), the column is divided into three subcolumns.
The lines L1, L2, L3 etc. divide the area into levels. The levels are determined by the succession of words in input. Thus each level bears the number of the word it represents. Every input word causes for each Ic in its Ic string the insertion of a marker into the Multistore column corresponding to that Ic; and the level of that marker in the Ic column corresponds to the input number and the position of that word in the sentence. Thus all the markers inserted for one word represent the correlational possibilities of that word.
If, on the line of level 1, an Ic of the string representing correlator type N of the first word has caused the insertion of a marker into the column corresponding to that Ic and if an Ic of string N of the second word has caused the insertion, into the line of level two, of a marker representing CF2 of the same Ic, this implicitly means that with the same correlator a correlation is made containing the first and the second word; this product No x, consisting of word 1 (S 'a') and word 2 (S 'b') is the product of correlator No y and type N. This product belongs to level 2 and when it has been assigned a string of Ic's by the appropriate rules of reclassification, it will be inserted

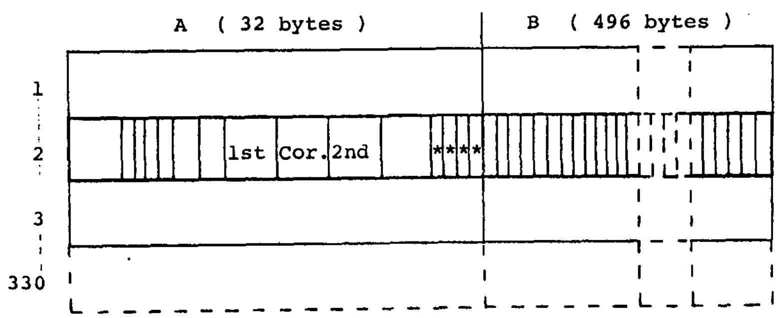
into the Multistore on the second level; this means that it can enter into combinations only with those words that belong to the immediately preceding level, or with products which contain the words of the immediately preceding level. Such a correlation, whenever it is made, would still belong to the level of product x. In our specific case product x could correlate only with an item of level zero, which does not exist, because product x is on level two and already contains word No. 1. Hence we can formulate a restrictive rule to the effect that a product can be a potential second correlatum in an N correlation only if its lower level is larger than 1. The Multistore system lends itself to the introduction of many such restriction rules.

When on a given level all products that have sprung from the insertion of markers corresponding to the word of that level have been reclassified,and the products originating from that reclassification have, in turn, been reclassified and have inserted their markers, and there are no more products to be reclassified, then the procedure inserts the next word and thus begins the next level. This means that once a subsequent word of the sentence has been inserted, all preceding words and products become 'inactive' pieces, having exhausted every possible attempt of correlation with 'active' pieces; from then on they represent merely latent correlational possibilities with subsequent items. This state of inactivity in the case of the last word of the sentence determines the end of the analysis. At this point the product (or products) that contains all words of the sentence is called 'complete' and represents the hierarchical structure of the sentence.

The first tentative program MP1 (Bibl. 1) was written for use on a GE 425 computer and its main purpose was to show the applicability of the Multistore system to correlational grammar and to check the method of programming based on 'significant addresses'.

The present program, MP2 (Bibl. 3) , is a revised and enlarged version of MP1 written for use on an IBM 360/67 computer. On the basis of our previous experience it can be considered an actual working tool.

Many solutions, as well as many restraints, depend on the fact that under many respects it is a machine-oriented program. The program is structured on a large area of the central core, divided into lines and columns, whose size is 528 x 330 bytes. Each line (330 all together) consists of 528 bytes and is divided into two sections: A and B. Section A contains all the data necessary to define a line; section B consists of 496 bytes, that is, of as many bytes as there are correlators operative in the system. Each line specifies as permanent data a reclassification rule - whose definition is given in section A of the same line - and the set of indices assigned by that rule (bit 6). The relevance of the rule to a given product is specified in the columns of section B.



* reclassification rule

Fig. 2

Each byte of section B is divided into 8 bits as illustrated below.

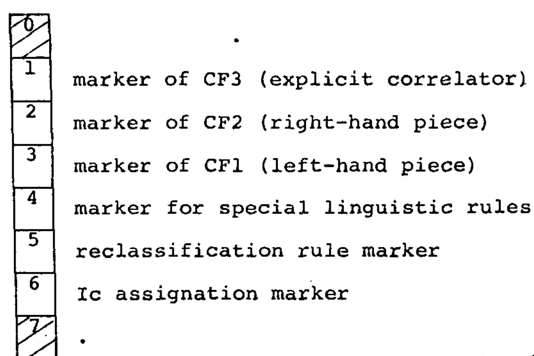


Fig. 3

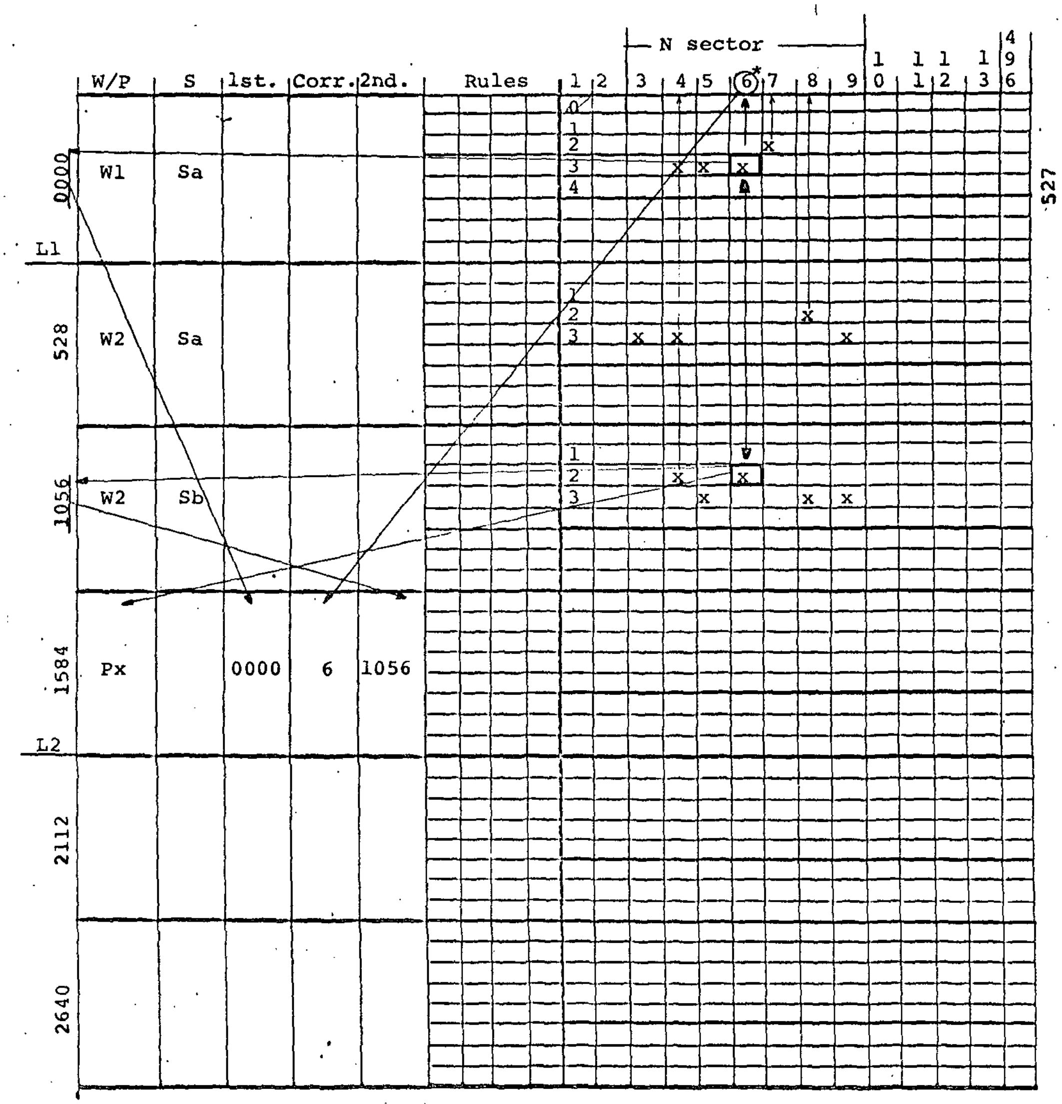
Bits 1 to 4 are therefore used in the matching procedure, whereas bit 5 and 6 are pre-established data to be used in the reclassification routine.

Procedure

Each S-value of a word occupies one line of the Multistore area and its specifications are recorded in section A of the same line. For each Ic contained in the string of that S-value of the word, a marker is inserted , according to the correlational function, in bit 1,2 or 3 of the corresponding byte of the line, that is, in the byte which bears that Ic as label.

According to its function, a marker can be a left-hand piece 'LH', and as such it is simply recorded, or a right-hand piece 'RH', in which case, immediately after it has been recorded, the column is searched for a complementary and contiguous LH piece. If this is found, an indication of product is recorded in the first free line of the Multistore; this address consists of three data: a) the address of the line where the LH piece was found, which is recorded in the area 'first correlatum';of the line of the product; b) the address of the line where the RH piece was recorded, which is recorded in the area 'second correlatum' and c) the relative address of the column which characterizes both both LH and RH pieces, which is recorded in the area 'correlator'.

After the product's specifications are recorded, and if there are no other LH pieces with which the RH piece in hand can combine, the routine for the insertion of Ic's is resumed. If the insertion of the next Ic of that S-value

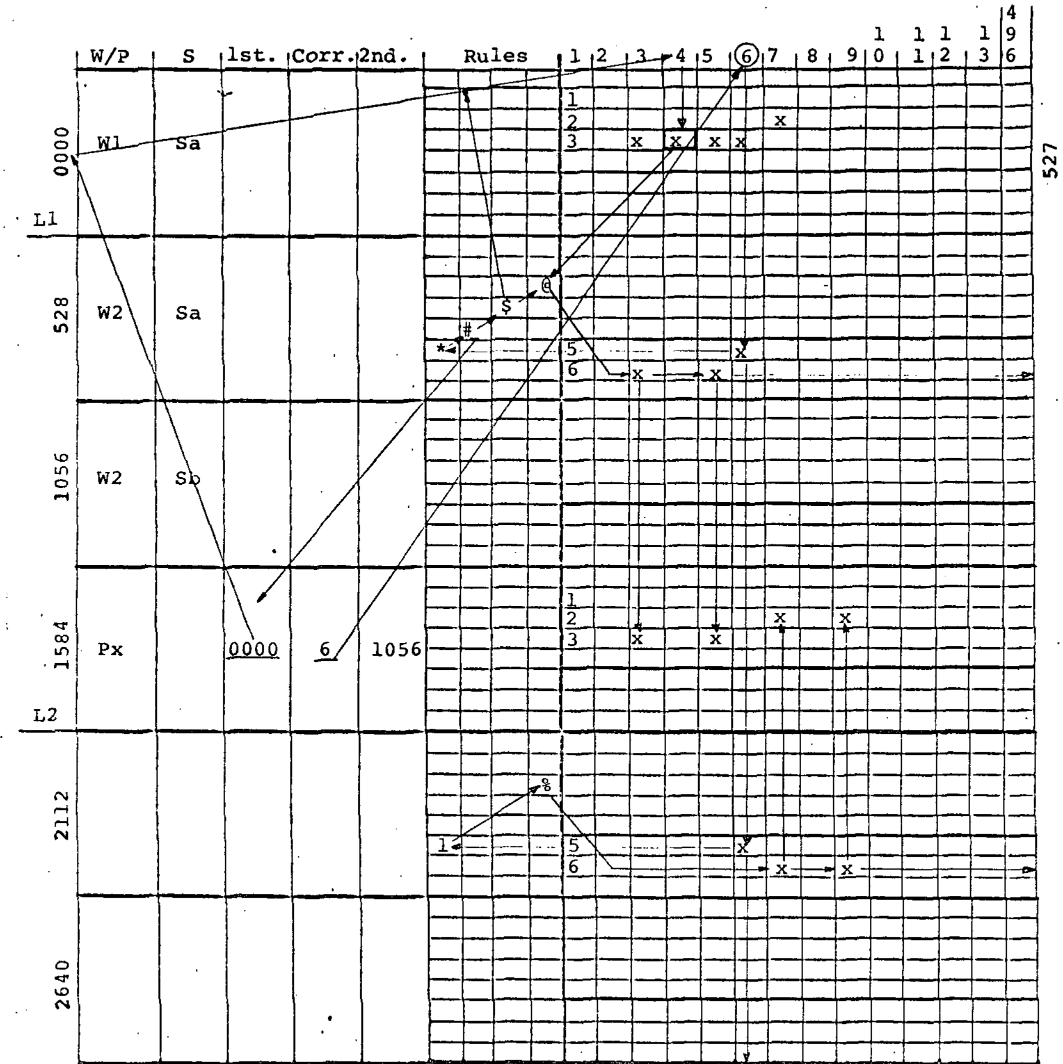


* The position 6 in the Multistore area corresponds to correlator No y in the same way as position 7 corresponds to correlator No z, and so on.

INSERTION DIAGRAM

of the word causes a new product to be made, the procedure is repeated and the product is recorded on the next free line of the Multistore area. Only when all the Ic's of the piece which has caused the production have been inserted, the reclassification routine takes place, starting from the first product newly recorded.

The information contained in the area 'correlator' of the line containing the product's record gives the address of the Multistore column dedicated to the correlator responsible for that product. The column is then searched , from the top down, for a bit 5 set ON (see Fig. 3 on p.10). If it is found, this implicitly means that on the line to which the bit belongs, there will be found the record of a reclassification rule relevant to the product to be reclassified. Section A of the same line contains the instructions concerning the assignation of the Ic's whose markers are contained in bit 6 (see Fig.3). A bit 6 set ON implicitly indicates either the column in which to check (if the rule requires it) the record of the first or second correlatum(the addresses of which are recorded in section A of the line in which the rule is recorded) for the presence or the absence (as specified by the rule) of bits 1 to 3 set ON (which represent Ic's); or it may simply indicate the place in which to insert a marker, i.e. a reclassification Ic, The routine for the insertion of reclassification markers is exactly the same as the routine for the insertion of markers for words.



* Conditioned rule. 1 Unconditioned rule. # Check on 1st correlatum.
$ Ic 4 CF1. @ Assign the string CF1 contained in the rule to the product.
% Assign the string CF2 contained in the rule to the product.

RECLASSIFICATION DIAGRAM

The analysis of the sentence is complete when the last marker of the last word-sense has been inserted and there are no further products to be reclassified or re-cycled. At this point the output routine starts. Three different kinds of output are produced:

a) a list of all the products made in the course of the analysis of the sentence;
b) a list of all Ic's assigned to each product during the reclassification routine ;
c) a graphic representation of the hierarchical structure of all 'complete' products (that is, containing all words of the sentence). This structure is equivalent to a tree structure with words at the terminals and correlators at the nodes. (see Appendix)

This is a general outline of the procedure of combination, production, reclassification and output. In addition to that there are several routines which meet special requirements. A special rule, for instance prevents specific RH pieces from becoming eligible LH pieces once a certain correlation - which contains them as RH pieces - has been made. A word like "LITTLE", for instance, in its function as a quantifier, once it has been correlated with the definite article and made the product "THE//LITTLE" cannot become LH piece in the correlation:

LITTLE // HE KNOWS

The indication 'discard' on print-out type 'a' - i.e. on the list of all the products made during the analysis - will show that "LITTLE" is no more available as LH piece for any other correlation.

Another restraint concerns some 'complete' products which, though grammatically correct, cannot be accepted as interpretation of the sentence. For instance, in a sentence like:

THEY // WERE READY

the structure which takes "WERE" as subjunctive is not acceptable, since it would require something else - an "IF" or "I WISH" etc. - to precede. In cases like this the indication 'non-sentence' appears in print-out type 'a'.

A set of special routines serves the purpose of recognizing idiomatic expressions. When one of them is recognized, inserted in the Multistore and reclassified -like any other product - the indication 'idiom' is printed on print-out type 'a'.

The whole program, including the Multistore area and buffers, is contained in the central core of the machine and occupies about 200 K. The system accepts as input,sentences of up to 16 words - a limit fixed in accordance with the average length of sentences in scientific texts (Bibl.5) and ample enough to allow any type of syntactic structure. Processing-times for 10-word sentences are about 1-1.5 seconds. Our present vocabulary is limited to 150 words for reasons of punched card maintenance. However, it could be enlarged without affecting the program.

The Multistore parser was developed for the automatic analysis of English sentences on the basis of correlational grammar, but it is in no way limited to this kind of grammar. Actually, by changing the input parameters, the symbols of the rules and the matching operations it could be used to handle the data of any kind of predictive grammar; neither are its dimensions critical; the Multistore area could be reduced or enlarged simply by altering the parameters in accordance with the storage capacity available in the machine.

Bibliography

1 'Multistore': A Procedure for Correlational Analysis (E.v.Glasersfeld, P.P.Pisani, J.B.Burns), Informal Report T-10, Automazione e Automatismi, vol. IX, No.2 Milan, Italy, 1965

2 Automatic English Sentence Analysis (Glasersfeld, Pisani, Burns, Notarmarco, Dutton), Final Report T-14 Grant AF EOAR 65-76, IDAMI Language Research Section, Milan, Italy, 1966.

3 The Multistore System MP-2 (E.v.Glasersfeld and P.P. Pisani), Scientific Progress Report, Grant AFOSR 1319-67 Georgia Institute for Research, Athens, Georgia, 1968.

4 The Multistore Parser for Hierarchical Syntactic Structures (E.v.Glasersfeld and P.P.Pisani) Grant AFOSR 1319-67, Georgia Institute for Research, Athens, Georgia 1969 (paper submitted to Communications of ACM)

5 Computational Analysis of Present-Day American English (Henry Kucera and W.Nelson Francis), Brown University Press, Providence, Rhode Island, 1967

APPENDIX, I

Complete Parsing

Input Sentence: 01 I 02 READ 03 IT 04 HIS 05 BRIGHT 06 BOOK 07 ON 08 TRAINS .

(Print-out type a)

P 0001	W 0001 01 $1	2220 N	W 0002 01 FS	C1 03	
P 0002	W 0001 01 $1	2250 N	W 0002 C4 FF	C1 03	
P 0003	W 0002 04 FS	4010 N	W 0003 01 $3	, C2 04	
P 0004	W 0002 03 FP	4010 N	W 0003 01 $3	, C2 04	
P 0005	W 0002 01 FS	4010 N	W 0003 01 $3	, C2 04	
P 0006	W 0001 01 $1	2250 N	P 0003	, C1 04	
P 0007	W 0001 01 $1	2220 N	P 0005	, C1 04	
P 0008	W 0004 02 SP	5230 N	W 0005 01 1A	C4 06	
P 0009	P 0008	2220 N	W 0006 01 VS	C4 07	
P 0010	P 0009.	4010 N	W 0003 01 $3	C3 07	
P 0011	W 0005 01 1A	5210 N	W 0006 02 $3	C5 07	DISCARD
P 0012	W 0004 01 FP	7016 N	P 0011	C4 07	
S 0013	P 0012	0550 E	W 0007 01 /7	C4 08	

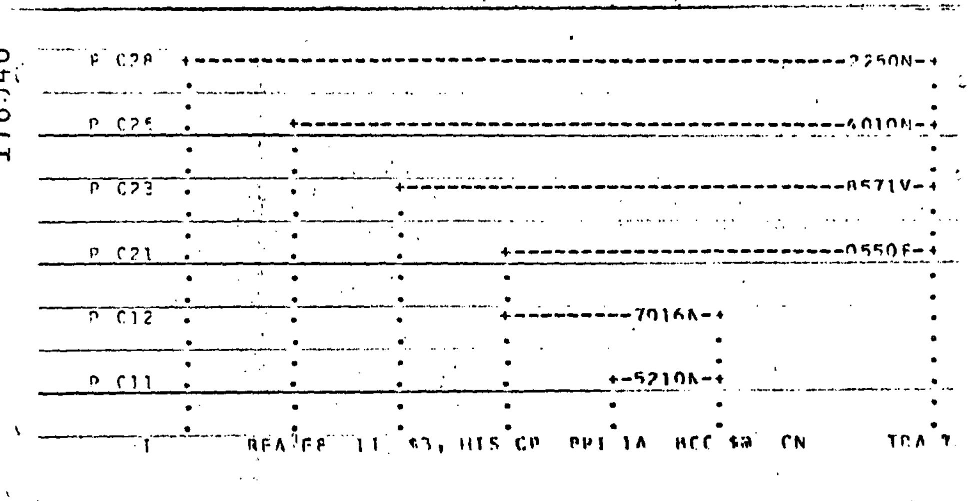
S CC15	W CCC6 01 VS	0550 E	W 0007 01 07		C4 C9
P C016	W CCC6 01 VS	4430 N	W CC07 C2 CN		C4 09
P CC17	F OCC8	2220 N	P 0016		C4 C9
P CC18	P CC17	4C1C N	W 0003 01 43		C3 09
P 0019	W 0CC6 01 VS	0550 E	W 0CC9 C2 7a	•	C6 C9
P CC2C	F CC10	0550 E	W 0008 02 7a	•	C3 09
P 0021	F 0C12	C55C E	W 0CC8 C2 7a	•	C4 C9
P CC22	F CCC8	2220 N	F 0019	•	C4 C9
P CC23	W CCC3 C1 43	6571 V	P 0C21	•	C3 09
P CC24	F 0C22	4010 N	W 0CC3 01 43	•	C3 09
P CC25	W CCC2 C4 F5	4C1C N	F 0023	•	C2 09
P 0026	W 0CC2 03 FP	4C10 N	P 0023	•	C2 09
P CC27	W CCC2 01 FS	4C10 N	F 0023	•	C2 09
P 0028	W 0CC1 01 41	2250 N	P 0025	•	C1 09

COMPLETE

II

178540

(Print-out type c)



```
P CC29          K CCC1 01 $1        2220 N    F 0027             .  C1 C9    COMPLETE
```

178542

IV

```
P C29  +---------------------------------------------------------2220N-+
       .                                                               .
P C27  .       +-----------------------------------------------4010N-+
       .       .                                                       .
P C23  .       .       +-------------------------------------9571V-+
       .       .       .                                               .
P C21  .       .       .       +---------------------------05505-+
       .       .       .       .                                       .
P C12  .       .       .       +---------7016N-+                       .
       .       .       .       .               .                       .
P C11  .       .       .       .       +-521CN-+                       .
       .       .       .       .       .       .                       .
       I       REA FS  IT  $3, HIS CP  ERI IA  ECC $2  CN       TP3 32.
```