Representation of Temporal Knowledge in Events : the formalism, and its potential for legal narratives

GIAN PIERO ZARRI

Centre National de la Recherche Scientifique (CNRS), 44, rue de l'Amiral Mouchez, 75014 Paris, France ; Tel:(33) (1) 43.13.56.84 ; Fax: (33) (1) 43.13.56.83; e-mail : zarri@ivry.cnrs.fr

ABSTRACT *Whereas temporal interval algebras appear to be the class of formalisms from Artificial Intelligence that is currently privileged by such researchers who are trying to handle temporal information as found in legal documents, there does not appear to be a consensus, within AI for Law, about a structured way of integrating the representation of the temporal data into the broader picture of treating the events or the overall plot. We describe here a knowledge representation formalism well-suited to take into account the temporal characteristics of narratives (of narrative documents). In these documents, the main part of the information content consists in the description of 'events' that relate the real or intended behaviour of some 'actors' (characters, personages, etc.). Narrative documents of an industrial and economic interest correspond, for example, to news stories, corporate documents (memos, policy statements, reports and minutes), normative and legal texts, intelligence messages, representation of the patient's medical records, etc. The formalism we present here is characterised by the following main properties: (i) it provides some general tools to deal with the 'fuzziness' which, in concrete situations, is inherently associated with the representation of any sort of 'timestamp'; (ii) it offers a way of implementing an efficient temporal reasoner, able to deal, for example, with the purely mechanical aspects of the well-known problem concerning the 'persistence of a situation'; (iii) it makes use of some second order representation tools (binding structures) to replace, to a certain extent, the interval algebra tools in the Allen style.*

1. Introduction

Temporal data are ubiquitous in any computer application where information about real life is captured. Such is the case of legal records. In laws themselves, a temporal pattern can be extracted from the given article of law. Argumentation, hypotheticals, natural-language texts, are also likely to involve some (even trivial) temporal relation, even though the situation envisaged may be admittedly removed from reality. Database records in legal or judiciary applications are likely to include some attribute about time. The temporal raw data that can be extracted are clearly not always emplotted into a broad story; much less so in the computer representations adopted. Yet, if you think of legal narratives about which a court is called to decide, it is easy to see how the set of temporal relations cannot be reasonably divorced from the commonsense texture of actions, events, and situations. For example, let us recall the distinction between the supposedly object-level narrative, what semiologist of law Bernard Jackson (1988, 1996), has termed the 'semantics' of the legal narrative, and what the presentations in courts make out of it, at a meta-level so to speak, or, in Jackson's terms, the 'pragmatics' of the legal narrative. It is essential for the discipline of AI & Law, therefore, that a handy representational tool be provided, which incorporates the treatment of the temporal dimension within a broader treatment of events, whether actual, or expected, or hypothetical.

This paper describes the main characteristics of the temporal knowledge representation system associated with NKRL, the 'Narrative Knowledge Representation Language' (see Zarri, 1994, 1997*a*, *b*, 2001; Zarri & Gilardoni, 1996). NKRL is a symbolic language used to describe the 'meaning' (the information content) of (legal) narratives, i.e. in concrete terms, the meaning of natural language (NL) 'narrative' documents. In these, the main part of the information content consists in the description of 'events' which relate the real or intended behaviour of some 'actors' (characters, personages, etc.): these try to attain a specific result, experience particular situations, manipulate some (concrete or abstract) materials, deliver or receive messages, set up some relationships with other personages, etc. These actors, as envisaged in NKRL for the purposes of capturing a representation of events in some domain of human activity, need not be human beings: we can have narrative documents concerning the vicissitudes in the journey of a nuclear submarine (the 'actor' or the 'personage') or the various avatars in the life of a commercial product. Besides normative and legal texts, narrative documents of an industrial and economic interest correspond, for example, to news stories, corporate documents (memos, policy statements, reports and minutes), intelligence messages, representation of the patient's medical records, etc.

NKRL has a long record of successful applications — particularly in the socio-economico-political field — which go back to the 1970s and to some ancestor systems like RESEDA (see, King *et al.*, 1977; Zarri, 1981). The most recent implementations of NKRL have been realised in the framework of two European projects: CONCERTO, Esprit P29159, and EUFORBIA, IAP P26505.

Among the reasons that are usually put forward to justify the introduction of the NKRL formalism, we can mention at least the following:

- Making use of a relatively simple, intuitive and easily manageable formalism, NKRL offers some interesting solutions to very hard problems concerning the 'practical' aspects of the knowledge representation endeavour. These include, the representation of implicit and explicit enunciative situations, of wishes, desires, and intentions (modals), of plural situations, of causality and of other second order, intertwined constructions — and, of course, of temporal information (see the NKRL papers mentioned in the 'References' for the technical details). Moreover, NKRL ascribes a great import to the possibility of discriminating exactly between the 'facts' or 'events' occurring in the real life and the 'notions' used to explain and understand these facts or events — a differentiation that is sometimes neglected in other knowledge representation systems.
- Another important characteristics of NKRL is linked with the fact that a sort of 'catalogue' of the basic conceptual structures ('templates', see Section 2) to be used to encode the concrete events is supplied as part and parcel of the definition of the language. This approach is particularly important for practical applications, and it implies, in particular, that: (i) a system-builder does not have to create himself the general structures needed to describe the events proper to a (sufficiently) large class of narrative texts and documents; (ii) it becomes easier to secure the reproduction and/or the sharing of previous results.

In Section 2, we will mention some general characteristics of NKRL, limiting ourselves to the notions strictly necessary to understand its temporal representation system. The fundamental properties of this last system will be detailed in Section 3 ; Section 4 will explain how the basic system, focused initially on the representation of the temporal characteristics of the single events, can be extended to the relationships between events and classes of events. In Section 5, code is developed for a fragment from a legal narrative by way of example.

2. Basic notions about NKRL

NKRL is a two layer language.

The lower layer consists of a set of general tools which are structured into four integrated subset (‘components’), the descriptive, factual, definitional and enumerative components.

The ‘events’ proper to a given domain are represented by using the descriptive and factual tools. Given the fuzziness that normally affects notions like those of event and of the related terms: state, situation, period, episode, history, process, action etc. — in NKRL, we have proscribed the use of these last locutions for clarity’s sake — we will clarify immediately what ‘event’ denotes in the context of our language. In NKRL, an event is the *temporally bounded* description of a particular set of interactions between one or more personages and other entities and objects that, like the personages, play a ‘role’ in the context of the description; to every event is then associated a proper, specific ‘duration’, which can also be ‘empty’ (point event). This means that, to each formal event represented in NKRL terms (‘occurrences’, see below), is linked a temporal interval on the time axis delimited by two ‘timestamps’ (Jensen *et al.*, 1994), t_1 and t_2 such that $t_1 \leq t_2$ — the meaning of ‘≤’, one of the seven relation of Vilain and Kautz’s Point Algebra (PA) (Vilain & Kautz, 1986), is here intuitive, see also subsection 3.2. For us, therefore, the formal representations of events like ‘At 6.30 pm on December 15th, John had a serious automobile accident’ (contracted episode), ‘John had several drinks during the day’ (continuous action, process), ‘At that moment, John was intoxicated’ (state), etc., are all occurrences associated with a duration that can go from zero to several hours.

Returning now to the components, the descriptive component concerns the tools used to produce the formal representations (called ‘predicative templates’) of *general classes* of narrative events, like ‘be present somewhere’, ‘to go or pass from one place to another’, ‘be affected by a given situation’, ‘having a negative attitude towards someone’, etc. — the reason for qualifying as ‘predicative’ the templates of the descriptive component will appear clearly in Section 4. These general classes — which are described independently from their possible instantiation into specific events — are obviously unconstrained from a temporal point of view; *no temporal interval is then associated with the predicative templates*. Templates are structured into an inheritance hierarchy, H_TEMP(lates), which corresponds, therefore, to a ‘taxonomy (ontology) of classes of events’.

The instances (‘predicative occurrences’) of the predicative templates, i.e. the NKRL representation of single, specific events like ‘Peter lives in Paris’, ‘John was driving this morning’, ‘Lucy is doubtful about the advantages of the new computer system’, ‘Mr. Smith has fired Mr. Brown’, are in the domain of the factual component. As already stated, any predicative occurrence c must be associated with the formal representation of the time interval i in which **holds**(c, i) is true — i.e. where the corresponding event is true. In the context of the factual component, the events taken into consideration must be ‘structured events’, i.e. they must be characterised by the explicit mention of an actor, an object, an instrument, etc. Correspondingly, and in opposition to the attribute-value data structures used, *inter alia*, for the NKRL concepts and individual, see below, the predicative occurrences (and the templates) are characterised by the basic following format :

$$(L_j (P_k (R_1\ a_1) (R_2\ a_2) \ldots (R_n\ a_n))).$$

In this expression, we denote with L_j the symbolic label identifying the template (class of events) or the occurrence (specific event), with P_k the predicate defining the general conceptual category of the class or in the event (a predicate can be defined as a labelled relation that exists among one or more arguments introduced by means of roles), with R_m the roles defining the function of the different arguments a_m with respect to the (single) predicate, see the examples in Figure 1 below. Presently, the predicates pertain to the set {BEHAVE, EXIST, EXPERIENCE, MOVE, OWN, PRODUCE, RECEIVE}, and the roles to the set {SUBJ(ect), OBJ(ect), SOURCE, DEST(ination), MODAL(ity), TOPIC, CONTEXT}. Note that, in accordance with our definition of event, the above predicates are to be equated to 'stative predicates', expressing a situation which does not change (holds) during the different subintervals i_r of *i*. Then MOVE means 'being in a state of motion during *i*', PRODUCE, 'being in a state of production', etc.

The definitional component of NKRL supplies the tools for representing the 'important notions' (concepts) of a given domain; in NKRL, a concept is, therefore, a definitional data structure associated with a symbolic label like *physical_entity*, *human_being*, *city_*, *car_* (the general class including all the automobiles, not a specific car), etc. These definitional data structures are, substantially, frame-like structures; moreover, all the NKRL concepts are inserted into a generalisation/specialisation (tangled) hierarchy that, for historical reasons, is called H_CLASS(es), and which corresponds well to the usual 'ontologies' of terms. To give rise to well-formed NKRL ontologies, the original notions must be conceived in terms of sets and subsets; moreover, no confusion is allowed between 'subsets' — giving rise to standard H_CLASS concepts, like *european_city* which is a specialisation of the concept *city_* — and 'instances', like paris_, see below.

The enumerative component of NKRL concerns then the formal representation of the instances (concrete examples, see john_, car_53, paris_, computer_15, not subsets) of some of the concepts (sortal concepts, see Zarri (1997*a*, *b*) of H_CLASS. In NKRL, their formal representations (once again, frame-like structures) take the name of 'individuals'. Individuals are characterised by the fact of being countable (enumerable), of being associated, like the predicative occurrences, with a temporal dimension (and, often, with a spatial dimension), and of possessing unique symbolic labels (john_, car_53, paris_, …). Throughout this paper, we will use the italic type style to represent a *concept_*, the roman style to represent an individual_.

The main element of the NKRL upper layer consists of the 'catalogue' mentioned in the 'Introduction'; in this, we can found a complete description of the formal characteristics and the modalities of use of the (nearly) 150 'basic templates' presently defined in the language — the catalogue is then a detailed definition of the elements of the H_TEMP hierarchy. By means of proper specialisation operations it is then possible to obtain from the basic templates the (specific) 'derived' templates that could be concretely needed to implement a particular, practical application, e.g., 'being intoxicated' from the basic template 'being in a given state', and the corresponding occurrences, e.g. 'at 6.30 p.m. on December 15th, John was intoxicated'.

Figure 1 shows a very simple example of NKRL code — see Section 5 for the representation of more complex situations. It translates the narrative fragment (a fragment of a Reuters' news): 'Brussels, July 2, 1993. British Airways Plc President Colin Marshall said in a Belgian newspaper interview the company's indebtedness was low …' — the full wording of this fragment will be analysed in Section 4 below.

In Figure 1, c1 and c2 are the symbolic names of two predicative occurrences, instances of basic NKRL templates. MOVE and EXPERIENCE are predicates; SUBJ(ect), OBJ(ect), DEST(ination), MODAL(ity) are roles. With respect now to the arguments, colin_marshall, british_airways, brussels_, newspaper_1, interview_1 are symbolic labels of individuals (enumerative component), i.e., they act as co-reference links allowing us to access, from the factual component environment, the corresponding structured objects (frames) of the enumerative component. These frames are not shown here for simplicity's sake, see however, Zarri (1997*a*, *b*). *chairman_*, *belgian_*, *indebtedness_* and *low_* are labels of concepts (definitional component, H_CLASS hierarchy); as a general rule, when the arguments represent general properties, or when no details are given about their concrete, peculiar characteristics, the arguments are represented as generic concepts instead of individuals. The 'attributive operator', SPECIF(ication), is one of the NKRL operators used to build up structured arguments, see Zarri and Gilardoni (1996). The SPECIF lists, with syntax (SPECIF e_1 p_1 ... p_n), are used to represent some of the properties which can be asserted about the first element e_1, concept or individual, of the list, e.g. *chairman_* and newspaper_1 in c1.

```
c1)   MOVE   SUBJ     (SPECIF colin_marshall (SPECIF chairman_ british_airways)): (brussels_)
             OBJ      #c2
             DEST     (SPECIF newspaper_1 belgian_)
             MODAL    interview_1
             date-1:  (2-july-93)
             date-2:

c2)   EXPERIENCE   SUBJ     british_airways
                   OBJ      (SPECIF indebtedness_ low_)
                   [ obs ]
                   date-1:  (2-july-93]
                   date-2:
```

Figure 1. A simple example of NKRL code.

A significant aspect of the descriptive / factual components concerns the fact that the single arguments, and the templates / occurrences as a whole, may be characterised by the addition of specific codes ('attributes') that give further details about their characteristics. For example, the 'location attributes' can be associated with particular arguments by using the colon code, ':', and are represented as lists — see occurrence c1. For the reasons given previously, the two temporal attributes date-1 and date-2 (see Figure 1) are necessarily associated with any well-formed predicative occurrence. They define, in fact, the time interval in which a predicative occurrence (the corresponding event) 'holds'. In Figure 1, this interval is reduced to a point on the time axis, as indicated by the single value, the timestamp 2-july-93, associated with the attribute date-1 (see subsection 3.2 below for the full details).

A particularly important category of attributes is represented by the 'modulators' (see Zarri, 1994, 1997*b*). Modulators are codes that are used to refine or modify the primary interpretation of a template or occurrence as given by the basic 'predicate — roles — argument' association: they work then as global operators which take as their argument the whole predicative template or occurrence. Modulators pertain to three categories: modality modulators, like ment(al), recip(rocity), soc(ial), for, against, etc.; deontic modulators, fac(ulty), interd(iction), oblig(ation), perm(ission); temporal modulators, like obs(erve) in Figure 1, attesting that the event c2 has been 'observed' at the date date-1 (see again subsection 3.2).

Finally, we will remark that the basic, MOVE template at the origin of c1 is systematically used to translate any sort of explicit or implicit 'transmission of an information' ('President Colin Marshall said …'). In this context, it is used according to a particular syntactic construction ('completive construction'), where the filler of the OBJ(ect) slot in the occurrence (here, c1) which instantiates the 'transmission' template is a symbolic label (here, c2) that refers to another occurrence, i.e. the occurrence bearing the informational content to be spread out ('… the company's indebtedness was low …').

3. The NKRL system of temporal representation

3.1. General remarks

In these last years, the temporal knowledge representation scene has been dominated by the debate about the 'Interval Algebra' (IA) proposed by James Allen, (1981, 1983, 1984), for applications such as automated planning and plan synthesis. An 'interval' is a finite length of time that starts and ends at definite points: it can be visually represented as a horizontal line with time going from left to right. According to Allen, a 'time specialist' dealing automatically with temporal information does not have to consider either absolute time or the duration of intervals, but merely the relations between intervals — i.e. it can leave unspecified the exact temporal relationship between intervals. Seven primitive relationships (predicates) between the temporal intervals i_1 and i_2 (and their inverse) are then defined in the interval algebra: i_1 **before** i_2, i_1 **equal** i_2, i_1 **meets** i_2 (i_1 is before i_2 but there is no interval between them, i.e. i_1 ends when i_2 starts), i_1 **overlaps** i_2 (i_1 starts before i_2, and they overlap), i_1 **during** i_2, i_1 **starts** i_2 (i_1 and i_2 shares the same beginning, but i_1 ends before i_2), i_1 **finishes** i_2 (i_1 and i_2 shares the same end, but i_1 begins before i_2). A set of transitive axioms, see Allen (1983), defines the behaviour of the above predicates; two examples are :

a) $\textbf{before}(i_1, i_2) \wedge \textbf{before}(i_2, i_3) \Rightarrow \textbf{before}(i_1, i_3)$;
b) $\textbf{meets}(i_1, i_2) \wedge \textbf{during}(i_2, i_3) \Rightarrow (\textbf{overlaps}(i_1, i_3) \vee \textbf{during}(i_1, i_3) \vee \textbf{meets}(i_1, i_3))$.

Thanks to the success of Allen's theory, much work has been directed towards bringing back Allen's work into the framework of first-order logic (see Allen & Hayes, 1985; Ladkin, 1987a), extending Allen's proposals (see Ladkin 1986; Ladkin & Maddux, 1987), investigating the relationships between the 'interval' and 'point' algebras (see Vilain & Kautz, 1986; Ladkin, 1987b; Tsang 1987, etc.). A temporal logic system introduced independently from the work of Allen, which presents some similarities with this last work while allowing the definition of dates and of the metric duration of events, is described in McDermott (1982). In this paper, McDermott introduces the concept of 'persistence of a situation' (see also subsection 3.3 below). Kowalski and Sergot's (1986) 'event calculus', is an attempt to set up a general system for reasoning about time and event in a logic programming framework; in a way, this system can also be considered as an extension of Allen's proposals. Finally, in the last few years, we can remark on a renewed interest for the point-based approaches. The majority of the most recent systems described in the literature can handle both metric (quantitative) and qualitative (interval-based) temporal information, see TMM (Time Map Management) (Dean & McDermott, 1987), TimeGraph, (Miller & Schubert, 1990), MATS (Metric/Allen Time System) (Kautz & Ladkin, 1991), Tachyon, (Arthur & Stillman, 1992), TimeGraph-II (Gerevini & Schubert, 1995), LATER (LAyered TEmporal Reasoner) (Brusoni *et al.*, 1997). TimeLogic, (Koomen, 1989) is, on the contrary, a pure interval-based system in the Allen's style.

Returning now to our problem, i.e. defining a temporal representation system for NKRL coherent with the premises expounded in the previous section, we can remark that a pure interval calculus does not seem well-suited for us for (at least) the following two reasons :

- Narrative texts are characterised by the presence of chains of events organised in a linear fashion (see Section 5) with timestamps that mark the beginning and the end of the different events. In these texts, we can find relatively few phenomena of superposition and of parallelism of the different chains. This means that disjunctions and conjunctions of Allen's predicates like those exemplified by the formulas (a) and (b) above are less necessary here than in other domains like planning and scheduling, i.e. those originally targeted by the interval algebra (IA) theory (see also Miller & Schubert, 1990). Assuming then a point algebra (PA) paradigm as the basis for the construction of our temporal representation system seems to be a reasonable choice. We can add that giving up the use of the implicit or explicit timestamps that can be found in a narrative documents constitutes indubitably an intolerable loss of information.

- From a more theoretical point of view, we can also remark that the computational efficiency of the interval algebra (IA) has often been questioned, e.g. the original constraint propagation algorithm at the core of the temporal reasoner associated with Allen's calculus (Allen, 1983) has been stigmatised as being inefficient in the case of large systems because of its $O(n^3)$ characterisation (see Vilain and Kautz, 1986). (The number of operations required in the case of one interval being added to the temporal network was a cubic function of the number of intervals already registered in the network). More in general, efficiency in both the IA and PA algebras depends on the extent to which 'disjunctive' combinations of points or predicates are allowed — see '**before**(i_1, i_3) $\vee$ **after**(i_1, i_3)' in IA, a disjunctive situation particularly important for planning and scheduling. If disjunctions are not permitted, IA and PA are equivalent, and tasks like that of deducing new relations from those that are already known can be executed in $O(n^2)$ time, where n is the number of intervals or points. If disjunction is permitted, problems like the previous one become intractable in IA (see, for example, Vilain & Kautz, 1986; Vilain *et al.*, 1989), while they can still be solved in polynomial time in PA. Researchers working in an IA framework have then tried to define tractable subsets of IA by translating disjunctive relations in IA into conjunctions in PA that involve the starting and ending points of the original intervals, see the 'continuous pointisable algebra' (van Beek & Cohen, 1990), and the 'ORD-Horn subalgebra' (Nebel & Bürckert, 1993) — the latter is more expressive but more expensive that the former. Only a limited number of IA disjunctions can be solved using this sort of techniques : e.g., '**before**(i_1, i_3) $\vee$ **meets**(i_1, i_3) $\vee$ **overlaps**(i_1, i_3)' may be tackled using a pointisable approach, but this is not possible for the disjunction '**before**(i_1, i_3) $\vee$ **after**(i_1, i_3)' introduced before (see also Gerevini & Schubert, 1994, 1995).

The approach chosen for NKRL is then, basically, point-oriented, even if it still allows us to perform at least some basic tasks in the IA style (see Section 4).

3.2. Categories and perspectives

In the context of NKRL, temporal phenomena concern mainly the occurrences of the factual component, see Section 2.

We recall here (i) that any predicative occurrence c must be necessarily associated with the formal representation of the time interval i in which **holds**(c, i) is true, and (ii) that i is defined by the timestamps associated as values to the two temporal attributes date-1 and date-2. Timestamps are composed of *sequences of integers* like <year, month, day, hour, minute, …>, where the left and right boundary represent, respectively, the maximum (e.g. years) and minimum (e.g. nanoseconds) 'temporal grain' chosen for a given application. In internal representation, each sequence is converted into a single real ; the equivalence 'timestamp ≡ real' preserve the order of the original timestamps on the time axis. From now onward, the term 'timestamp' will denote the real number corresponding to the original sequence. For clarity's sake, however, we will continue to use an intuitive, symbolic notation for timestamps, in the style of the value '2-july-93' in Figure 1 — which corresponds, therefore, to the sequence <year, month, day>. Pragmatic solutions have been adopted to deal with the problem of 'lacunary dates' given that, in, for example, historical applications, not all the elements of the original sequence corresponding to a timestamp can be known simultaneously. For example, assuming that 'maximum grain = years', sequences with the year unknown are not permitted. In this case, a set of possible years must be specified by the domain specialist: a copy of the associated occurrence is created for each possible year, and the set of occurences is added to the factual knowledge base.

Given the above assimilation of timestamps with reals corresponding to points of the time axis, it is now possible to make use of the standard arithmetical properties to establish the relationships between timestamps, and to calculate, when necessary, the duration of the intervals. With respect to the relationships, and expressing the generic timestamp as t_k we will have, for example, the following simple axioms (see also Miller & Schubert, 1990):

$$((t_1 \leq t_2) \wedge (t_2 \leq t_3)) \Rightarrow (t_1 \leq t_3)$$
$$(t_1 \leq t_2) \Leftrightarrow (t_1 < t_2) \vee (t_1 = t_2)$$
$$(t_1 \leq t_2) \Rightarrow (t_2 \geq t_1)$$
$$(t_1 < t_2) \Rightarrow \neg (t_2 \leq t_1)$$
$$(t_1 \geq t_2) \Rightarrow \neg (t_2 < t_1)$$
$$((t_1 \leq t_2) \wedge (t_2 \leq t_1)) \Rightarrow (t_1 = t_2), \text{ etc.}$$

To calculate the metric duration dt of an interval bounded by the timestamps t_1 and t_2, it is of course necessary to reconvert t_1 and t_2 into the original sequences in order to express these two timestamps in terms of 'significant multiples' of the minimum grain. 'Significant' means that, e.g. for two timestamps including the years 1996 and 1997 in the original sequence, and assuming that 'years' and 'days' are, respectively, the maximum and the minimum grain, only one year period will be considered when converting into days. Denoting with $g(t_k)$ the significant multiple of the minimum grain corresponding to the generic timestamp t_k, we will have simply

$$dt(t_1, t_2) = g(t_2) - g(t_1) \text{ and, if } t_1 \leq t_2 \leq t_3,\ dt(t_1, t_3) = dr(t_1, t_2) + dt(t_2, t_3).$$

If we examine now the qualitative relationships between the duration of an occurrence c_k (the duration of the corresponding event) and the temporal information carried by the temporal attributes date-1 and date-2, we will detect several situations.

In the simplest case, the duration is fully defined, as in the representation of the event : 'Between July 15 and September 5, 1997, John was hospitalised'; in the corresponding predicative occurrence, we do actually have two timestamps t_1 and t_2, $t_1 \equiv$ 15-july-1997 and $t_2 \equiv$ 5-september-1997 which allow us to localise exactly the boundaries of the event. From the point of view of the temporal information, this situation can thus be schematised as in Fig 2: the two timestamps t_1 and t_2 are associated, respectively, with the temporal attributes date-1 and date-2.

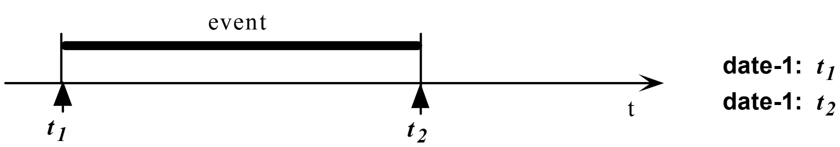


Figure 2. The duration of the event is fully defined.

We can now introduce a first fundamental concept of our system of temporal representation, that of 'category of dating'. The first temporal attribute, date-1, is said to be represented 'in subsequence' — the event begins to be true at the timestamp t_1 (generalised date) associated with this attribute — and the second date-2, which corresponds to the upper temporal limit of the event, is said to be represented 'in precedence'. The two attributes (their values), and the category of dating, subsequence or precedence, associated with the attributes, permit us to reconstruct the temporal interval (in Allen's meaning) corresponding to the event we are taking into consideration.

It is often necessary to deal with an event in which *only one* of the two boundaries, t_1 or t_2, is to be considered — for example, when it is necessary to supply special information about the circumstances at the *beginning* or *end* of the event. Another possibility is that only an *intermediate* timestamp t_3, between t_1 and t_2, is known. In all these cases, NKRL requires to make use only of the *first* temporal attribute, date-1, i.e. the single timestamp available is systematically associated as value with date-1, the second attribute, date-2, being 'empty'. The three cases are differentiated by using one of a particular class of modulators (see Section 2), the 'temporal modulators', to be linked with the global coded event. The 'beginning of an event' (timestamp t_1) will be represented by making use of the modulator 'begin', and the 'filled' temporal attribute date-1 is then represented 'in subsequence'. To indicate the 'end of an event' (timestamp t_2), the modulator 'end' is used, and the temporal attribute, date-1, must be represented 'in precedence'. If, finally, a particular moment within an event is to be indicated — for example, see the occurrence c2 of Figure 1 above, to express the information that *it appears* that British Airway's indebtedness was low on July 2, 1993, but without, at this level, giving any information about the beginning or end of this state of low indebtedness, *which extends beyond the given date* — the modulator 'obs(erve)" is used. In this last case, the non-empty temporal attribute, date-1, is now said to be represented 'in coincidence'. The three cases illustrated in this paragraph are summarised in Figure 3.

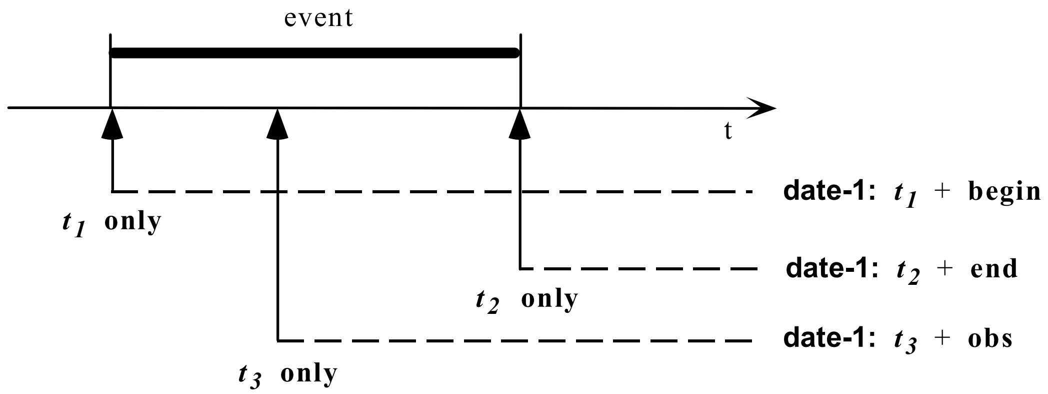
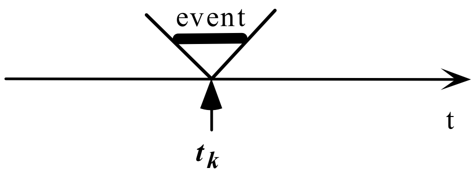


Figure 3. The duration of the event is unknown.

Eventually, a last case corresponds to the 'point events', i.e. events that have a metric duration $dt \leq$ the minimum temporal grain considered. An example is given by the event represented by the occurrence c1 in Figure 1: Colin Marshall's speech has probably occupied only a relatively small slice of July 2, 1993. This sort of event appears then as concentrated in a particular 'point' of the time axis, see Figure 4. The corresponding occurrences are characterised by the following format: (i) the timestamp t_k representing the date of the point event is associated as a value to the temporal attribute date-1; (ii) the temporal attribute date-2 is 'empty';

(iii) no temporal modulator is to be associated with the occurrence (see again Figure 1). The category of date-1 is the 'coincidence'.



date1: t_k

date2:

without temporal modulator

Figure 4. Point event.

Whatever the timestamp to be considered and the associated category, an important source of fuzziness is associated with the accuracy, or rather the lack of accuracy, with which this timestamp can be located on the time axis. The solutions proposed here ('perspectives') generalise some remarks that go back to Kahn and Gorry (1977; see also Miller & Schubert, 1990, p. 110).

In NKRL, 'perspectives' correspond to different ways of 'capturing' a timestamp. We have defined five different perspectives: 'direct perspective' (no fuzz, as in 'July 2, 1993, at noon'); 'inclusion fork' ('between April 7 and September 2, 1993'); 'limit from which' ('after January 1, 1993': a way of indirect dating, as in the case of a letter A which does not bear a date but mentions having received a letter B, which is dated 'January 1, 1993', this last date is thus a 'limit from which' for letter A); 'limit to which' (the symmetric case, 'before December 2, 1993': letter A does not bear a date, but is mentioned in a letter C, which is dated); 'circa perspective' (no lower or upper bounds, as in the case of a letter A which does not bear a date, but mentions the celebration of a feast day 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, 'about December 25, 1993'). We will stress here that the type of perspective which affects the temporal information associated as a value with a temporal attribute is *completely independent* of the category according to which the attribute itself is represented. In a piece of information of the type: 'John has been hospitalised in 1997, over a period whose first limit is probably between 10 and 30 June 1997, and the second between 1 and 15 September of the same year', the corresponding predicative occurrence would have both the temporal attributes date-1 and date-2 'filled', the first 'in subsequence' and the second 'in precedence' but, for both, the perspectives associated with the timestamps 'filling' these temporal attributes would be of the 'fork' type. We can remark that the 'category' concerns the temporal attributes; the 'perspective', the values (timestamps) to be associated with such attributes.

Thus, if we exclude direct dating, these perspectives all specify a range of possible values for the timestamp to be recorded. The only thing which distinguishes them from each other is the way in which this range is indicated in the original information sources (see Figure 5); *we can see that it can always be reduced to a 'fork'*, where *both* limits can be specified, *one* only ('from which' and 'to which'), or *none* of them ('circa').

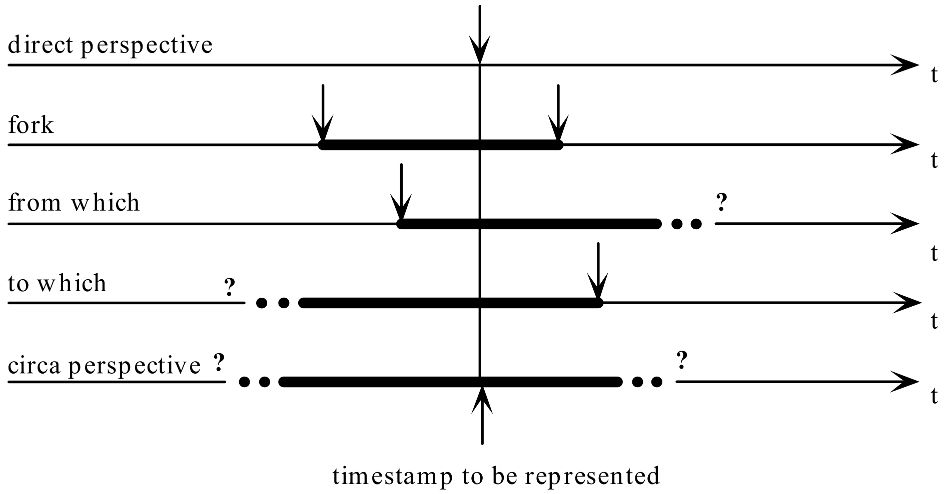


Figure 5. Timestamps and perspectives.

When both the limits are not provided, the missing limit(s) must be restored. Such a reconstruction could be executed (i) manually by the domain expert at the moment of building up the knowledge base of occurrences, making use of the context of the event to be coded and of its personal knowledge, or (ii) automatically by inference, at the processing time, using some algorithm designed to calculate 'so many days' or 'so many months' etc. before and/or after the effective timestamp (date) known and some sort of conceptual representation of the context, see, for example, the 'methods' by Kahn and Gorry (1977, pp. 95-98). What we want to emphasise here is that, with respect to the formal aspects of our temporal knowledge representation system, the values associated with the temporal attributes date-1 and date-2 are *always,* in reality, represented by *a vector of two elements* (two timestamps). Each time we have to deal with *non-direct perspectives* (fork, from which, to which, circa, see again Figure 5), the two elements of the vector are both explicitly expressed, giving the limits of a fork inside which is situated the (unknown) 'correct' timestamp to be associated with the attribute. On the other hand, the vector expressing a value to be represented in direct perspective only contains one explicitly expressed timestamp. The temporal attributes associated with the occurrence translating the previous example, 'John has been hospitalised in 1997 …' will then be coded as :

```
date-1:  (10-june-97 30-june-97)
date-2:  (1-september-97 15-september-97)
```

3.3. Temporal reasoning

We will sketch here the solution to the *purely mechanical aspects* of a simple 'persistence of a situation' problem, which consists in inferring that 'if fact P is true now, it will remain true until noticed otherwise'. Situations like this are then subsumed by the well-known (and particularly vicious) frame problem (see, for example, Pylyshyn 1987). We will see that our persistence problem can be handled by using solely some *conceptual indexes* based mainly on the concepts of category and perspective introduced in the previous subsection. More precisely, we will suppose of dealing with a knowledge base of *NKRL predicative occurrences* in the style of c1 and c2 of Figure 1; our indexing scheme combines three different kinds of NKRL elements:

- specific classes of *predicate's arguments* — i.e., instances (individuals) of specific concepts of the NKRL definitional component (see Section 2) which are particularly interesting in the domain under consideration: for example, the instances of *human_being* (the 'characters');

- the *semantic predicates* which are necessarily associated with each one of the predicative occurrences of the base;
- the *temporal information* which characterises each occurrence.

This information is organised in the following way:

- To every individual pertaining to the particular class of arguments selected (e.g., to each character that appears in the knowledge base) corresponds a *primary index*, by means of which it is possible to find all the predicative occurrences in which that individual (character) is mentioned.
- To every predicate corresponds a *conceptual record* within a primary index.
- For every individual and predicate (i.e. for every record of the primary indexes), the temporal *category* of the corresponding occurrences is taken into account by partitioning the timestamps of the occurrences into *three groups*, according to whether the corresponding temporal attributes are represented in precedence, in coincidence or in subsequence.
- The *perspective* of these timestamps is taken into account, inside each group, by separating the timestamps into *three subgroups*, according to whether they correspond to: a direct date (DD), the first limit of a fork (F1), or the second limit of a fork (F2).
- In fact, no timestamps are really inserted at the primary index level: a record of the primary index comprises *nine fields* (3 categories, i.e. three 'groups', by 3 perspectives, i.e. three 'subgroups'), each containing the address of a *secondary index*. A secondary index is a *list of 'pairs'*, where each 'pair' comprises: (i) a timestamp; (ii) the symbolic label (see 'c1', etc., in Figure 1) of the corresponding predicative occurrence. The secondary index lists are labelled as 'list DD', 'list F1', 'list F2'; inside each list, the pairs are ordered according to the increasing temporal order of the timestamps they contain.
- Each timestamp included within an occurrence will thus insert a reference to this occurrence (the label of the occurrence) either *once* in a list DD of the secondary index, or *twice* (once in list F1 and once in list F2), depending on whether the timestamp was represented by a direct date or by a fork (non-direct perspective).

We can summarise the different criteria used for gaining access to information in the knowledge base of predicative occurrences as shown in Table 1; Figure 6 shows the general structure of a primary index.

Table 1. Access to information in the knowledge base.

Access to	By selecting
The *individual* (e.g., a character)	The corresponding primary index.
The *predicate*	A conceptual record in this primary index.
The *category*	A group in this record.
The *perspective*	A subgroup (field, secondary index, list).
The *temporal values* (timestamps)	A pair in this list.
The *occurrence*	(Direct access by the second element of the pair).

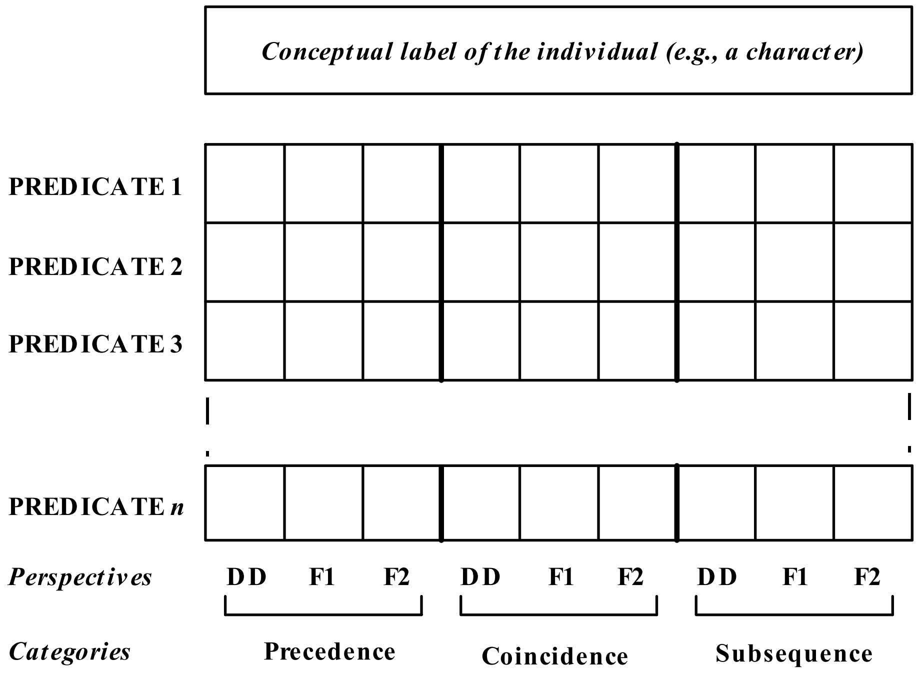


Figure 6. General structure of a primary index.

Returning now to the persistence problem, we will present, with the help of Figure 8, a plain description of the algorithm that solves this problem in its simplest form. This last corresponds to asking a formal NKRL query — a 'search pattern' (see, e.g. Zarri & Azzam, 1997) — where the two bounds of the 'search interval' are both explicitly expressed, as in the example of the question: 'Was John at the hospital in July/August 1997?', to be unified, e.g. with the information in the style of 'On June 12, 1997, John was admitted to hospital', see Figure 7. A search pattern (see the second part of Figure 7), represents (in NKRL terms) the general properties of information to be searched for, by filtering or unification, within a knowledge base of occurrences — a search pattern can be considered an NKRL equivalent of a natural-language query. The two timestamps of the pattern constitute the 'search interval' linked with this pattern, to be used to limit the search for unification to the slice of time that it is considered appropriate to explore. A formal definition of the algorithm is given in the Appendix A ; the extension to more complex query formats is straightforward.

```
c3)    EXIST    SUBJ       john_: hospital_1
                [ begin ]
                date-1:    (2-june-97)
                date-2:

(?w    IS-PRED-OCCURRENCE
       :predicate      EXIST
       :SUBJ           john_
       (1-july-1997, 31-august-1997))
```

Figure 7. A simple example of search pattern.

In Figure 8, each list is the symbolic representation of a *secondary index*, that is, of a sequence of pairs <timestamp, label of occurrence>. The nine lists represented in this figure correspond to a *record* of the primary index characterised by a given semantic predicate (see also Table 1): the list pDD is therefore the list 'precedence - direct perspective', the list pF1 'precedence - first limit of a fork', the list cDD 'coincidence - direct perspective', etc. The nine lists are therefore divided by the two limits, 'bound1' (e.g., 1-july-1997 in the pattern of Figure 7) and 'bound2' (31-august-1997) specified in the search pattern into three 'periods' (before bound1; between bound1 and bound2; after bound2) as shown in Figure 8. If we imagine, for the time being, that the database of occurrences only contains occurrences in *direct perspective* (the lists pF1, pF2, cF1, cF2, sF1 and sF2 are *empty*), portions of the 'type DD' lists are to be selected according to the following criteria:

- The part of the list cDD corresponding to *period 2* is *definitely* to be selected. The occurrences characterised by a temporal attribute date-1 represented 'in coincidence' (point events, or occurrences associated with the modulator 'obs') that have a (unique) timestamp included within this period take place in fact, at least partly, inside the search interval bound1 - bound2.
- The part of the list pDD corresponding to *period 2* is also to be selected: the corresponding occurrences *definitely end* (precedence = *until to* …) after bound1 and before bound2, and are therefore located, at least partly, inside the search interval. The part of the list sDD corresponding to the same period is also to be considered, since the corresponding events *definitely begin* (subsequence = *starting from* …) inside the search interval. This last case corresponds to the situation of the occurrence c3 in Figure 7 above, which satisfies then the question: in Figure 7, date-1 is represented 'in subsequence', and the timestamp 2-june-97 is a direct date.
- The part of the list pDD corresponding to *period 1* is to be excluded, since it refers to occurrences which *definitely end* before bound1; similarly, occurrences in the list sDD corresponding to *period 3* are to be excluded, since they *definitely begin* after bound2.

Occurrences in the list pDD in *period 3* (occurrences which *end* after bound2) are *potentially* to be selected. However, before accepting one of these events, one must check that the same occurrence (the same label) *does not also appear* in the pairs of the list sDD in *period 3* (this event would then *begin* and *end after* bound2). A situation of this type could be that of an event like: 'John has been hospitalised between September and December 1997' with respect to the same search interval of Figure 7, '1-july-1997 (bound 1) - 31-august-1997 (bound2)'. The corresponding occurrence (to be discarded) would in effect be classed twice in period 3, both 'in subsequence', date-1: (1-september-1997), as well as 'in precedence', date-2: (31-december-1997). The part of the list sDD in *period 3* therefore acts as an '*exclusion filter*' for the corresponding part of the list pDD (the occurrences which are classed in both these two parts must not be selected); the same for the part of the list pDD in *period 1* with respect to the corresponding part of sDD. We are then able to deal with the 'persistence' phenomena associated with the presupposition that 'if fact P is true now, it will remain true until noticed otherwise'.

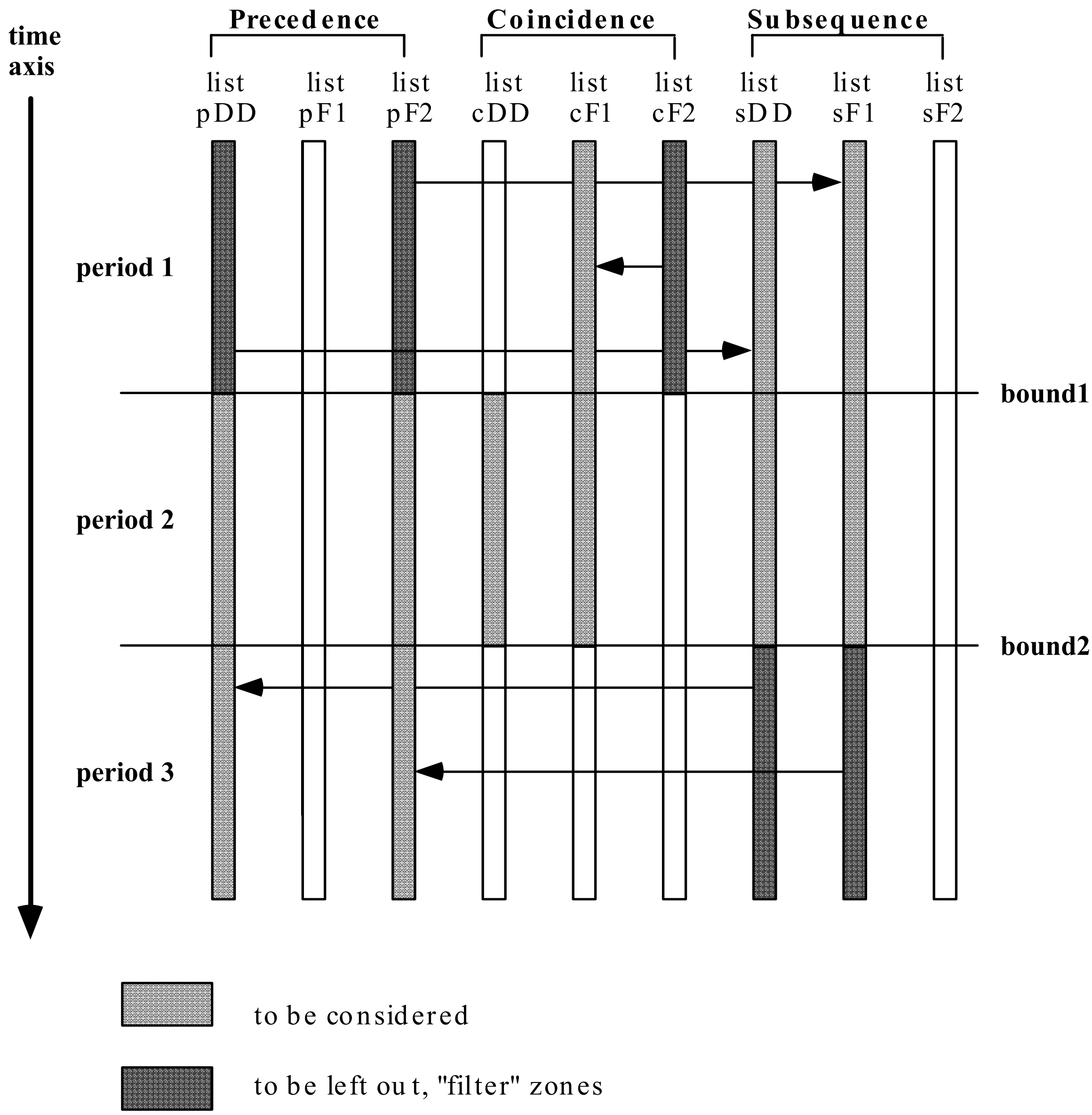


Figure 8. Graphical representation of the selection algorithm.

To extend this algorithm to the general case (the lists relative to the non-direct perspectives are *non-empty*), the following considerations must be added:

- The same group of *occurrence labels* appears in the pairs associated with the lists pF1 and pF2 (the same for the lists cF1 - cF2 and sF1 - sF2). If an occurrence is characterised in fact by a fork 'filling' its *second* temporal attribute, date-2 (classed *in precedence*), its label will appear in a pair of the list pF1, linked with the timestamp representing the first bound of the fork, as well as in the list pF2, linked with the timestamp representing the second bound of the fork.
- We can then avoid considering the lists pF1 and sF2: the '*deciding conditions*' for *precedence* in fact will be tested on the one list pF2, in which the timestamps are 'lower

down in the list' in comparison with those contained in the corresponding pairs in the list pF1 ; the deciding conditions for posteriority will be tested on the list sF1, in which the dates are 'higher up'. If we consider an occurrence carrying the following temporal information (forks): date-1: (10-june-97, 30-june-97); date-2: (1-september-97, 15-september-97), the timestamps which will finally be examined for the selection will be 10-june-97 (subsequence) and 15-september-97 (precedence).

- If we take the same argument for the lists pF2 and sF1 as we did for the corresponding DD lists, we see that the portions to be *selected* are those of the list pF2 which belong to *periods 2 and 3*, and those of the list sF1 which belong to *periods 1 and 2*. The part of the list pF2 in *period 1* and that of the list sF1 in *period 3* are definitely to be *excluded*; they act, however, as exclusion filters with regard to the part of the list sF1 in period 1 and to that of the list pF2 in period 3, respectively.
- As for coincidence, it can easily be shown that the portions 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 respect to the part of the list cF1 in the same period.

What we have just explained is summarised in Figure 8. The 'ligther' areas represent the list parts to be considered; the 'sharper' ones represent parts which are definitely to be excluded, they also represent exclusion filters with regard to the arrowed parts.

We can observe that the naturalness and the expressive power of the procedure just expounded — based only on the two concepts of dating category and perspective — compare favourably with a pure interval algebra and other temporal reasoning techniques such as, for example, those described in Dean & McDermott (1987).

4. Higher-order temporal relationships

In the previous Sections, occurrences and templates have been considered 'in isolation'. But, obviously, being able to represent single events or single classes of events is not enough: it is also necessary to represent the logico-semantic links which can exist between disjoint events or classes of events (the 'co-ordination' and 'subordination' links, using a metaphor from the domain of natural language). The general way to solve this problem in NKRL — see Zarri (1994, 1997*a*, *b*) — is to use some sort of 'binding' (as opposed to 'predicative') structures. The binding templates and occurrences are second-order structures represented under the form of lists where the elements are symbolic names (conceptual labels) of *predicative* templates and occurrences — they are then characterised by the absence of any predicate or role. The lists are labelled by using one of the 'binding operators' listed in Table 2, i.e. the binding operator represents always the first element of the list. The syntax of the binding structures is in general: (BIND-OPERATOR l_1, l_2 ... l_n) — see, however, the remarks in Table 2 about ASSOC and the four causal operators. The l_k are labels of predicative and binding templates or occurrences.

We can now supply the full representation of the example of Figure 1 above: 'Brussels, July 2, 1993. British Airways Plc President Colin Marshall said in a Belgian newspaper interview the company's indebtedness was low **following a capital increase**' (see Figure 9).

In this figure, the content of the message is represented by the binding occurrence c5. This last occurrence — meaning that event c2, the main event, has been caused by event c4 — is labelled using one (CAUSE) of the four operators which define together the taxonomy of causality of NKRL (see Table 2 and Zarri, 1992*a*, p. 705). The new occurrence c4 translates an event that has been interpreted as a 'point event' (no temporal modulator) (see subsection 3.2). The category of the temporal attribute date-1 is the coincidence; the (non-direct) perspective of the value associated with date-1 is 'limit to which' (before July 2), as indicated by the fact the *first* limit of the fork is a 'reconstructed date', included then into round brackets.

Table 2. Binding operators.

Operator	Acronym	Mnemonic Description
Alternative	ALTERN	The disjunctive operator. Only a single element of the associated list must be considered ; however, this element is not known *a priori*.
Co-ordination	COORD	The collective operator: all the elements of the list must, obligatorily, be considered together to give rise to a valid binding relationship.
Enumeration	ENUM	The distributive operator: each element of the list must be considered in order to produce a valid binding relationship, but they satisfy this relationship separately.
Association	ASSOC	The binding lists built up using ASSOC and the following causal operators contain only three elements, the operator and two arguments (labels of predicative or binding templates and occurrences). The first argument of ASSOC relates the origin of a given 'temporal development', i.e., of a complex episode involving several events; the second argument gives the development itself. ASSOC is both: (i) a sort of temporal COORD; (ii) a generic operator which subsumes the four causal operators below.
Cause	CAUSE	The 'strict causality' operator, introducing a necessary and sufficient causal relationship between the first and the second arguments of the list, the latter explaining the former.
Reference	REFER	The 'weak causality' operator, introducing a necessary but not sufficient causal relationship between the first and the second arguments'
Goal	GOAL	The 'strict intentionality' operator: the first argument is necessary to realise the second, and the second is sufficient to explain the first.
Motivation	MOTIV	The 'weak intentionality' operator : the first argument is not necessary to realise the second, but the second is sufficient to explain the first.

Given that all the predicative occurrences mentioned in a binding occurrence are necessarily associated with explicit temporal indications, it is not difficult to assess their temporal arrangement within the binding occurrence. A problem could, on the contrary, arise for the binding template, given that the predicative template are, by definition, atemporal (see Section 2).

The problem is not too troublesome, given that the eight binding operators of Table 2 are all temporally characterised: the first three (ALTERN, COORD, ENUM) assert the 'contemporaneity' of the associated events, the fourth (ASSOC), because of its definition, ensures that its second argument is 'posterior to' the first one, the last four, the causal operators, have the associated temporal valence that is graphically illustrated in Figure 10. As we can see from this figure, CAUSE and GOAL express, respectively, a sort of *strict* causality or purpose; REFER and MOTIV a *weak* causality and purpose. Expressed in different terms, we can say that the use of CAUSE and GOAL is only permitted in the presence of a 'material implication' $C_k \supset C_l$; it is evident that this last condition is not respected in the two examples of Figure 10 which illustrate the use of CONFER and MOTIV.

```
c1)   MOVE       SUBJ     (SPECIF colin_marshall (SPECIF chairman_ british_airways)): (brussels_)
                 OBJ      #c5
                 DEST     (SPECIF newspaper_1 belgian_)
                 MODAL    interview_1
                 date-1:  (2-july-93)
                 date-2:

c2)   EXPERIENCE      SUBJ      british_airways
                      OBJ       (SPECIF indebtedness_1 low_)
                      [ obs ]
                      date-1:   (2-july-93)
                      date-2:

c4)   PRODUCE    SUBJ   british_airways
                 OBJ    (SPECIF capital_increase_1 british_airways)
                        date-1:  ((2-july-92) 2-july-93)
                        date-2:

c5)   (CAUSE  c2  c4)
```

Figure 9. The example of Figure 1 revisited.

As a simple example of temporal interpretation of the binding structures, we reproduce in Figure 11 the predicative and binding templates included in the NKRL description (corresponding to the individual art._57) of a fragment of normative text, the beginning of article no. 57 of the French General Taxation Law (see Zarri, 1995, for some details about the techniques used to automatically produce the NKRL representation of complex normative texts). Article no. 57 is the main source used to settle cases concerning a possible 'indirect transfer of revenues abroad'. The beginning of this article reads as follows, according to a rough English translation : 'To determine the income tax payable by companies which are under the authority of, or which exercise a control over, companies domiciled abroad …'.

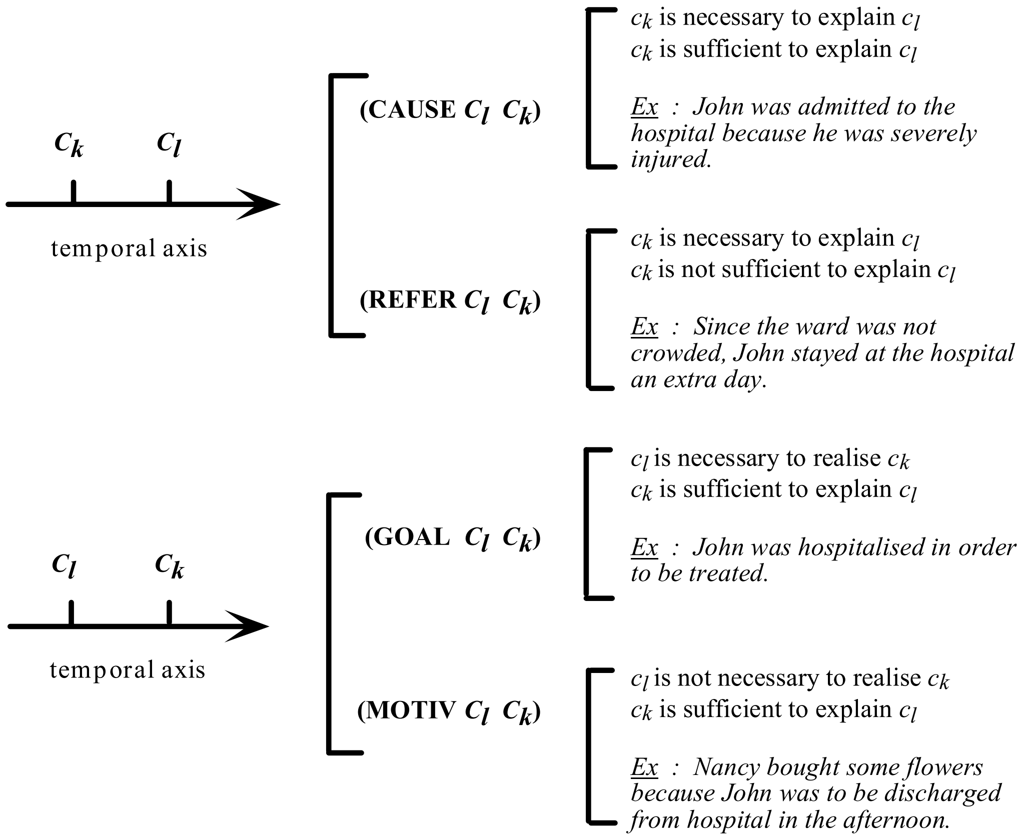


Figure 10. Time and causality in NKRL.

t1) PRODUCE SUBJ x_1
OBJ (SPECIF *calculation_ income_tax*)
DEST x_2 : france_

x_1 = *human_being _or_social_body* ; x_2 = *company_*

t2) OWN SUBJ x_2 : france_
OBJ (SPECIF *control_power* x_3)

x_3 = *company_* ; $x_2 \neq x_3$

t3) EXIST SUBJ x_3 : *foreign_country*

t4) OWN SUBJ x_3 : *foreign_country*
OBJ (SPECIF *authority_* x_2)

t5) EXIST SUBJ x_2: france_

t6) (ALTERN (COORD t1 t2 t3) (COORD t1 t4 t5))

t7) (GOAL … t6)

Figure 11. Binding and predicative templates.

Objects labelled t1 - t5 are predicative templates; we can see that, at the difference of the predicative occurrences, they are characterised by the presence of variables, and of constraints on the variables. t6 and t7 are binding templates. The binding template t6 represents the crucial alternative that is expressed by the fragment of text examined: the calculation of the income tax (t1) may concern a French company x_2 that controls (t2) a generic foreign company x_3 or is controlled (t4) by such a company. From a temporal representation point of view, the alternative (and the coordinations) of t6 concerns general models of events that must be taken into consideration simultaneously. Template t7 expresses the fact that the normative precept enunciated later in the wording of Art. 57 ('… the revenues indirectly transferred abroad must be added to the results registered in the books'), represented in NKRL terms by a new binding template, is not only *strictly necessary* (GOAL) to allow the realisation of the task (determination of the income tax …) described in t6, but it also *precedes*, obviously, the execution of this task. A complete analysis of Art. 57 can be found in Zarri (1992*b*). Even if the temporal relationships that can be asserted between NKRL elements making use of the binding operators of Table 2 do not cover, of course, all the possibility of Allen's IA — and they would be, then, insufficient in a planning and scheduling environment — they are largely adequate in a narrative context.

It may also be worth noting that the templates t1 - t5 of Figure 11 are (simplified representations of) 'derived' templates, i.e. specialisations, according to the characteristics of the particular NL text at hand, of 'basic' templates pertaining to the H_TEMP hierarchy of the NKRL descriptive component (see Section 2). For example, the template t1 is a specialisation of a basic template describing all types of events concerning the 'production' of numerical values (including estimations, modifications of pre-existing values, etc.): this basic template contains the generic NKRL concepts (H_CLASS taxonomy) *setting_of_numerical_value* and *numerical_value* instead of, respectively, their specific terms (subtypes) *calculation_* and *income_tax* that appear in t1 — note that *income_tax* pertains also to the H_CLASS subtree *law_concepts* (multiple inheritance). Template t2 is a specialisation of the basic template which translates the idea of 'being endowed with a sort of authority' : the general *type_of_power* concept appearing as an OBJ(ect) argument in this basic template has been replaced in t2 with a more specific *control_power* concept pertaining to the same subtree of H_CLASS, and a SPECIF(ication) list has been added; etc.

5. An example of coding of a legal narrative

Having tutored the reader into the formalism we are going to apply, let us consider an example of a legal narrative. One option could have been to select a notecase from a textbook, or a specialised publication, or even the legal column of some newspaper or other general publication. However, then extracting just a skeleton consisting of the temporal structure could well yield a temporally rather trivial situation. Through interaction with an expert of the field, Ephraim Nissan, we settled on an 'action story', a legal narrative seen through the lens of Tony Lord (1996) in a South East London borough local paper's column devoted to the local chronicle of old. Moreover, the way we selected our source — from a newspaper— enabled us to avoid involving ourselves with the representation of subtle concepts from Law. Consider then the following summarised account of the retelling, in Lord's paper, of a robbery attempt committed at a cinema shortly after World War I by two men, John and George.

Nellie, the cashier of a cinema, locked the door of the paybox, that December evening in 1919, and carried the heavy bag of coins upstairs, into manager Mr. Edward's office, where she and the manager would check the day's takings. Right after the money was put in the safe, the door of the office was pushed open and two men burst in. One of them requested the money; should his victims move, he would shoot, he threatened, pointing a German automatic pistol to the manager's head. At this point the two intruders had moved into the room, away from the open door. Nellie slipped past the second man who made a grab at her but just failed to hold her and she ran screaming down into the street. The manager was successful at defending himself, to escape the office while trapping inside, which he kept doing with difficulty, the armed man (the only robber left in the room), until Nellie arrived with a policeman, Sergeant Buchanan, who then overcame and arrested the armed man. The other man, who turned out to be his brother, was arrested a few days later. The pistol was also found. The background of the armed man was not in his favour. In court, John, the armed man, admitted his guilt, whereas his brother George pleaded not guilty, a claim to which the jury was to be unreceptive. The narrative fragment to be coded is now the following :

> At the Old Bailey, the following February, John admitted his guilt but George pleaded not guilty, telling the court that the two of them had taken shelter from the rain in the doorway of the cinema. John had heard the chinking of money being counted and had taken out the gun which he had bought from a Russian in a pub and run upstairs. George said he followed and tried to pull his brother away. The jury did not think this was a very likely story and found both of them guilty'.

Some general remarks apply here :

- For exhaustiveness' sake, and to emphasise the flexibility of the NKRL language, the 'translation' presented below is a 'literal' one. This means that we have rendered here into NKRL even unessential details, like '… (the gun) he had bought from a Russian in a pub …', that, normally, should be disregarded in an 'industrial' exploitation of a narrative.
- All the occurrences of the translation are instances of standard 'basic templates' pertaining to the NKRL 'catalogue' (see Section 2). No new template has been created specially for the purpose in hand.
- The concepts (and their instances) used in the translation are, in their great majority, standard H_CLASS concepts, i.e. concepts that are common to many possible applications. A few *ad hoc* concepts, like *cinema_doorway*, *sheltering_* or *chinking_* have been created, however, to keep the translation as much as simple and comprehensible as possible. Note that, as already remarked in Section 2, when the arguments of the semantic predicates represent general properties, or when there is no precise need for creating specific individuals to be used again, e.g. in other occurrences (co-reference), the arguments are represented as generic concepts instead of individuals.

5.1. NKRL coding

c6) (COORD c7 c8 c9)

The coding of the fragment includes three main elements : the confession of John (c7), the declaration of innocence of George (c8), that is accompanied by a long development (c13), the reaction of the jury (c9).

c7)	MOVE	SUBJ	john_: (old_bailey)
		OBJ	#c10
		MODAL	*acknowledgment_*
		date-1:	(1-february-1920 29-february-1920)
		date-2:	

Assuming, for simplicity's sake, that the minimum temporal grain considered for the coding of the fragment is the day, the event to be represented (John's confession; see also the associated occurrence c10) has a metric duration ≤ of this grain; c7 and c10 are then conceived as point events whose category of dating is 'coincidence', see subsection 3.2. The perspective is 'fork', i.e. the confession took up a fraction of an (unknown) day in February 1920 (the year 1920 was, of course, a leap year).

c10)	EXPERIENCE	SUBJ	john_
		OBJ	*guilty_*
		date-1:	(1-february-1920 29-february-1920)
		date-2:	

c8)	MOVE	SUBJ	george_: (old_bailey)
		OBJ	#c11
		MODAL	*plea_*
		date-1:	(1-february-1920 29-february-1920)
		date-2:	

c11) (ASSOC c12 c13)

George's speech is formed of two parts: he says he is innocent with respect to the robbery (c12); he then describes a series of events (c13) that should confirm his thesis. See Table 2 for the meaning of ASSOC.

c12)	EXPERIENCE	SUBJ	george_
		OBJ	*innocence_*
		date-1:	(1-february-1920 29-february-1920)
		date-2:	

c13) (COORD c14 c15 c16)

George's explanation is formed of three parts: the two brothers had taken shelter from the rain in the doorway of the cinema (c14); having heard the chinking of money being counted, John had run upstairs with the gun (c15); George had tried to pull his brother away (c16).

c14) (GOAL c17 c18)

c17)	MOVE	SUBJ	(COORD1 john_ george_):
		OBJ	(COORD1 john_ george_): (cinema_doorway_1)
		date-1:	(26-december-1919 31-december-1919)
		date-2:	

This occurrence illustrates one of the most characteristic constructions of NKRL: the displacement of a human being from a location to a different one is always coded under the form of this human being, as the SUBJ(ect), that MOVE(s) himself as an OBJ(ect). The initial location is linked with the SUBJ, the final location with the OBJ. The event to be coded here is considered, like c7 etc., as a point event. The fork of dates associated with date-1 is deduced from the information given at the beginning of the narrative: 'Miss Nellie … locked the door of the paybox … that December evening in 1919 … Business had been good with people glad to come out for some entertainment after being stuck in the house … through the [X]mas holidays'.

c18)	[*] EXPERIENCE	SUBJ	(COORD1 john_ george_)
		OBJ	(SPECIF *sheltering_ rain_*)
		date-1:	(26-december-1919 31-december-1919)
		date-2:	

The event represented by the second argument of a GOAL (or MOTIV, see Table 2) construction is always interpreted as 'conjectural', i.e. as the intended, but not necessarily fulfilled, result of the event represented by the first argument, here c17. As a consequence, the second argument of a GOAL/MOTIV construction is characterised by the presence of a 'conjectural validity attribute', code '*', see also occurrence c27 below.

c15) (CAUSE c19 c20)

The explanation of John's behaviour (c19) is that he had heard the chinking of money (c20).

c19) (COORD c21 c22)

John's behaviour consists in taking out the gun (c21) and running upstairs (c22).

c21) (REFER c23 c24)

The purchase of the gun (c24) is considered as a necessary but not sufficient condition for taking out this gun (see Table 2).

c23)	PRODUCE	SUBJ	john_
		OBJ	(SPECIF *take_out* gun_1)
		date-1:	(26-december-1919 31-december-1919)
		date-2:	
c24)	PRODUCE	SUBJ	john_: (pub_1)
		OBJ	(SPECIF *purchase_* gun_1)
		SOURCE	russian_1
		date-1:	((1-november-1919) 31-december-1919)
		date-2:	

The perspective of the date associated with date-1 is 'to which', see the 'reconstructed date' (in parentheses, subsection 3.2) that constitutes the first term of the fork: the purchase of the gun happened *before* the events that occurred in that evening of December 1919. The coding of the fork is here especially intricate, given the vagueness of the temporal information associated with this particular narrative. We have taken as second term of the fork the upper limit that can be deduced from the indications concerning Christmas holidays (see the comment on occurrence c17 above); with respect to the first term (the reconstructed date), we have assumed, very arbitrarily indeed, that the purchase was a relatively recent one.

c22)	MOVE	SUBJ	john_: (cinema_doorway_1)
		OBJ	john_: upstairs_1
		MODAL	*running_*
		date-1:	(26-december-1919 31-december-1919)
		date-2:	
c20)	RECEIVE	SUBJ	john_: (cinema_doorway_1)
		OBJ	#c25
		MODAL	*hearing_*
		date-1:	(26-december-1919 31-december-1919)
		date-2:	

The basic template that gives rise to occurrence c20 is used to code the *acquisition* of an information and is, therefore, symmetrical with respect to the basic template that, making use of the predicate MOVE, is employed to code the *transmission* of an information, see, for example, the occurrences c7 and c8 above.

c25)	PRODUCE	SUBJ	human_being_1: (upstairs_1)
		OBJ	(SPECIF *counting_* money_1)
		MODAL	*chinking_*
		date-1:	(26-december-1919 31-december-1919)
		date-2:	
c16)	(GOAL c26 c27)		
c26)	MOVE	SUBJ	george_: (cinema_doorway_1)
		OBJ	george_: (upstairs_1)
		MODAL	(SPECIF *pursuit_* john_)
		date-1:	(26-december-1919 31-december-1919)
		date-2:	
c27)	[*] PRODUCE	SUBJ	george_
		OBJ	(SPECIF *hindrance_* john_)
		date-1:	(26-december-1919 31-december-1919)
		date-2:	
c9)	(CAUSE c28 c29)		
c28)	BEHAVE	SUBJ	jury_1: (old_bailey)
		OBJ	(COORD1 john_ george_)
		MODAL	(SPECIF declaration_1 *guiltiness_*)
		[against]	
		date-1:	(1-february-1920 29-february-1920)
		date-2:	

The template at the origin of this occurrence pertains to the set of NKRL constructions used to represent the 'attitude toward someone/something' domain. The OBJ(ect) role contains the 'target' of the attitude; the 'value', positive or negative, is given by the modulator, 'against' in this case.

```
c29)   EXPERIENCE      SUBJ      jury_1
                       OBJ       skepticism_
                       TOPIC     #c13
                       date-1:   (1-february-1920 29-february-1920)
                       date-2:
```

TOPIC = concerning, apropos of …

6. Conclusion

In this paper, we have described a temporal knowledge representation formalism that is part of a knowledge representation language for narrative documents, NKRL. This formalism, essentially a 'point'-based one, is characterised by the main following properties: (i) it provides some general and coherent tools in order to deal with the 'fuzziness' which, in concrete situations, is inherently associated with the representation of any sort of timestamp; (ii) it provides a method of implementing an efficient temporal reasoner, able to deal, for example, with the purely mechanical aspects of the well-known problem concerning the 'persistence of a situation'; (iii) it can make use of some second-order representation tool (binding structures) to replace, to a certain extent, the explicit interval algebra tools in the Allen's style. Original concepts of this type of representation are, *inter alia*, the notions of category and perspective. The 'category of dating' characterises the association of a generic temporal attribute with the beginning (subsequence), the end (precedence) or a particular moment (coincidence) of a given event. The 'perspective of dating' is used to define the degree of precision (for example, 'around' a particular date) with which we can locate on the time axis the timestamp associated as a value to a temporal attribute. These two concepts are at the base of the 'conceptual indexing tools' used to implement the NKRL temporal reasoners. We have then developed in some details the code corresponding to the NKRL representation of a fragment from a legal narrative. Note that fine-tuning the instance of representation to carefully account for the legal dimension in a way that is acceptable in a judiciary context was beyond the scope of this paper, but can be incorporated incrementally by the interested reader who know now how to proceed if she or he is to adopt our temporal formalism.

References

Allen, J.F. (1981) "An Interval-Based Representation of Temporal Knowledge", in *Proceedings of the 7th International Joint Conference on Artificial Intelligence - IJCAI/81*. San Francisco: Morgan Kaufmann.

Allen, J.F. (1983) "Maintaining Knowledge About Temporal Intervals", *Communication of the ACM* **26**, 823-843.

Allen, J.F. (1984) "Towards a General Theory of Action and Time", *Artificial Intelligence* **23**, 123-154.

Allen, J.F., and Hayes, P.J., "A Common-Sense Theory of Time", in *Proceedings of the 9th International Joint Conference on Artificial Intelligence - IJCAI/85*. San Francisco: Morgan Kaufman.

Arthur, R., and Stillman, J. (1992) *Temporal Reasoning for Planning and Scheduling* (Technical Report). Schenectady (NJ): AI Lab. of the General Electric Research Center.

Brusoni, V., Console, L., Terenziani, P., and Pernici, B. (1997) "LATER : Managing Temporal Information Efficiently", *Intelligent Systems and Their Applications* **12**(4), 56-63.

Dean, T.L., and McDermott, D.V. (1987) “Temporal Data Base Management”, *Artificial Intelligence* **32**, 1-55.
Gerevini, A., and Schubert, L. (1994) “On Point-Based Temporal Disjointness - Research Note”, *Artificial Intelligence* **70**, 347-361.
Gerevini, A., and Schubert, L. (1995) “Efficient Algorithms for Qualitative Reasoning about Time”, *Artificial Intelligence* **74**, 207-248.
Jackson, B.S. (1988) “Narrative Models in Legal Proof”, *International Journal for the Semiotics of Law* **1**, 225-246.
Jackson, B.S. (1996) “‘Anchored Narratives’ and the Interface of Law, Psychology and Semiotics”, *Legal and Criminological Psychology* **1**, 17-45.
Jensen, C.S., Clifford, J., Elmasri, R., Gadia, S.K., Hayes, P., and Jajodia, S., eds. (1994) “A Consensus Glossary of Temporal Database Concepts”, *ACM Sigmod Record* **23**(1), 52-64.
Kahn, K.M., and Gorry, A.G. (1977) “Mechanizing Temporal Knowledge”, *Artificial Intelligence* **7**, 87-108.
Kautz, H., and Ladkin, P. (1991) “Integrating Metric and Qualitative Temporal Reasoning”, in *Proceedings of the Ninth National Conference on Artificial Intelligence - AAAI/91*. Cambridge (MA): MIT Press/AAAI Press.
King, M., Ornato, M., Zarri, G.P., Zarri-Baldi, L., and Zwiebel, A. (1977) “Ghosts in the Machine: An AI Treatment of Medieval History”, in *Proceedings of the 5th International Joint Conference on Artificial Intelligence - IJCAI/77*. San Francisco: Morgan Kaufmann.
Koomen, J. (1989) *The Timelogic Temporal Reasoning System* (Technical Report 231). Rochester (NY): Computer Science Department of the University of Rochester.
Kowalski, R., and Sergot, M. (1986) “A Logic-based Calculus of Events”, *New Generation Computing* **4**, 67-95.
Ladkin, P.B. (1986) “Time Representation: A Taxonomy of Interval Relations”, in *Proceedings of the Fifth National Conference on Artificial Intelligence - AAAI/86*. Cambridge (MA): MIT Press/AAAI Press.
Ladkin, P.B. (1987a) “Models of Axioms for Time Intervals”, in *Proceedings of the Sixth National Conference on Artificial Intelligence - AAAI/87*. Cambridge (MA): MIT Press/AAAI Press.
Ladkin, P.B. (1987b) “The Completeness of a Natural System for Reasoning with Time Intervals”, in *Proceedings of the 10th International Joint Conference on Artificial Intelligence - IJCAI/87.* San Francisco: Morgan Kaufmann.
Ladkin, P.B., and Maddux, R.D. (1987) *The Algebra of Convex Time Intervals: Short Version* (Report KES.U.87.2). Palo Alto (CA): Kestrel Institute.
Lord, T. (1996) “Copper foils the Electric's shock” (‘Coffee Break’ Column). London: *Bexley Mercury*, November 21.
McDermott, D.V. (1982) “A Temporal Logic for Reasoning About Processes and Plans”, *Cognitive Science* **6**, 101-155.
Miller, S., and Schubert, L.K. (1990) “Time Revisited”, *Computational Intelligence* **6**, 108-118.
Nebel, B., and Bürckert, H.J. (1993) *Reasoning about Temporal Relations: A Maximal Tractable Subclass of Allen’s Interval Algebra* (Research Report RR-93-11). Saarbrücken: Deutsches Forschungszentrum für Künstliche Intelligenz GmbH.
Pylyshyn, Z.W., ed. (1987) *The Robot's Dilemma : The Frame Problem in Artificial Intelligence*. Norwood (NJ): Ablex.
Tsang, E.P.K., “Time Structures for AI”, in *Proceedings of the 9th International Joint Conference on Artificial Intelligence - IJCAI/87*. San Francisco: Morgan Kaufmann.
Vilain, M., and Kautz, H. (1986) “Constraint Propagation Algorithms for Temporal Reasoning”, in *Proceedings of the Fifth National Conference on Artificial Intelligence - AAAI/86*. Cambridge (MA): MIT Press/AAAI Press.
Vilain, M., Kautz, H., and van Beek, P. (1990) “Constraint Propagation Algorithms for Temporal Reasoning: A Revised Report”, in *Readings in Qualitative Reasoning about Physical Systems*, Weld, D.S., and de Kleer, J., eds. San Francisco: Morgan Kaufmann.

Zarri, G.P. (1981) "Building the Inference Component of an Historical Information Retrieval System", in *Proceedings of the 7th International Joint Conference on Artificial Intelligence - IJCAI/81*. San Francisco: Morgan Kaufmann.

Zarri, G.P. (1992*a*) "The 'Descriptive' Component of a Hybrid Knowledge Representation Language", in *Semantic Networks in Artificial Intelligence*, Lehmann, F., ed. Oxford: Pergamon Press.

Zarri, G.P. (1992*b*). Semantic Modeling of the Content of (Normative) Natural Language Documents, in *Avignon 92 - Proceedings of the Specialized Conference on Natural Language Processing and Its Applications*. Paris: EC2&Cie.

Zarri, G.P. (1994) "A Glimpse of NKRL, the 'Narrative Knowledge Representation Language'", in *Knowledge Representation for Natural Language Processing in Implemented Systems - Papers from the 1994 Fall Symposium*, Ali, S., ed. Menlo Park (CA): AAAI Press.

Zarri, G.P. (1995) "Knowledge Acquisition from Complex Narrative Texts Using the NKRL Technology", in *Proceedings of the 9th Banff Knowledge Acquisition for Knowledge-Based Systems Workshop*, Gaines, B.R., and Musen, M., eds., vol. 1. Calgary: Department of Computer Science of the University.

Zarri, G.P. (1997*a*) "Conceptual Modeling of the 'Meaning' of Textual Narrative Documents", in *Foundations of Intelligent Systems - Proceedings of 10th International Symposium on Methodologies for Intelligent Systems, ISMIS'97*, Ras, Z, and Skowron, A., eds. Berlin: Springer-Verlag.

Zarri, G.P. (1997*b*) "NKRL, a Knowledge Representation Tool for Encoding the 'Meaning' of Complex Narrative Texts", *Natural Language Engineering - Special Issue on Knowledge Representation for Natural Language Processing in Implemented Systems*, **3**, 231-253.

Zarri, G.P. (2001) "Conceptual Modelling and Knowledge Management for Narrative Documents", in *Information Modelling and Knowledge Bases XIII*, Kangassalo, H., Welzer, T., Jakkola, H., Rozman, I., and Kawaguchi, E., eds. Amsterdam: IOS Press.

Zarri, G.P., and Azzam, S. (1997) "Building up and Making Use of Corporate Knowledge Repositories", in *Knowledge Acquisition, Modeling and Management - Proceedings of the European Knowledge Acquisition Workshop, EKAW'97*, Plaza, E., and Benjamins, R., eds. Berlin: Springer-Verlag.

Zarri, G.P., and Gilardoni, L. (1996) "Structuring and Retrieval of the Complex Predicate Arguments Proper to the NKRL Conceptual Language", in *Proceedings of the 9th International Symposium on Methodologies for Intelligent Systems, ISMIS'96*, Ras, Z, and Michalewicz, M., eds. Berlin: Springer-Verlag.

Appendix A

Let us introduce the following conventions :

- The term 'reference' (**r**) indicates the *conceptual label* of a predicative occurrence; this label is the second element of a pair of the secondary indexes. If *D* is any set of pairs, **r***D* gives the set of the corresponding references (labels).
- If *R1* and *R2* are two sets of references, $R1 \wedge R2$ gives the set of the *common references* (intersection), $R1 \vee R2$ the set of *all the references* belonging to the two sets (union), $R1 - R2$ the references of *R1* which are *not included* in *R2* .
- The sets of pairs corresponding to the nine lists of the secondary indexes are labelled *pDD*, *pF1*, *pF2* etc. as in Figure 8; *pDD(1)*, etc., is the subset of *pDD* corresponding to '*period 1*' (see again Figure 8); $\mathbf{r}pDD(1,2) = \mathbf{r}pDD(1) \vee \mathbf{r}pDD(2)$ is the set of *all the references* pertaining to the list *pDD* which are in period 1*or* period 2, i.e. all the references for which the accompanying dates *d* satisfy $d < bound2$.

Considering, for the moment, only the references in *direct perspective*, the references corresponding to period 2 are given by: $\mathbf{r}pDD(2) \vee \mathbf{r}cDD(2) \vee \mathbf{r}sDD(2)$. The references of the period 1 correspond to: $\mathbf{r}sDD(1) - \mathbf{r}pDD(1)$ — where the first term represents the eligible references and the second the associated filter (see again Figure 8); analogously, the references of period 3 are given by: $\mathbf{r}pDD(3) - \mathbf{r}sDD(3)$. Taking the union of all these subsets, we have:

(a) $\mathbf{r}pDD(2) \vee [\ \mathbf{r}pDD(3) - \mathbf{r}sDD(3)\ ] \vee \mathbf{r}cDD(2) \vee \mathbf{r}sDD(2) \vee [\ \mathbf{r}sDD(1) - \mathbf{r}pDD(1)\ ]$

We can observe now that, keeping in mind that the lists are *chronologically* ordered, a predicative occurrence cannot be classified contemporaneously in precedence (*until to …*) in period 2 and in subsequence (*starting from …*) in period 3: accordingly, $\mathbf{r}pDD(2) \wedge \mathbf{r}sDD(3)$ is *necessarily empty*. Therefore:

$$\mathbf{r}pDD(2) \vee [\ \mathbf{r}pDD(3) - \mathbf{r}sDD(3)\ ] = \mathbf{r}pDD(2) - [\ \mathbf{r}pDD(2) \wedge \mathbf{r}sDD(3)\ ] \vee$$
$$[\ \mathbf{r}pDD(3) - \mathbf{r}sDD(3)\ ] = [\ \mathbf{r}pDD(2) - \mathbf{r}sDD(3)\ ] \vee [\ \mathbf{r}pDD(3) - \mathbf{r}sDD(3)\ ] =$$
$$[\ \mathbf{r}pDD(2) \vee \mathbf{r}pDD(3)\ ] - \mathbf{r}sDD(3) = \mathbf{r}pDD(2,3) - \mathbf{r}sDD(3)\ .$$

Analogously, $\mathbf{r}sDD(2) \vee [\ \mathbf{r}sDD(1) - \mathbf{r}pDD(1)\ ] = \mathbf{r}sDD(1,2) - \mathbf{r}pDD(1)$. From the above, it follows that (a) can be reduced to:

(b) $[\ \mathbf{r}pDD(2,3) - \mathbf{r}sDD(3)\ ] \vee \mathbf{r}cDD(2) \vee [\ \mathbf{r}sDD(1,2) - \mathbf{r}pDD(1)\ ]$.

This last formula, which corresponds exactly to the graphical situation of Figure 8 when only direct perspectives are considered, is more useful than (a) because it involves only *five* sublists (instead of seven) and a *reduced number* of date comparisons with bound1 and bound2.

To extend (b) to the general case, taking into account the considerations expressed at the end of subsection 3.3 (we can avoid examining *pF1* and *sF2*), and applying to the lists *sF1* and *pF2* the same remarks we have used for the lists *sDD* and *pDD*, we can see that the references to be considered are, *in subsequence*, $[\ \mathbf{r}sF1(1,2) - \mathbf{r}pF2(1)\ ]$ and, *in precedence*, $[\ \mathbf{r}pF2(2,3) - \mathbf{r}sF1(3)\ ]$. With respect to the category 'coincidence' (a *local timestamp*), the pairs which are eligible must have a timestamp $t < bound2$ in the list *cF1* (first limit of a fork), and a timestamp $t > bound1$ in the list *cF2* (second limit of a fork). Consequently, the eligible reference can be selected by using one of the three equivalent formulations: $\mathbf{r}cF1(1,2) \wedge \mathbf{r}cF2(2,3)$; $\mathbf{r}cF1(1,2) - \mathbf{r}cF2(1)$; $\mathbf{r}cF2(2,3) - \mathbf{r}cF1(3)$. Adding all these terms to (b) we will obtain the final formal expression of the algorithm of Figure 8:

(c) $[\ \{\mathbf{r}sDD(1,2) \vee \mathbf{r}sF1(1,2)\} - \{\mathbf{r}pDD(1) \vee \mathbf{r}pF2(1)\}\ ] \vee [\ \{\mathbf{r}pDD(2,3) \vee \mathbf{r}pF2(2,3)\}$
$- \{\mathbf{r}sDD(3) \vee \mathbf{r}sF1(3)\}\ ] \vee \mathbf{r}cDD(2) \vee [\ \mathbf{r}cF1(1,2) - \mathbf{r}cF2(1)\ ]$.