Information Technology: Research and Development (1983), *2* (89–108)

AN OUTLINE OF THE REPRESENTATION AND USE OF TEMPORAL DATA IN THE RESEDA SYSTEM

G. P. ZARRI

Centre National de la Recherche Scientifique, Laboratoire d'Informatique pour les Sciences de l'Homme, 54, Boulevard Raspail, 75270 Paris Cedex 06, France

(*Received 15 March 1983*)

ABSTRACT

Temporal data are of fundamental importance in a biographical database. In the RESEDA system we have given them particular attention, and this brought about a certain number of original solutions, for their recording in the base as well as for their use in search operations (inferences). For the recording, we have perfected a notation system which is relatively complex, but enables the representation not only of the 'category of representation' of a temporal marker, that is, its position in relation to the given event, but also of all the imprecisions (incomplete dates, non-direct 'perspectives') which can affect the recording of that marker. In as far as their use goes, the 'preselection' mechanism—which deals with the construction of a first tool enabling access to the information which may fill the requirements imposed by a certain 'search model' and which is one of the essential mechanisms in RESEDA—operates almost exclusively on the semantics of the temporal data appearing in the model and the temporal data recorded in the database.

1. INTRODUCTION

1.1 Until fairly recently, researchers in artificial intelligence (AI) had not paid a great deal of attention to the problems connected with the representation and the use of temporal data; they seemed to prefer to leave this area to the abstract speculation of logicians who, in the last 20 years, have suggested all kinds of special 'temporal logics' (Prior, 1967; Mattison, 1969; Rescher and Urquhart, 1971, etc.). Among the early AI-inspired works on this subject, one can nevertheless quote those of Findler and Chen (1971), Bruce (1972) and Hendrix (1973). Also, it must not be forgotten that researchers interested in the use of AI techniques in the automatic comprehension of natural language have always had to concern themselves, more or less explicitly, with the problems related to the temporal dimension of the discourse. Among the works of this type where the interest in temporal phenomena is particularly prominent I shall quote, as an example, Steedman (1977), Jakob (1978) and Hirschman (1981).

This situation of relative disinterest is now changing rapidly as AI techniques are being applied in concrete areas, since there are few human activities where temporal

0144-817X/83/02 0089-20 $03.00

phenomena can be disregarded. In this context, the need to be able to account for such data appears to be particularly urgent where the application of AI techniques to medicine is concerned: in this area, there are systems ranging from COSTAR (Beaman *et al.*, 1974), TOD (McShane *et al.*, 1979) and OCIS (Blum and Lenhard, 1979), which allow only limited representation and use of temporal information, to systems which are much more ambitious in this subject, of which I shall recall CASNET (Weiss *et al.*, 1978)—where the temporal dimension of pathological phenomena is taken into account by a causal-associational network which summarizes the mechanisms of disease—PATREC (Mittal and Chandrasekaran, 1980), VM (Fagan, 1980), ALVEN (Tsotsos, 1981), MECS-AI (Koyama *et al.*, 1981), etc.

The renewed interest, in an AI context, in the representation and processing of temporal data is much more general, however. Among recent works inspired at least partly by practical preoccupations I shall quote Badler (1975), Kahn and Gorry (1977), Jones *et al.* (1979), Marburger *et al.* (1981), Sembugamoorthy (1981), Grover (1982) and Vilain (1982); and on a more theoretical level, Hayes (1978a, b), Moore (1980), Allen (1981) and McDermott (1982).

An annotated bibliography on 'The role of time in information processing'—which is not only concerned with works of AI origin but also gives place to those inspired by other areas of application, that of databases for example—has recently been brought out by Bolour *et al.* (1982). Anderson (1981) is also the author of an interesting thesis on the semantics of time in a database context; her work is founded on Abrial's conceptual data model.

1.2 The aim of this paper is to provide some general information about the way in which temporal data are represented and used in the RESEDA system. RESEDA is an information retrieval system equipped with 'deep level' (Davis, 1982; Hart, 1982) reasoning capability in the field of complex biographical data management. The term 'biographical data' must be understood in its widest possible sense: that is, any event, in the public or private life, physical or intellectual, etc., that it is possible to gather about the personages we are interested in. In the present state of the system, this information concerns a well-defined period in time (approximately between 1350 and 1450) and a particular subject area (French history). We are at the moment involved in adapting RESEDA's methodology to the processing of biographical data other than historical (medical, legal or military data).

The construction of the RESEDA prototype has been financed by grants from the Délégation Générale à la Recherche Scientifique et Technique (RESEDA/O, CNRS-DGRST Contract n° 75.7.0456), the Institut de Recherche d'Informatique et d'Automatique (RESEDA/1, CNRS-IRIA Contract n° 78.206), and the Centre National de la Recherche Scientifique within the framework of the Action Thématique Programmée Intelligence Artificielle (ATP n° 507568). This prototype is written in VSAPL and is implemented at the Centre Inter-Régional de Calcul Electronique (CIRCE) in Orsay, France.

RESEDA differs from 'traditional' information retrieval systems in three ways:

1. The information stored in the system's database is not in the form of 'documents' in the usual sense of the term ('full text' or bibliographical references) but in the form of 'facts': every 'episode' in the lives of our personages which it is possible to collect and represent. This information is recorded using a particular data definition language ('metalanguage') which

uses knowledge representation techniques.

2. A user interrogating the base obtains not only information which was directly introduced into it but also implicit information found using *inference mechanisms* particular to the system.
3. Temporal information, obviously, has a fundamental importance within a biographical database. We have given it special consideration, which has led to a number of original solutions, concerning its coding as well as the way it is put to use during search operations.

2. THE ENCODING OF TEMPORAL INFORMATION WITHIN RESEDA

2.1 First of all I shall provide a few fundamental ideas about the way in which biographical data are represented inside RESEDA's database.

The biographical information which constitutes the system's database is organized in the form of units called 'coded episodes' or 'planes'. There are several different types of plane; the 'predicative plane', the most important, corresponds to a 'flash' which illustrates a particular moment in the 'life story' of one or more persons. A predicative plane is made up of some of five possible 'predicates' (BE-AFFECTED-BY, BEHAVE, BE-PRESENT, MOVE, PRODUCE), to which one or more 'modulators' may be attached. The modulators' function is to specify and delimit the semantic role of the predicate. Of course, the 'meaning' of the modulator plus predicate is 'defined'—as for all elements of the RESEDA metalanguage—by the general behaviour of the system rather than by the usual function of these codes in natural language. The predicate of the plane is accompanied by 'case slots' (Bruce, 1975; Rosner and Somers, 1980) which introduce the arguments of the predicate. Dating and space location information is also given with a predicative plane, as is the bibliographic authority for the statement.

The extremely simple example given in Figure 1 should provide a clearer idea of what I have just explained. It is the representation of 'Montreuil was opposed, between 1413 and 1416, to the Burgundian party on the subject of the Hundred

```
1) against + BEHAVE SUBJ  Montreuil : Paris
                    OBJ   burgundians
                    ARG   hundred-years-war
                    date1 : 1413
                    date2 : 1416
                    bibl. : Valois, IV
```

FIG. 1. Formal representation of an elementary episode

Years War'; the bibliographical authority is the historian Valois. The codes given in capital letters indicate the predicate and the cases associated with it: 'against' is a modulator; 'date1–date2' is a pair of temporal markers which give the duration of the episode. 'Montreuil' is a '*vedette*', i.e., one of the historical personages whose 'life story' is recorded in the database; 'burgundians' and 'hundred-years-war' are entries in RESEDA's lexicon, which provides the historical background of the system. 'Paris' is obviously the 'subject location'.

The elementary episodes recorded in the base can obviously be linked together; if the original historical documents give us some explanation of the Montreuil's attitude expressed in plane 1, we must be able to explicitly mark in the base a 'causality' relationship between the coded episodes which translate this explanation and plane 1. The logical links between coded episodes which can be deduced directly from the original documents are represented in RESEDA by a system of labelled pointers. Further on, in 2.4, I shall give a few details about the use of pointers which together define the particular 'taxonomy of causality' (Wilks, 1975, 1977) which has been adopted for RESEDA.

2.2 If we now move on to provide some details on the way information concerning the dating of episodes is represented in RESEDA, the first thing to be specified is that the predicates of the metalanguage that I have just introduced always translate a 'state'. The label 'PRODUCE' can then, for example, be considered as an abbreviation of the expression 'to be in a productive state', 'MOVE' for the expression 'to be in a state of movement', etc. This means that all elementary information (planes) recorded in the base—for example plane 1 in Figure 1—represents a static, isolated situation in the course of the life of one or more personages, a 'flash' which has a shorter or longer duration, specified by the dating elements (date1–date2) which accompany it.

The relationship between the state expressed by the episode and the temporal information which specifies its duration can assume various forms.

2.21 In the simplest case, such as the one in Figure 1, the state is taken 'in its entirety', in as much as we do actually have two temporal markers t1 (1413) and t2 (1416) which allow us to localize—with a degree of precision varying in each case, see 2.3—the beginning and end of the state. The situation is thus as in Figure 2. In this case, the two date blocks of the episode are filled with the two temporal markers

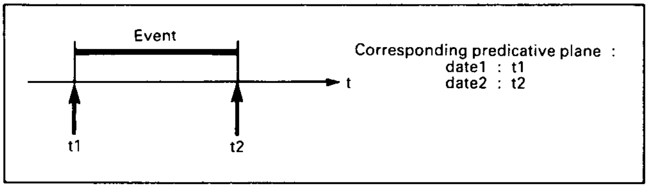

FIG. 2. Temporal information specifying an episode of the type 'state taken in its entirety'

t1 and t2, *see again* Figure 1. The first date block is said to be represented 'in posteriority'—the state of the episode begins at the date indicated in this block—and the second, which constitutes the 'upper' temporal limit of the state, is represented 'in anteriority'.

Of course, it is possible for the episode to be coded to be of the type 'state taken in its entirety' but that, in fact, the documents we have only provide one of the two temporal markers needed to identify it (or, even, neither of them). In this case, only one of the two date blocks will be filled with actual temporal information (markers t1 or t2), while the other (or both) will be filled with a special code, code 'dash', '—'. Note that this code has a well-defined meaning, that of 'a temporal marker that we should know but is not accessible temporarily because of the state of the available documentation', and is thus conceptually totally different from the 'empty' code, *see* 2.22, which means the logical impossibility of filling a date block.

2.22 It is often necessary to record an event in which only one of the two limits of the corresponding state is to be used, be it t1 or t2—for example, when it is necessary to supply special information about the circumstances at the beginning or end of a state; another possibility is that only an intermediary date, between t1 and t2, is known. In all these cases, only the first date block date1 is filled with the single date available, the second date2 being filled with the 'empty' code. These three cases are differentiated by one of a particular class of modulators, the 'temporal modulators', which, as usual, is added to the predicate of the coded episode. The beginning of a state (marker t1) will be represented by using the modulator 'begin', and the filled date block date1 is said to be represented 'in posteriority'. To indicate the end of a state (marker t2) the modulator 'end' is used, and the date block date1 is said to be represented 'in anteriority'. If, finally, a particular moment within a state is to be indicated—for example, to express the information that it is known that in June 1415 Montreuil opposed the Burgundians on the subject of the Hundred Years War, but without, at this level, giving any information about the beginning or end of this state of opposition, which extends beyond the stated date—the modulator 'const' (from the French *constater* = to find, to recognize) is used. The modulated predicate 'const + against + BEHAVE', in conjunction with a coding of the type 'date1: June-1415; date2: ', will be used to represent this statement of the relationship between Montreuil and the Burgundians at this particular time. The filled date block date1 is now said to be represented 'in contemporaneity'.

The three cases illustrated in the preceding paragraph are summarized in Figure 3.

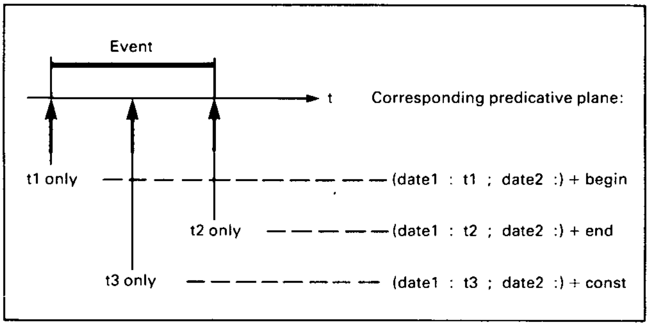

FIG. 3. A particular moment during an event is singled out

2.23 Figure 4 illustrates the case where a 'punctual state' is associated with events of the type 'On 15th July 1394, Montreuil wrote a letter to his friend Clamanges...', where the writing of the letter probably took only a brief moment during that day, and thus the episode seems to be concentrated around a point inside the interval of time—in this case, 15th July—theoretically associated with the given temporal marker. The situation is in some aspects symmetrical with that characterized by the use of the modulator 'const'; on the other hand, to express the information 'It is known that Montreuil used the whole of 15th July 1394 to write the letter . . .' both date blocks date1-date2 would be filled with the same information '15-july-1394'.

In a coded episode of the punctual type, the only date block that is filled goes with the absence of special temporal modulators. The date block is represented 'in contemporaneity'.

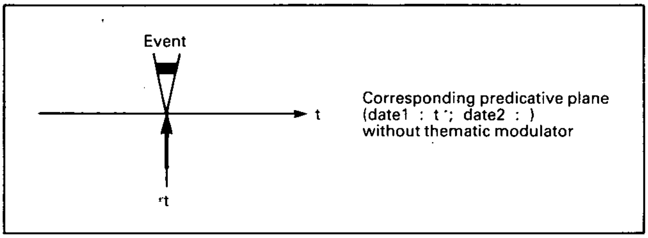

FIG. 4. 'Punctual state'

2.3 As a result of the fundamental decision to think of predicates as 'state predicates', the representation of RESEDA's temporal data is therefore necessarily a representation in terms of intervals. An identical philosophy has been adopted by other researchers, Allen (1981) and Vilain for example—'. . . intervals are the most computationally natural way of representing time' (Vilain, 1982: 197). With regard to the notions associated with the codes 'begin', 'end', 'const', etc., they already exist in Findler and Chen (1971); on this subject, *see again* Vilain (1982), etc.

To summarize now what was explained in 2.2, one could say that to every temporal marker t corresponds a filled date block, and vice versa. The place of the temporal marker in relation to the state associated with the event we are coding defines the 'category' of the dating: anteriority, contemporaneity or posteriority, associated with the date block. Thus it is this date block, a translation of the temporal marker to be represented, which 'localizes', in a first approximation, the given event on the time axis.

Whenever the temporal marker to be represented and the associated dating category is to be decided, there is the problem of taking into account the accuracy, or rather the lack of accuracy, with which this marker is known. This accuracy can be of two types: indirect 'perspectives' and omissions in the dates. The solutions adopted in RESEDA to handle this type of problem are, in my opinion, sufficiently original.

2.31 The 'perspectives' correspond to different ways of 'capturing' a temporal marker. To give an intuitive idea of the concept of 'perspective', I shall draw up a list of the usual ways of representing a date taking once more the example of letter A, a brief note written on a single day:

1. Letter A bears a date written by the author: the dating is 'direct' (15th July 1394).
2. Letter A does not bear a date, but mentions the celebration of a feast the date of which is known (without saying if the feast has passed or is to come): letter A is thus situated around that date, circa (around 8th July 1394).
3. Letter A does not bear a date but mentions receiving a letter B, which is dated: this date is thus a *terminus a quo* for letter A (after 1st July 1394).
4. Letter A does not bear a date, but is mentioned in a letter C, which is dated: this date is thus a *terminus ante quem* for letter A (before 2nd August 1394).
5. Letter A does not bear a date, but we know of both letter B and C (as defined above): these two dates provide an 'inclusion fork' for letter A (between 1st July and 2nd August 1394).

In each of these cases, the category of the temporal marker remains the same (contemporaneity), but not the precision (perspective) with which the marker (date block) is recorded: the type of perspective which affects the temporal indications of a particular date block is completely independent of the category according to which the date block is represented. In a piece of information of the type 'It is known that Clamanges was pontifical secretary during the period whose first limit is between April and July 1347, and second between October and December 1403', the corresponding coded episode would have both date blocks filled, the first 'in posteriority' and the second 'in anteriority', but the perspective would for both be of the 'fork' type.

Thus if we exclude direct dating, these perspectives all specify a range of possible dates for the temporal marker to be coded. The only thing which distinguishes them from each other is the way in which this range is indicated by the documents analysed (*see* Fig. 5); we can see that it is always a 'fork' and that the source can specify both limits of this fork, one only (*a quo* and *ante quem*), or none of them (*circa*).

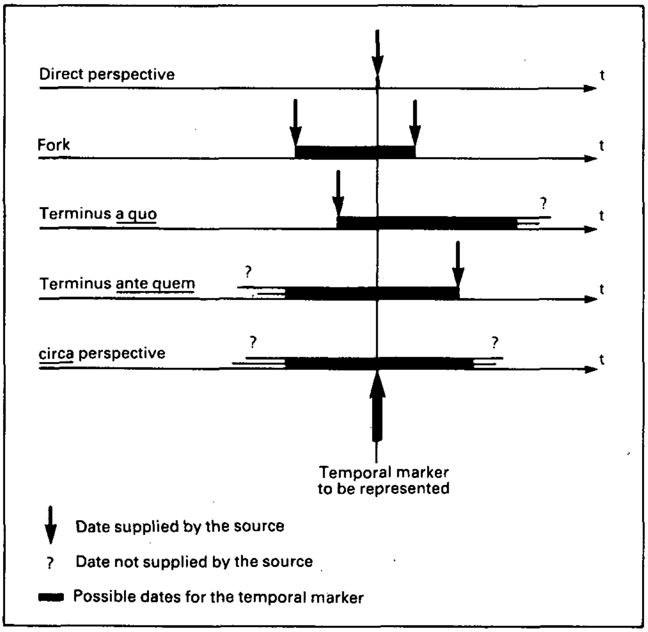

FIG 5. Various forms of imprecision ('non-direct perspectives') in the representation of a temporal marker

When the limits are not both provided by the source, the problem is to find a way of palliating this lack, since, for example, knowledge of the possible dates for a temporal marker to be represented is necessary to determine whether a given episode is relevant, as far as the temporal information is concerned, vis-à-vis a user's query. Such a reconstruction could be executed automatically, at the time of research within the database, using some algorithm designed to calculate 'so many days' or 'so many months' before and/or after the effective date given by the historical source. Such a numeric, even parametered, threshold would always have the inconvenience of working 'blind', not knowing the context of the known facts in the base. A reconstruction 'by inference', taking this context into account, would undoubtedly be the ideal solution; it would, however, present considerable difficulties in the establishment and the formalization of the rules to be implemented, and probably would not be very efficient from a computational point of view.

This is why it seemed desirable, at least for the time being, to ask the documentalist who provides the primary data to be introduced into the base to undertake this reconstruction himself, that is, each time it is necessary to record an episode characterized by non-direct perspectives, to replace the question marks in Figure 5 with estimations. In effect, he has at his disposal the context of the event to be coded and his personal knowledge, which together constitute much better approach criteria than could be supplied by the most elaborate algorithm.

Thus we can conclude by stating that, each time we have to deal with non-direct perspectives, the date blocks characterized by these perspectives in fact contain two dates, giving the limiting values of a fork inside which is situated the temporal marker associated with the block. A date that is reconstructed—i.e., one that is not found in the source being analysed—will be of course distinguished from the original dates, in order to respect the integrity of the indications of the source. On the other hand, date blocks corresponding to markers coded in direct perspective only contain one date (the original date).

2.32 With regard to the problem of 'lacunary dates', it is obvious that if a date is normally defined by a <year, month, day> triplet, those associated with biographical data do not always have their three elements known simultaneously. Thus in RESEDA we have to deal, in many cases, with lacunary dates.

If we agree to denote any element (year, month or day) that is 'full' or known by *f* and any element that is 'empty' or unknown by *e*, there are obviously four types of possible configuration for a date:

1. Date completely known: *fff*.
2. Date with the day unknown: *ffe*.
3. Date with the month unknown: *fef* and *fee*.
4. Date with the year unknown: *eff*, *efe* and *eee*.

I shall not dwell here on the solutions used in RESEDA to solve this type of problem, since the solutions again involve, in many cases, a 'reconstruction' of the missing elements which is done by the documentalist; for more details, *see* Zarri *et al.* (1980: 42–44). For example, dates with the year unknown are not permitted in as much as the historian *must*, at the time of coding, specify the set of years which are reasonably possible, by making use of the context provided by the source that is analysed and/or his own knowledge. *f*xx configurations thus obtained are treated separately as dates with the year not missing.

2.4 I have limited myself to giving, in the preceding paragraphs, information about the more 'visible' characteristics of RESEDA's system of temporal data representation, concentrating in particular on the notions which are absolutely necessary to be able to follow the arguments in Section 3.2. But it is obvious—given the type of information which RESEDA is called on to process (biographical information)—that the 'time' phenomenon is deeply impregnated into the whole system, and that—besides the explicit coding of temporal data which I have just been talking about—a form of 'implicit' representation of this type of information is very often associated with other mechanisms of RESEDA.

The links between coded episodes created by labelled pointers, *see* 2.1, are obviously one of the important means of conveying this implicit information. Thus, the pointer ASSOC(iation) enables all the coded episodes belonging to a single logical unity (for example, all the various vicissitudes of a trial) to be traced back to the plane or group of planes which is/are at the origin of this sequence (for example, the coded episode relating to the lodging of a complaint). There is an obvious analogy between this mechanism and that of the 'reference events' of Kahn and Gorry (1977), and I shall recall that one of the functions of the 'reference events' is that they are used to time other events.

The function, associated with the labelled pointers, of implicit indicators of temporal relationships becomes particularly clear when one considers the group of pointers which together define the taxonomy of causality particular to RESEDA; as is noted by Mittal and Chandrasekaran, for example (1980)—but *see also*, Weiss *et al.* (1978), etc.—temporality and causality are always strictly associated. Figure 6 shows some details of the rules governing this system of pointers. Intuitively, the taxonomy is linked to a double dichotomy in the causal field: (a) cause—consequence; (b) weak—strong. But this distribution is in fact based on more objective criteria, shown in Figure 6. From this point of view, the first criterion of the dichotomy is recognizing if the explicative argument—plane (b), introduced by one of the correlator-pointers of the field—is previous to (= CAUSE or CONFER) or subsequent to (= FINAL or MOTIV) the argument that it explains—plane (a)—to which the causal expansion is attached. The second criterion of the dichotomy is based on the traditional distinction between 'necessary' and 'sufficient'; the coexistence of the two elements characterizes a strong causality (= CAUSE or FINAL) whilst the existence of only one of the two implies weak causality (= CONFER or MOTIV).

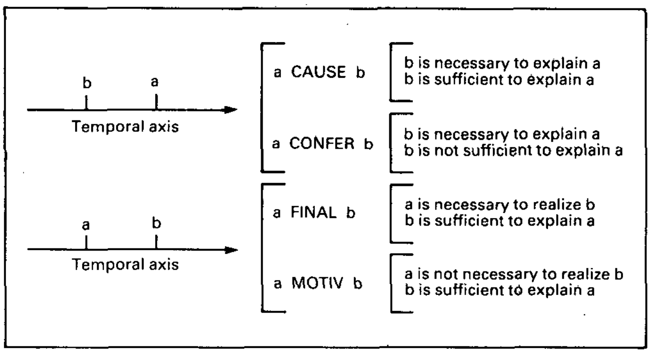

FIG. 6. 'Taxonomy of causality' proper to the RESEDA system

3. THE USE OF TEMPORAL DATA IN SEARCH OPERATIONS INSIDE THE BASE

3.1 I shall first of all quickly review the general characteristics of the research procedures implemented in RESEDA.

When the system is considered from the point of view of its utilization, the fundamental concept which must be introduced is that of the 'search model'. A 'search model' gives the essential elements, expressed in terms of the RESEDA metalanguage, of a coded episode which it is necessary to search for in the database.

A search model may originate from outside the system, if it is the direct translation of a query posed by a user. On the other hand, it may be automatically generated by the system during the execution of an inference procedure.

Let us suppose, for example, that a user is interrogating the system about the relationships between Montreuil and the Burgundians party concerning the Hundred Years War. In this case, the user himself creates the search model given in Figure 7, with the aid of a prompting program. The only notable difference between this formalism and the formalism required for the representation of the episodes in the database (*see* Fig. 1) is that of the presence of a 'search interval', 'bound1–bound2', which the user will employ to limit the period he considers appropriate to explore according to the information for which he is searching. Therefore, the search interval has the function of limiting the planes to be examined, and has only an indirect relationship with the temporal information of the 'date' type which is associated with each of the episodes recorded in the base.

```
BEHAVE  SUBJ     Montreuil
        OBJ      burgundians
        ARG      hundred-years-war
        bound1 : 1400
        bound2 : 1420
```

FIG. 7. Simple 'search model' enabling the retrieval of the plane in Figure 1

In this case, it is obvious that the search model in Figure 7 may be directly matched with the plane 1 in Figure 1; this, of course, is the exception rather than the rule. In the case of a dead end, a first class of inference rules may be applied to the model, the 'transformations'.

To keep to an extremely simple example, consider the transformation of Figure 8, allowing us to change a search model formulated in terms of 'end + BE-PRESENT' into a new one in terms of 'MOVE', which can be submitted, in turn, to the usual match procedures. This formal rule translates the common sense rule 'If someone goes from one place to another, he has certainly left his starting point': the justification of the use of substitution in Figure 8 lies in the fact that any information about some personage *x* having moved from *k* to *l* is at the same time a response to any query about the possibility of his no longer being at place *k*. Note that, in the terms of RESEDA's metalanguage, the movements of a personage are always expressed in the form of a subject *x* which moves itself as an object. The values which replace the variables in the retrieved plane (or planes) using the transformed model must obviously respect the 'restrictions' associated with all the variables which appear in the transformation.

```
t1) end+BE-PRESENT SUBJ x : k ➤ MOVE SUBJ x : k
                                      OBJ x : l

x = <personage>
k,l = <location>
k ≠ l
```

FIG. 8. Simple inference rule of the 'transformation' type

A second specimen of transformation is that given in Figure 9; the common sense rule underlying this formalism is: 'If a person *x* has a university degree *w*, then this person has followed some course *v*' (one or several persons *y* have 'produced' the course *v* with the intention of *x*). '$w = f(v)$' is an abbreviated way of expressing that there must be coherence between the diploma obtained and the courses followed.

```
t2) PRODUCE SUBJ y ➤ BE-AFFECTED-BY  SUBJ x
            OBJ  v                   OBJ  w
            DEST x

x = <personage>
y = <personage>|<personages>
x ≠ y
v = <university-course>
w = <degree-obtained>
w = f(v)
```

FIG. 9. More complex 'transformation' rule

Very often the passage from one model to another is subject to certain conditions; it is only possible to substitute one model for the other if a particular condition has been satisfied ('conditional transformations'). This is verified by checking the existence of episodes within the base which are able to guarantee the appropriate context.

Even when taking into consideration this first category of inference rules, the behaviour of the system such as it has been described up to now is entirely classic in type. RESEDA has, however, a second, more original method of research: it is possible to search for the implicit 'causes', in the widest sense of the word, of an attested fact in the base. For example, if the user, in submitting the query in Figure 7, obtained in reply the plane in Figure 1, and if we assume that the 'causes' of this negative attitude of Montreuil towards the Burgundians are not explicitly recorded in the database, he will now be able to ask the system to automatically produce a *plausible* explanation of this attitude by using a second category of inference rules, the 'hypotheses'.

In order to give a first idea, on an intuitive level, of the functioning of the hypotheses, Figure 10 shows the formulation in natural language of four typical hypotheses of the RESEDA system.

h1) . . . one might cease to act on behalf of some other person
BECAUSE
one has abused that person's confidence (e.g by misrepresenting his views to a third party)

h2) . . . one might leave something (in one's will) to a (religious) community
BECAUSE
one had some special connection with this community

h3) . . . one might take a particular attitude in an argument
BECAUSE
one has close links with one of the parties in a conflicting situation

h4) . . . one might lose a position (of civil servant)
BECAUSE
one is replaced by a supporter of the (political) party which has just taken power

FIG. 10. Formulation in natural language of four characteristic 'hypothesis' rules

The first part of each of these rules corresponds to a particular class of confirmed facts (planes) for which one asks the causes. For example, the plane in Figure 1 is clearly an exemplification of the first part of the hypothesis h3 in Figure 9. In RESEDA's terminology, the formal drafting of this part is called a 'premiss'. The second part (the condition) gives instructions for searching the database for information which would be able to justify the fact which has been matched with the premiss. That is, if planes matching the particular search models which can be obtained from the 'condition' part of the hypothesis can be found in the database, it is considered that the facts represented by these planes *could* constitute a justification for the plane premiss and are then returned as the response to the user's query. An important point to notice is that search models generated by the hypotheses, like any search models, can also be *transformed* in the case of an initial failure of the match procedures.

In the case of the enquiry about the causes of Montreuil's attitude, the search model generated by the hypothesis h3 in Figure 10 enable the retrieval from the biographical base of the planes in Figure 11: plane 2 translates the fact that Montreuil (and Gontier Col) were amongst the Armagnac rank between 1400 and 1415, while plane 3 tells us that the Armagnacs and the Burgundians were in conflict between 1407 and 1425 (recip = recip(rocity)).

```
2) BE-AFFECTED-BY SUBJ  armagnacs
                  OBJ   (COORD1 Montreuil Col)
                  date1 : 1400
                  date2 : 1415
                  bibl.  : Le-Duc

3) recip + against + BEHAVE  SUBJ  (COORD1 armagnacs burgundians) :
                                        FRANCE
                             OBJ   (COORD1 armagnacs burgundians) :
                                        FRANCE
                             date1 : 1407
                             date2 : 1425
                             bibl:  : consensus
```

FIG. 11. Predicative planes provided by the hypothesis h3 in Figure 10 as a reply to the question concerning the causes of Montreuil's attitude (plane 1 in Fig. 1)

Details about the process of constructing the system's inference rules, and in particular the hypotheses, by 'simulating' a historian's reasoning, can be found in Zarri (1979, 1981).

3.2 From what I have illustrated in the preceding section, it seems clear that the problem of associating data in the base with a search model—produced directly by a user's question or generated by the system in the case of an inference procedure, hypothesis or transformation—is one of the chief problems to be solved so that the system will function correctly.

The strategy followed has been to divide the match process into two phases:

1. A 'preselection' phase, which uses the techniques of access to the base that are part of the 'index level' of the system (*see* 3.21) in order to select—by means of, amongst others, the correspondence between the dates indicated in the search interval and those encoded in the episodes of the base and transferred to the

index level—a preliminary list of coded episodes which are likely to satisfy the model in question.

2. A match phase 'proper', in which the episodes retained during the preselection phase are effectively matched against the model.

The reason for adopting a two-step strategy is mainly linked to the fact that the syntactic structure of the coded episodes is normally far more complex than it appears at a first glance when examining the very simple examples given in Figures 1 and 11. Thus, it is an advantage to delay the match with the model until it is reasonably certain that a group of episodes *a priori* relevant has been isolated.

3.21 The index level organization has been explained in detail in Zarri *et al.* (1980); I will restrict myself here to giving certain general principles.

The information taken into account at this level includes:

1. The personages implied in the base.
2. The predicate (BE-AFFECTED-BY, BEHAVE, BE-PRESENT, MOVE, PRODUCE) in the coded episodes.
3. The temporal information.

This information is organized in the following way:

1. To every personage implied in the base corresponds a 'primary index', by means of which it is possible to find all the predicative planes in which that personage appears.
2. To every predicate corresponds a 'line' of the primary indexes.
3. For a given personage and predicate, the dating category of the stored episodes is taken into account at the index level by dividing the temporal markers into three 'groups', according to whether they are represented in anteriority, in contemporaneity or in posteriority.
4. The dating perspective of these markers is taken into account, inside each group, by separating these dates into three 'lists' according to whether they are: a direct date (list DD), the first limit of a fork (list F1), or the second limit of a fork (list F2). Each marker used inside an episode will thus insert a reference to this episode either once in list DD, or once in list F1 and once in list F2, depending on whether the marker was represented by a direct date or by a fork (non-direct perspective).
5. Each element of these lists is a 'doublet' comprising:

   (a) A date <year, month, day>.
   (b) The address of a predicative plane.

Access to	By selecting
The personage	The corresponding primary index
The predicate	A line in this primary index
The category	A group in this line
The perspective	A list in this group
The date	A doublet in this list
The episode	(Direct access by the second element of the doublet)

FIG. 12. Criteria for access to information stored in system's 'index level'

In brief, we gain access to the information retained at preselection level as shown in Figure 12.

As far as internal representation is concerned we have:

1. For a primary index, a matrix with fixed structure shown in Figure 13, providing access to the secondary indexes (each of the forty-five elements of the primary index, defined by a predicate, a category, and a perspective, is the address of a secondary index).

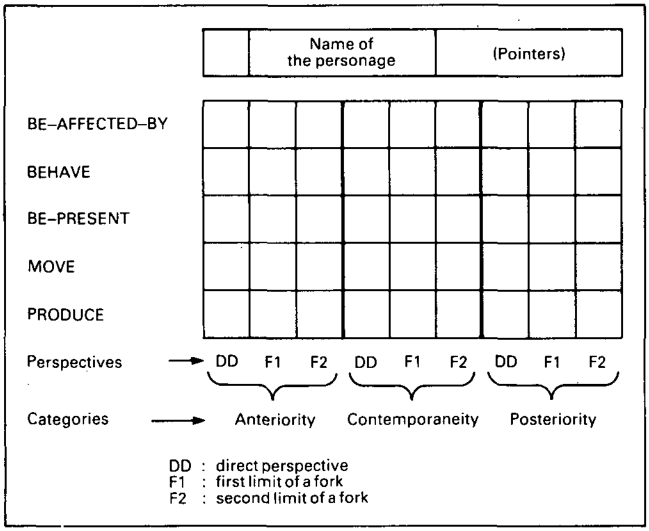

FIG. 13. Organization of primary index

2. For a secondary index, a matrix with variable structure—two lines and as many columns as there are doublets in the corresponding list—providing access to the predicative planes.

Thus we obtain, in a natural manner, a three-file organization for the biographical data in the base: primary indexes, secondary indexes, planes.

In order to clarify what I have just stated, I shall explain how plane 2 of Figure 11 is taken into account at the index level. This episode is concerned with the indexes of 'Montreuil' and 'Col' in the primary indexes file, during the same period and with identical modalities; so I will limit myself to explaining the treatment of the index for Montreuil.

The predicate BE-AFFECTED-BY indicates that the representation at the primary indexes level of the episode only concerns the first line of the 'Montreuil' index. This episode comprises two date blocks, the first represented 'in posteriority' and the second 'in anteriority', which stops the central three columns being brought into play. The perspective of each block is direct: each of them, therefore, only carries a single date and not two (the first and second limits of a fork), as would be the case for non-direct perspectives. Finally, the treatment of the episode in the primary indexes deals with the 1st (row 1, column 1) and the 7th (row 1, column 7) elements of the 'Montreuil' matrix.

The treatment of plane 2 in Figure 11 at the secondary indexes level will include:

1. The insertion of a doublet—consisting of the date '1400' and the address of the episode in the file of the biographical database proper—into the list of a secondary index corresponding to the 7th element of the 'Montreuil' matrix and that of the 'Col' matrix of the primary indexes (BE-AFFECTED-BY + posteriority).
2. The insertion of a doublet—consisting of the date '1415' and of the same address in the biographical database—into the list corresponding to the 1st element of the 'Montreuil' matrix (ditto 'Col') of the primary indexes (BE-AFFECTED-BY + anteriority).

If plane 2 had included a date block, for example the first, characterized by a non-direct perspective, it would have been necessary to insert six doublets (three for 'Montreuil' and three for 'Col') into the lists of the secondary indexes because each of the dates constituting the fork, which has translated the non-direct perspective, would have given way to a doublet in the secondary lists addressed by the 8th (first limit of the fork) and 9th (second limit of the fork) elements of the 'Montreuil' (and 'Col') matrix of the primary indexes.

3.22 The preselection algorithm (or rather the set of preselection algorithms) must confront very complex logical problems in order to finally isolate the doublets to be retained.

Indeed, it is not sufficient to affirm that a doublet whose first element specifies a date within the limits of the search interval will be selected. Categories and perspectives also have a role to play; to give just a very banal example, consider the coded episode which translates the information 'Robert de Bonnay was named bailiff of Mâcon on 27th September 1413', classed at the level of the secondary indexes in a single list 'direct perspective–posteriority' (the date supplied in the filled date block of the episode, which is accompanied by the thematic modulator 'begin', specifies, in effect, the beginning of a state). A search model of the type 'Was Robert de Bonnay bailiff of Mâcon in the period delimited by the years 1415–1420?' will, in general, have to select the doublet <date, address> corresponding to the above episode, even if this date is outside the search interval specified by the model under scrutiny. Indeed, in the absence of data in the biographical base which explicitly affirm the opposite, it has to be supposed, conforming to the notion of 'posteriority', that Robert de Bonnay was still the bailiff during the years indicated by the model.

It is not possible, given the length of this article, to give a detailed description of the preselection algorithms. I shall therefore restrict myself to presenting here, with the help of Figure 14, an intuitive formulation of the algorithm which must be applied in the case where there is no temporal modulator (begin, const, end) in the search model (*see* the model in Fig. 7), for example. The formal definition of these algorithms is given in Zarri *et al.* (1980: 54–70); details of their implementation in APL are given in Zarri *et al.* (1981: 87–103).

In Figure 14, each 'list' is the symbolic representation of a secondary index, that is, of a sequence of doublets <date, address>. The six lists represented correspond to a row of primary index: the list aDD is therefore the list 'anteriority–direct perspective', the list aF1 'anteriority–first limit of a fork', the list cDD 'contemporaneity–direct perspective', etc. (*see* Fig. 13). The doublets of the secondary indexes are

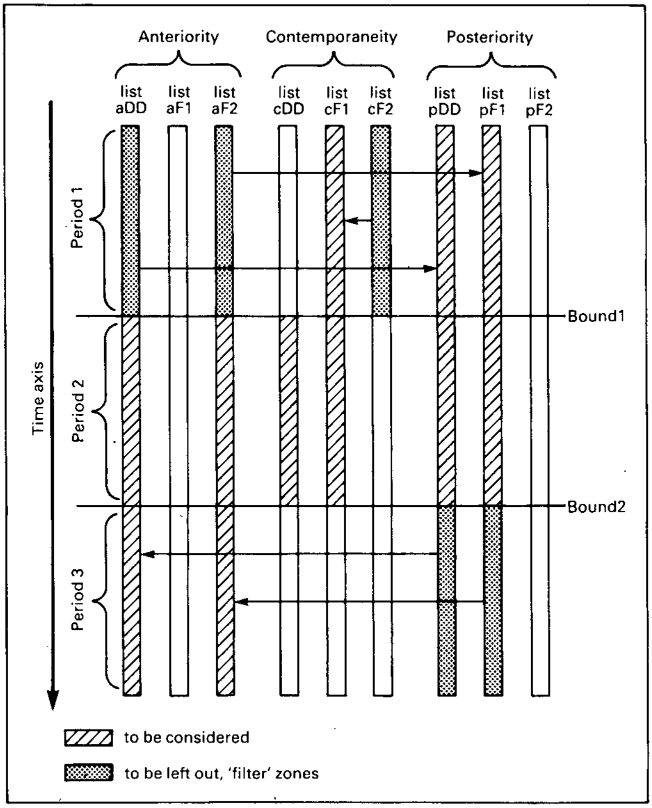

FIG. 14. Graphical representation of preselection algorithm which must be applied when there is no 'temporal modulator' in the search model

recorded in increasing order of the dates. The nine lists are therefore divided by the two limits 'bound1–bound2' of the search model being tested into the three 'periods' (before bound1; between bound1 and bound2; after bound2) as shown in Figure 14.

If the model has no thematic modulators, the planes to be selected from the actual biographical base are those which, in general, satisfy the following conditions:

1. The planes whose addresses are associated with doublets of the secondary index which are in period 1 in posteriority, but not in anteriority.
2. The planes specified by doublets in period 2 whatever their category.
3. The planes specified by doublets in period 3 in anteriority, but not in posteriority.

This corresponds to the intuitive notion that, in the case of the model in Figure 7, for example, the episodes classed in posteriority (*starting from* . . .) concerning dates before 1400 (bound1), and those classed in anteriority (*until* . . .) concerning dates after 1420 (bound2) must be retained; *see also*, at the beginning of this section, the example of the nomination of Robert de Bonnay. I shall now give a more detailed explanation.

If we imagine first of all that the base only contains episodes in direct perspective (lists aF1, aF2, cF1, cF2, pF1 and pF2 are empty), parts of lists of type DD to be considered for a search model with no thematic modulators are selected according to the following criteria:

1. The part of the list cDD corresponding to period 2 is definitely to be selected: the episodes characterized by a punctual date located within this period in fact take place, at least partly, inside the search interval.
2. The part of the list aDD corresponding to period 2 is also to be selected: the corresponding episodes definitely end after bound1 and before bound2, and are therefore located, at least partly, inside the search interval. The part of the list pDD corresponding to the same period is also to be considered, since the corresponding episodes definitely begin inside the search interval.
3. The part of the list aDD corresponding to period 1 is to be excluded, since it refers to episodes which definitely end before bound1; similarly, episodes in the list pDD corresponding to period 3 are to be excluded, since they definitely begin after bound2. Episodes in the list aDD in period 3 (which end after bound2) are potentially to be selected; but before accepting one of these episodes, it must be checked that the same episode does not also appear in the element of the list pDD in period 3 (this episode would then begin, and end, after bound2). A situation of this type could be, with regard to the search interval in Figure 7, that of a plane 1 in which the date blocks would carry the information 'date1:1427; date2:1430'; this plane would therefore be classed in period 3 'in anteriority' (1430) as well as 'in posteriority' (1427). This situation can be compared to that of a plane carrying the information 'date1:1418; date2: 1427' which, on the other hand, must be accepted. The part of the list aDD in period 1 therefore acts as an 'exclusion filter' for the corresponding part of the list pDD (the episodes which are classed in both these two parts must not be preselected); the same for the part of the list pDD in period 3 with respect to the corresponding part of the list aDD.
4. At the end of the algorithm, the addresses of these planes which would have been selected twice since they are at the same time present in the 'right' sections of the lists aDD and pDD must obviously only be considered once (*see*, for example, the plane 1 in Fig. 1).

To extend this algorithm to fit the general case of a search model with no temporal modulators, that is, in considering the lists relative to the non-direct perspectives to be non-empty, the following considerations must be added:

1. The same group of plane addresses appears in the doublets associated with the lists aF1 and aF2 (the same for the lists cF1–cF2 and pF1–pF2): if a plane contains in fact a fork in its second date block (classed in anteriority), its address will appear in the list aF1 associated with the date representing the first bound of the fork, as well as in the list aF2 associated with the date representing the second bound of the fork.
2. We can then avoid considering the lists aF1 and pF2: the 'deciding conditions' for anteriority in fact will be tested on the one list aF2, in which the dates are 'lower down in the list' in comparison with those contained in the corresponding doublets in the list aF1; the deciding conditions for posteriority will be tested on the list pF1, in which the dates are 'higher up'. If we consider, with regard to the model in Figure 7, a plane carrying the following date information: 'date1:

between 1398 and 1403; date2: between 1418 and 1423', the dates which will finally be examined for the preselection will be '1398' (posteriority) and '1423' (anteriority).

3. If we take the same argument for the lists aF2 and pF1 as we did for the corresponding DD lists, we see that the parts to be selected are those of the list aF2 which belong to periods 2 and 3, and those of the list pF1 which belong to periods 1 and 2. The part of the list aF2 in period 1 and that of the list pF1 in period 3 are definitely to be excluded; they act,. however, as 'exclusion filters' with regard to the part of the list pF1 in period 1 and to that of the list aF2 in period 3, respectively.
4. As for contemporaneity, it can easily be shown that the parts to be selected are those of the list cF1 in periods 1 and 2; the part of the list cF2 in period 1 acts as an 'exclusion filter' with regard to the part of the list cF1 in period 1.

What I have just explained is summarized in Figure 14. The 'hatched' areas represent the list parts to be considered; the 'dotted' areas represent parts which are definitely to be excluded, they also represent exclusion filters with regard to the arrowed parts.

In the case of preselection for a search model containing a thematic modulator 'begin' it can be shown (for example, *see* Zarri *et al.*, 1980: 66) that the parts to be considered only occur in p lists, that is, pDD in period 2 and pF1 in periods 1 and 2; the part of pF2 in period 1 is definitely to be excluded, but it acts as a filter for the part of pF1 in period 1; etc.

4. CONCLUSION

What I have just illustrated can only confirm that temporal data are of fundamental importance in a biographical database. In the RESEDA system we have given them particular attention, and this has brought about a certain number of original solutions, concerning their recording in the base as well as their use in search operations.

For the recording, we have perfected a notation system which is relatively complex, but enables the representation not only of the category of representation of a temporal marker, that is, its position in relation to the given event, but also of all the imprecisions (incomplete dates, non-direct perspectives) which can affect the recording of the marker.

In as far as their use goes, I have shown how the preselection mechanism—which deals with the construction of a first tool enabling access to the information which may fill the requirements imposed by a certain search model, and which is one of the essential mechanisms in RESEDA—operates almost exclusively on the semantics of the temporal data on the one hand appearing in the model and on the other hand recorded in the database.

ACKNOWLEDGMENTS

My thanks to R. LeLouche, who has perfected the logic of the preselection algorithms, and to G. Lee, who has carried out the coding of these algorithms.

REFERENCES

Allen, J. F. (1981) *Maintaining knowledge about temporal intervals.* Rochester (NY): University of Rochester Computer Science Department (Technical Report 86).

Anderson, T. L. (1981) *The database semantics of time.* Washington: University of Washington Computer Science Department (Ph.D. thesis).

Badler, N. I. (1975) *Temporal scene analysis: conceptual description of objects movements.* Toronto: University of Toronto Computer Science Department (Technical Report 80).

Beaman, P. D., Justice, N. S. and Barnett, G. O. (1979) A medical information system and data language for ambulatory. *Computer 12* (11), 9–17.

Blum, B. I. and Lenhard, R. E., Jr. (1979) An oncology clinical information system. *Computer 12* (11), 42–50.

Bolour, A., Anderson, T. L., Dekeyser, L. J. and Wong, H. K. T. (1982) The role of time in information processing: a survey. *ACM Sigart Newsletter 80*, 28–48.

Bruce, B. (1972) A model for temporal references and its application in a question answering program. *Artificial Intelligence 3*, 1–25.

Bruce, B. (1975) Case systems for natural language. *Artificial Intelligence 6*, 327–360.

Davis, R. (1982) Expert systems: where are we? And where do we go from here? *The AI Magazine 3* (2), 3–22.

Fagan, L. M. (1980) *VM: representing time-dependent relations in a medical setting.* Palo Alto (Calif.): University of Stanford Computer Science Department (Ph.D. thesis).

Findler, N. and Chen, D. (1971) On the problem of time retrieval, temporal relations, causality, and coexistence. *Proceedings of the Second International Joint Conference on Artificial Intelligence—IJCAI/71 (London 1971).* Ann Arbor (Mich.): University Microfilms International.

Grover, M. D. (1982) A synthetic approach to temporal information processing. *Proceedings of the National Conference on Artificial Intelligence—AAAI/82 (Pittsburgh 1982).* Menlo Park (Calif.): The American Association for Artificial Intelligence.

Hart, P. E. (1982) Directions for AI in the eighties. *ACM Sigart Newsletter 79*, 11–16.

Hayes, P. J. (1978a) *The naive physics manifesto.* Genève: Institut pour les études sémantiques et cognitives (Working Paper 34).

Hayes, P. J. (1978b) *Naive physics I: ontology for liquids.* Genève: Institut pour les études sémantiques et cognitives (Working Paper 35).

Hendrix, G. G. (1973) Modeling simultaneous actions and continuous processes. *Artificial Intelligence 4*, 145–180.

Hirschman, L. (1981) Retrieving time information from natural-language texts. *Information Retrieval Research.* (R. N. Oddy, S. E. Robertson, C. J. van Rijsbergen and P. W. Williams, eds.) pp. 154–171. London: Butterworths.

Jakob, F. (1978) *Le traitement de l'expression du temps, un développement d'un système de compréhension du langage naturel.* Paris: Université de Paris VI (Thèse de 3e cycle).

Jones, S., Mason, P. and Stamper, R. (1979) LEGOL 2.0: a relational specification language for complex rules. *Information Systems 4*, 293–305.

Kahn, K. M. and Gorry, A. G. (1977) Mechanizing temporal knowledge. *Artificial Intelligence 7*, 87–108.

Koyama, T., Kaihara, S., Minamikawa, T. and Kurokawa, T. (1981) Time-oriented features for medical consultation systems. *Proceedings of the Seventh International Joint Conference on Artificial Intelligence—IJCAI/81 (Vancouver 1981).* Menlo Park (Calif.): The American Association for Artificial Intelligence.

Marburger, H., Neumann, B. and Novak, H. J. (1981) Natural language dialogue about moving objects in an automatically analyzed traffic scene. *Proceedings of the Seventh International Joint Conference on Artificial Intelligence—IJCAI/81 (Vancouver 1981).* Menlo Park (Calif.): The American Association for Artificial Intelligence.

Mattison, R. (1969) *An introduction to the model theory of first-order predicate logic and a related temporal logic.* Santa Monica (Calif.): Rand Corporation (Memo RM-5580-1-PR).

McDermott, D. (1982) A temporal logic for reasoning about processes and plans. *Cognitive*

Science 6, 101–155.

McShane, D. J., Harlow, A., Kraines, R. G. and Fries, J. F. (1979) TOD: a software system for the ARAMIS data bank. *Computer 12* (11), 34–40.

Mittal, S. and Chandrasekaran, B. (1980) Organizing data bases involving temporal information. *Proceedings of the International Conference on Cybernetics and Society (Cambridge, Mass., 1980)*. New York: Institute of Electrical and Electronics Engineers.

Moore, R. (1980) *Reasoning about knowledge and action.* Menlo Park (Calif.): SRI International (Technical Report 191).

Prior, A. (1967) *Past, Present, Future.* Oxford: University Press.

Rescher, N. and Urquhart, A. (1971) *Temporal Logic.* New York: Springer-Verlag.

Rosner, M. and Somers, H. L. (1980) *Case in linguistics and cognitive science.* Genève: Institut pour les études sémantiques et cognitives (Working Paper 40).

Sembugamoorthy, V. (1981) Analogy-based acquisition of utterances relating to temporal aspects. *Proceedings of the Seventh International Joint Conference on Artificial Intelligence—IJCAI/81 (Vancouver 1981)*. Menlo Park (Calif.): The American Association for Artificial Intelligence.

Steedman, M. J. (1977) Verbs, time and modality. *Cognitive Science 1*, 216–234.

Tsotsos, J. K. (1981) Temporal event recognition: an application to left ventricular performance. *Proceedings of the Seventh International Joint Conference on Artificial Intelligence—IJCAI/81 (Vancouver 1981)*. Menlo Park (Calif.): The American Association for Artificial Intelligence.

Vilain, M. B. (1982) A system for reasoning about time. *Proceedings of the National Conference on Artificial Intelligence—AAAI/82 (Pittsburgh 1982)*. Menlo Park (Calif.): The American Association for Artificial Intelligence.

Weiss, S. M., Kulikowski, A., Amarel, S. and Safir, A. (1978) A model-based method for computer-aided medical decision-making. *Artificial Intelligence 11*, 145–172.

Wilks, Y. (1975) *Seven theses on artificial intelligence and natural languages.* Castagnola (Lugano): Institut pour les études sémantiques et cognitives.

Wilks, Y. (1977) What sort of taxonomy of causation do we need for language understanding? *Cognitive Science 1*, 235–264.

Zarri, G. P. (1979) What can artificial intelligence offer to computational linguistics? The experience of the RESEDA project. *Advances in Computer-Aided Literary and Linguistic Research.* (D. E. Ager *et al.*, eds.) Birmingham: The University of Aston in Birmingham.

Zarri, G. P. (1981) Building the inference component of an historical information retrieval system. *Proceedings of the Seventh International Joint Conference on Artificial Intelligence—IJCAI/81 (Vancouver 1981)*. Menlo Park (Calif.): The American Association for Artificial Intelligence.

Zarri, G. P., Ornato, M., Lee, G., Lelouche, R., Meissonnier, V., De Vries, P., Hammond, L. and Pomian, J. (1981) *Projet RESEDA/1: Rapport sur les recherches effectuées du ler juillet 1980 au ler juillet 1981. Paris: Laboratoire d'Informatique pour les Sciences de l'Homme* (Rapp. LISH/224).

Zarri, G. P., Ornato, M., Lee, G., Lelouche, R., Meissonnier, V., De Vries, P. and Roccatti, M. (1980) *Projet RESEDA/1: Rapport sur les recherches effectuées du ler octobre 1979 au ler juillet 1980.* Paris: Laboratoire d'informatique pour les Sciences de l'Homme (Rapp. LISH/187).