Representing and Managing 'Narrative' Terrorism Information

Gian Piero Zarri
University Paris4/Sorbonne, France
gpzarri@paris4.sorbonne.fr

ABSTRACT

In this paper, we evoke first the ubiquity and the importance of the so-called 'nonfictional narrative' information, with a particular emphasis on the terrorism- and crime-related data. We show that the usual knowledge representation and 'ontological' techniques have difficulties in finding complete solutions for representing and using this type of information. We supply then some details about NKRL, a representation and inferencing environment especially created for an 'intelligent' exploitation of narrative information. We will also supply some examples concerning a "terrorism in Southern Philippines" general context to illustrate our approach.

Keywords

Narratives, terrorism, knowledge representation, inference.

INTRODUCTION

'Narrative' information concerns the account of some real-life or fictional story (a 'narrative') involving concrete or imaginary 'personages'. In this paper, we will deal with those *(multimedia) nonfictional narratives* that are typically embodied into corporate memory documents (memos, policy statements, reports, minutes, documentation archives for product development…), news stories, normative and legal texts, medical (or financial, cadastral, administrative…) records, many intelligence messages, surveillance videos or visitor logs, actuality photos and video fragments for newspapers and magazines, eLearning and Cultural Heritage material (text, image, video, sound…), plotting and narrative course of actions for videogames etc.

Note, in particular, that dealing with nonfictional narrative material is of paramount importance for analysis and management of any sort of crisis situation and, more in general, for enhancing the ability to fight terrorism and other crimes. For example, six critical mission areas have been identified in the "National Strategy for Homeland Security" report (2002). Of these, at least two, "Intelligence and Warning" and "Domestic Counter-terrorism" are based on the processing of nonfictional narrative information in order, e.g., to "… find cooperative relationships between criminals and their interactive patterns". Managing nonfictional narrative information must then be considered as an essential component of the emerging science of "Intelligence and Security Informatics" (ISI), as defined, e.g., in (Chen and Wang, 2005; Chen, 2006).

From a concrete point of view, 'nonfictional narratives' deal with the description of spatially and temporally characterized 'events' that relate, at some level of abstraction, the behavior or the state of some real-life 'actors' (characters, personages, etc.): these try to attain a specific result, experience particular situations, manipulate some (concrete or abstract) materials, send or receive messages, buy, sell, deliver etc. Note that:

- The term 'event' is taken here in its *most general meaning*, covering also strictly related notions like fact, action, state, situation, episode, activity etc.
- The 'actors' or 'personages' involved in the events *are not necessarily human beings*: we can have narratives concerning, e.g., the vicissitudes in the journey of a nuclear submarine (the 'actor', 'subject' or 'personage') or the various avatars in the life of a commercial product.
- Even if a large amount of nonfictional narratives are embodied within natural language (NL) texts, this is *not necessarily true*: narrative information is really '*multimedia*'. A photo representing a situation that, verbalized, could be expressed as "The US President is addressing the Congress" is not of course an NL document, yet it surely represents a narrative.

In this paper, we will present an Artificial Intelligence tool, NKRL, “Narrative Knowledge Representation Language”, see (Zarri, 2003; Zarri, 2005; Zarri, 2008) that is, at the same time:

- a *knowledge representation system* for describing in some detail the essential content (the ‘meaning’) of complex nonfictional narratives;
- a *system of reasoning (inference) procedures* that, thanks to the richness of the representation system, is able to *automatically* establish ‘interesting’ relationships among the represented data;
- an *implemented software environment*.

The paper will be illustrated by examples concerning (mainly) a recent application of NKRL techniques on the news stories inserted in a “Southern Philippines terrorism” corpus used in a recent European project, see [Zarri, 2005]. The success of this application has revealed NKRL as a possible, useful tool to be used in any sort of ‘defense’ and ‘crisis management’ applications. An exploratory study concerning the use of NKRL to detect specific crisis situations through an in-depth conceptual analysis of news stories about Afghanistan is presently carried on by the author of this paper in collaboration with the French “*Délégation Générale pour l’Armement*” (DGA) and a French high-tech company.

NARRATIVES, KNOWLEDGE REPRESENTATION AND THE NKRL SOLUTIONS

From a *theoretical point of view*, narratives constitute the object of a full discipline, the ‘narratology’, whose aim can be defined as that of producing *an in-depth description of the ‘syntactic/semantic structures’ of the narratives*, i.e., the narratologist is in charge of dissecting narratives into their component parts in order to establish their functions, their purposes and the relationships among them. A good introduction to this domain is (Jahn, 2005).

Even if narratology is particularly concerned with *literary analysis* (and, therefore, with ‘fictional’ narratives), these last years some of its varieties have acquired a particular importance also from an Artificial Intelligence and Computer Science point of view see, e.g., ‘storytelling’ (Soulier, 2006) and ‘eChronicles’ (Güven, Podlaseck and Pingali, G., 2005). However, a fundamental aspect of the emerging ISI techniques concerns the possibility of executing ‘in-depth reasoning’ on the original narrative material: see, e.g., the possibility of discovering hidden relationships among criminals evoked in the previous Section. Unfortunately, this aspect that is not taken into consideration sufficiently in depth in both Storytelling (mainly interested, according to the Narratology tradition, with the ‘organizational structure’ of the material under consideration) and in eChronicles (mainly interested in the accumulation of narrative materials more than in the ‘intelligent’ exploitation of their inner relationships). Implementing powerful reasoning techniques requires firstly, on the other hand, to have the possibility of *relaying on some sort of in-depth knowledge representation particularly well adapted to the domain to be taken into consideration* – in our case, the ‘security’ and ‘crisis management’ aspects of the nonfictional narrative domain.

The *n*-ary problem

Usual ontologies – both in their ‘traditional’ and ‘Semantic Web’ versions – organize the ‘concepts’ into a hierarchical structure able to supply them with an elementary form of definition through the declaration of their subsumption relationships (‘IsA’ links). A more accurate form of definition is obtained by associating with them a set of *binary relationships* – basically of the ‘property/value’ type. Semantic Web languages like RDF and OWL – and the great majority of the tools for setting up ‘ontologies’ – are then denoted as ‘*binary*’. The combination of these two representational principles is largely sufficient to provide a *static* definition of the concepts and of their properties.

Unfortunately, this is not true when we consider the *dynamic behavior* of the concepts, i.e., we want to describe their *mutual relationships* when they take part in some concrete action, situation etc. (‘events’), see the very simple narrative “John gives a book to Mary”. In this example, “give” is now an *n-ary (ternary) relationship* that, to be represented in a *complete and unambiguous way*, asks for a form of complex syntax where the arguments of the predicate, i.e., “John”, “book” and “Mary”, are introduced by some sorts of ‘*conceptual roles*’ such as, e.g., “agent of give”, “object of give” and “beneficiary of give” respectively. For representing the ‘meaning’ of narrative documents, the notion of ‘*role*’ must then be necessarily added to the traditional ‘generic/specific’ and ‘property/value’ representational principles in order to specify the *exact function* of the different components of an event within the *formal description* of this event. Note that the argument often raised stating that an *n*-ary relation can always be converted to a set of binary ones *without loss of semantics* is incorrect with respect to the last part of this sentence. In fact, it is true that, from a pure formal point of view, any *n*-ary relationship with *n* > 2 can always be reduced to a set of binary relationships. However, this fact does not change at all the *intrinsic, 'semantic'* n-*ary nature* of a simple statement like "Mary gives a book to Bill" that, to be fully understood, requires that all the constituents of the *n*-ary representation – predicates, roles, arguments of the predicate etc. – must *necessarily be managed at the same time as a coherent block*, see [Zarri, 2005b] for the formal details. The *impossibility of reducing n-ary to binary from a conceptual and semantic point of view* has, as a practical consequence, the need of using specific *n*-ary tools for reasoning and inference when complex, 'ontological' problems must be dealt with *in a not restricted way*.

Moreover, in a narrative context, we must also take care of those '*connectivity phenomena*' like causality, goal, indirect speech, co-ordination and subordination etc., that link together the basic 'elementary events'. It is very likely, in fact, that, dealing with the sale of a company, the global information to represent is something like: "Company X has sold its subsidiary Y to Z *because* the profits of Y have fallen dangerously these last years *due to* a lack of investments" or, returning to the previous example, that "John gave a book to Mary yesterday *as a* present for her birthday".

The NKRL solution

NKRL makes use of a well-formed and complete solution to the *n*-ary problem, based on the notions of 'conceptual predicate' and 'conceptual role'. Returning then to the "Mary gives a book…" simple example above, a representation that captures all the 'meaning' of this elementary narrative amounts to:

- Defining MARY_, BILL_ and BOOK_1 as 'individuals', instances of general 'concepts' like human_being and information_support. Concepts and individuals are, as usual, collected into a standard 'binary' ontology.
- Defining an *n*-ary structure organized around a predicate like GIVE or PHYSICAL_TRANSFER, and associating the above individuals with the predicate through the use of conceptual roles that specify their 'function' within the global narrative. MARY_ will then be introduced by an AGENT (or SUBJECT) role, BOOK-1 by an OBJECT (or PATIENT) role, BILL_ by a BENEFICIARY_ role.

Formally, an *n*-ary structure defined as above can be described as:

$$(L_i\ (P_j\ (R_1\ a_1)\ (R_2\ a_2)\ \ldots\ (R_n\ a_n)))\ , \qquad (1)$$

where L_i is the symbolic label identifying the particular *n*-ary structure (e.g., that corresponding to the "Mary gives a book…" example), P_j is the conceptual predicate, R_k is the generic role and a_k the corresponding argument (the individuals john_, mary_ etc.). As already stated above, the whole conceptual structure represented by (1) must be considered *globally*.

A GENERAL SURVEY OF NKRL

We will now enter into some details about NKRL, trying to show how this language can be used for applications that can be likened to crisis management and 'standard' defense tasks.

Ontologies and NKRL

NKRL innovates by adding to the usual ontologies of concepts an 'ontology of events', i.e., a new sort of hierarchical organization where the nodes correspond to *n*-ary structures called 'templates'. In the NKRL environment, the 'ontology of concepts' is called HClass (*hierarchy of classes*). The 'ontology of events' is called HTemp, *hierarchy of templates*; Figure 1 reproduces the 'symbolic labels' of part of the templates included in the Produce: branch of HTemp.

Instead of using the traditional *object (class, concept) – attribute – value* organization, templates are generated from the *n*-ary association of *quadruples* according to the general schema represented by (1). Predicates pertain to the set {BEHAVE, EXIST, EXPERIENCE, MOVE, OWN, PRODUCE, RECEIVE}, and roles to the set {SUBJ(ect), OBJ(ect), SOURCE, BEN(e)F(iciary), MODAL(ity), TOPIC, CONTEXT}; predicates and roles are then 'primitives'. An argument a_k of the predicate, see (1), denotes indirectly through a 'variable' either a simple 'concept' or a structured association ('expansion') of several concepts. In both cases, the concepts can only be chosen among those included in the HClass hierarchy; this fact, linked with the 'primitive' character of predicates and roles, allows us to reduce considerably the potential combinatorial explosion associated with formulas like (1).

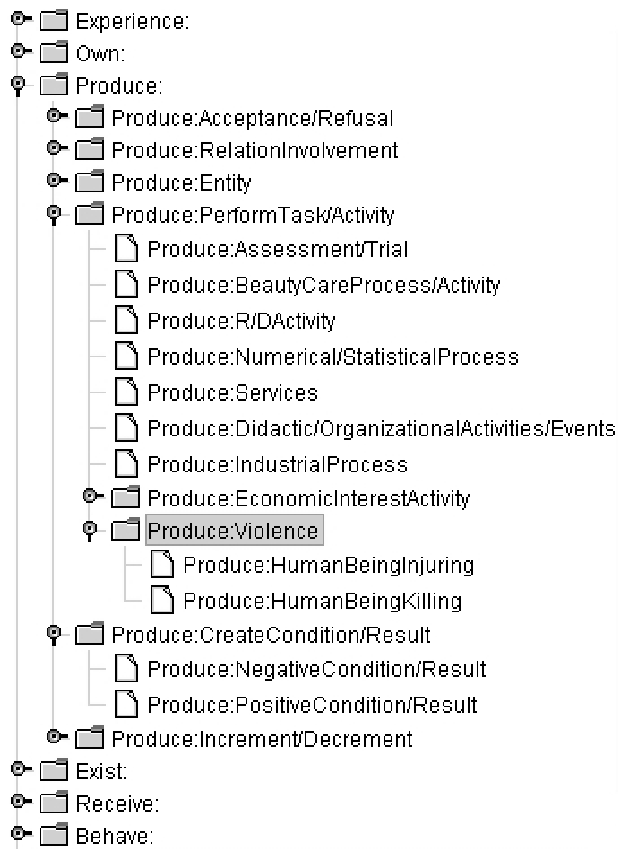

Figure 1. PRODUCE branch of HTemp, the 'ontology of events'

Templates represent formally *generic classes of elementary events* like "move a physical object", "be present in a place", "send/receive a message", "build up an Internet site", etc., see (Zarri, 2003; 2008). When a particular event pertaining to one of these general classes must be represented, the corresponding template is 'instantiated' to produce what, in the NKRL's jargon, is called a '*predicative occurrence*'. To represent then a simple 'terrorism' narrative like: "On November 20, 1999, in an unspecified village, an armed group of people has kidnapped Robustiniano Hablo", we must select firstly in HTemp the template corresponding to 'execution of violent actions', see Figure 1 and Table 1a below.

As it appears from Table 1a, the arguments of the predicate (the a_k terms in (1)) are represented by variables with associated constraints. When deriving a predicative occurrence like mod3.c5 in Table 1b, the role fillers in this occurrence must conform to the constraints of the father-template. For example, ROBUSTINIANO_HABLO (the 'BEN(e)F(iciary)' of the action of kidnapping) and INDIVIDUAL_PERSON_20 (the unknown 'SUBJ(ect)', actor, initiator etc. of this action) are both 'individuals', instances of the HClass concept individual_person: this last is a specialization of human_being_or_social_body, see, in Table 1a, the constraint on the variables *var1* and *var6*. The 'attributive operator', SPECIF(ication), of Table 1b, is one of the four operators used for the set up of 'structured arguments' ('expansions'); the (recursive) SPECIF lists, with syntax (SPECIF e_i $p_1 \ldots p_n$), are used to represent the properties or attributes that can be asserted about the first element e_i, concept or individual, of the list, see (Zarri, 2003).

Until now, we have evoked the NKRL solutions to the problem of representing *elementary (simple) events*. To deal with the '*connectivity phenomena*' already mentioned, NKRL makes use of second order structures created through *reification* of the conceptual labels (like mod3.c5 in Table 1) of the predicative occurrences. A first, simple example concerns the filler of the CONTEXT role in the occurrence mod3.c5 of Table 1b: in this case ('completive construction'), the 'context' of the kidnapping is supplied by a whole predicative occurrence, mod3.c6, telling us that the kidnapping happened when Robustiniano Hablo was on his way home with his father.

a)
name: Produce:Violence
father: Produce:PerformTask/Activity
position: 6.35
NL description: 'Execution of Violent Actions on the Filler of the BEN(e)F(iciary) Role'

```
PRODUCE    SUBJ       var1: [(var2)]
           OBJ        var3
           [SOURCE    var4: [(var5)]]
           BENF       var6: [(var7)]
           [MODAL     var8]
           [TOPIC     var9]
           [CONTEXT   var10]
           {[modulators], ≠abs}

var1  =  human_being_or_social_body
var3  =  violence_
var4  =  human_being_or_social_body
var6  =  human_being_or_social_body
var8  =  violence_, weapon_, criminality/violence_related_tool, machine_tool, general_characterising_property,
         small_portable_equipment
var9  =  h_class
var10 =  situation_, symbolic_label
var2, var5, var7  =  geographical_location
```

b)

```
mod3.c5)  PRODUCE   SUBJ      (SPECIF INDIVIDUAL_PERSON_20 weapon_wearing  (SPECIF cardinality_
                                                                   several_)): (VILLAGE_1)
                    OBJ       kidnapping_
                    BENF      ROBUSTINIANO_HABLO
                    CONTEXT #mod3.c6
                    date-1:   20/11/1999
                    date-2:

Produce:Violence
```

On November 20, 1999, in an unspecified village, an armed group of people has kidnapped Robustiniano Hablo.

c)

```
PRODUCE
SUBJ : human_being :
OBJ :   violence_
BENF : human_being :
date1 : 1/1/1999
date2 : 31/12/1999
```

Is there any information in the system concerning violence activities during 1999?

Table 1. Building up and querying predicative occurrences.

More complex examples of second order constructions are the 'binding occurrences', i.e., NKRL structures consisting of lists of symbolic labels of *predicative occurrences*. The lists are differentiated making use of specific binding operators like GOAL, COND(ition) and CAUSE, see (Zarri, 2003; 2008). Let us suppose we would now state that: "…an armed group of people has kidnapped Robustiniano Hablo *in order to* ask his family for a ransom", where the new elementary event: "the unknown individuals will ask for a ransom" corresponds to a new predicative occurrence, e.g., mod3.c7. To represent this situation completely, we must add to the two previous predicative occurrences a *binding occurrence*, e.g., mod3.c8, to link together the conceptual labels mod3.c5 (corresponding to the kidnapping occurrence, see also Table1b) and mod3.c7 (corresponding to the new predicative occurrence that describes the intended result). mod3.c8 will have then the form: "mod3.c8) (GOAL mod3.c5 mod3.c7) "; its meaning can be paraphrased as: "the activity described in mod3.c5 is focalized towards (GOAL) the realization of mod3.c7", see again (Zarri, 2003; 2008).

Inference procedures

First Level of the Inference Procedures

The *basic building block* for all the NKRL querying and inference procedures is the Fum, Filtering Unification Module. It takes as input specific NKRL data structures called 'search patterns'. Formally, these patterns correspond to *specialized/partially instantiated HTemp templates*, where the '*explicit variables*' that characterize the templates (*var*$_i$, see Table 1a) *have been replaced by concepts/individuals compatible with the constraints originally imposed on these variables*. In a search pattern, the concepts are used as '*implicit variables*'. When trying to unify a search pattern, as a *formal query*, with the predicative occurrences of the knowledge base, a concept can then *match* the individuals representing its own instances and all its subsumed concepts in HClass with their own instances. The set of predicative occurrences unified by a search pattern constitutes the *answer* to the query represented by the pattern.

A simple example of search pattern, translating the query: "Is there any information in the system about violence events occurred during the year 1999?" is reproduced in Table 1c, producing the occurrence mod3.c5 (Table 1b) as one of the possible answers.

'Hypotheses' and 'Transformations'

The *high-level inferencing operations* correspond mainly to the use of two classes of inference rules, 'hypotheses' and 'transformations'. Execution of both requires employing a real InferenceEngine, having Fum as its core mechanism.

Hypotheses correspond, in a sense, to the 'scenarios' and 'hyperscenarios' defined in (Hobbs and Potts, 2000), even if their SCML (Scenario Markup Language) is a simple XML-based DTD (Document Type Definition) that is far from providing all the representational and inferential opportunities proper to NKRL. For simplicity's sake, we will mention below an elementary example of hypothesis that implies only two 'reasoning steps' – whereas, normally, the hypothesis rules used in a 'crisis management' or 'defense' context require the use of several of these steps, see also Tables 4 and 5 below. Let us suppose we have directly retrieved, thanks to a search pattern, the occurrence conc2.c34, see Table 2a, which corresponds to the information: "Pharmacopeia, an USA biotechnology company, has received 64,000,000 dollars from the German company Schering in connection with a R&D activity". We will suppose, moreover, that this occurrence is not *explicitly* linked to other occurrences in the base by second order elements. Under these conditions, we can activate the InferenceEngine of NKRL, asking it to try to *link up automatically* the information found by the search pattern with other information present in the base. If this is possible, this last information will represent a sort of 'causal explanation' of the information originally retrieved – i.e., in our example, an 'explanation' of the money paid to Pharmacopeia by Schering. A hypothesis rule that could fit our case is hypothesis *h1* reproduced in Table 2b.

InferenceEngine works according to a backward chaining approach with chronological backtracking. The first set of operations corresponds then to the execution of the Exeprem sub-module of InferenceEngine, and consists in trying to unify, *using Fum*, the *premise* of the hypothesis, see Table 2b, and the event (the payment in our case, see conc2.c34) to be 'explained' – more exactly, in trying to unify *the event and the different search patterns derived from the premise by systematically substituting to the variables var1 and var2, see Table 2b, the associated constraints*. The premise variable *var1* can only be substituted by the constraint company_; two substitutions, *var2* = human_being and *var2* = company_ are instead possible for *var2*. A first search pattern will be then built up by substituting human_being for *var2*, i.e., a first unification with the event to explain will be tried by using a pattern corresponding to a payment done by an *individual person* instead of a *company*. This unification obviously fails.

The engine then 'backtracks' making use of a second sub-module of InferenceEngine, Reexec. The association *var2* = human_being is removed and the engine builds up a new pattern using the value *var2* = company_ that will unify the value SCHERING_ in conc2.c34. The engine can then continue the processing of the hypothesis *h1*: the two values *var1* = PHARMACOPEIA_ and *var2* = SCHERING_ are then passed to the first condition schema (*cond1*), see Table 2b. The search patterns derived from this condition schema will be tested by a third sub-module of InferenceEngine, Execond. This is called *whenever there exist conditions favorable for advancing in the hypothesis*, i.e., for being able to process a new condition schema. Exeprem and Execond perform then the forward traversal of the choice tree, with Reexec being called whenever the conditions for a backtracking exist. The difference between Exeprem and Execond consists mainly in the fact that, in an Execond context, the unification of the search patterns is tested *against the general knowledge base of occurrences to find possible unifications with these occurrences* while, in an Exeprem context, unification concerns only the patterns derived from the premise and the starting occurrence.

a)

```
conc2.c34)  RECEIVE SUBJ    (SPECIF PHARMACOPEIA_ (SPECIF biotechnology_company USA_))
                    OBJ     (SPECIF money_ usa_dollar (SPECIF amount_ 64,000,000))
                    SOURCE  (SPECIF SCHERING_ (SPECIF pharmaceutical_company  GERMANY_))
                    TOPIC    r_and_d_activity
                    date1 :
                    date2 :
```

Pharmacopeia, a USA biotechnology company, has received 64,000,000 dollars by Schering, a German pharmaceutical company, in relation to R&D activities.

b)

HYPOTHESIS *h1*

premise :

```
RECEIVE   SUBJ      var1
          OBJ       money_
          SOURCE    var2
```

var1 = company_ ; *var2* = human_being, company_

A company has received some money from another company or a physical person.

first condition schema (*cond1*) :

```
PRODUCE    SUBJ      (COORD var1 var2)
           OBJ       var3
           BENF      (COORD var1 var2)
           TOPIC     (SPECIF process_ var4)
```

var3 = mutual_relationship, business_agreement
var4 = artefact_

The two parties mentioned in the premise have concluded an agreement about the creation of a some sort of 'product'.

second condition schema (*cond2*) :

```
PRODUCE    SUBJ      var1
           OBJ       var4
           MODAL     var5
           CONTEXT   var3
```

var5 = industrial_process, technological_process

The company that received the money has actually created the product mentioned in the first condition schema.

Table 2. An example of hypothesis rule.

As usual, many deadlocks are generated in the course of the Execond operations. Without entering into further details we will, eventually, find in the base an instantiation of *cond1* corresponding to an event of the form: "Pharmacopeia and Schering have signed two agreements concerning the production by Pharmacopeia of a new compound, COMPOUND_1". The values associated with the variables *var3* (r_and_d_agreement and sale_agreement) and *var4* (COMPOUND_1) in *cond1* will then be used to create the search patterns derived from *cond2*. It will then be possible to retrieve an occurrence corresponding to the information: "In the framework of an R&D agreement, Pharmacopeia has actually produced the new compound". The global information retrieved through the execution of the hypothesis, see Table 3, can then supply a sort of 'plausible explanation' of the Schering's payment: Pharmacopiea and Schering have concluded some agreements for the production of a given compound, and this compound has been actually produced by Pharmacopeia.

With respect now to 'transformations', the underlying principle consists in using these rules to 'transform' the original query (the original search pattern) into one or more different patterns that *are not strictly 'equivalent' but only 'semantically close' to the original one.*

The start occurrence :

```
conc2.c34)    RECEIVE   SUBJ      (SPECIF PHARMACOPEIA_ (SPECIF biotechnology_company USA_))
                        OBJ       (SPECIF money_ usa_dollar (SPECIF amount_ 64,000,000))
                        SOURCE    (SPECIF SCHERING_ (SPECIF pharmaceutical_company  GERMANY_))
                        TOPIC      r_and_d_activity
                        date1 :
                        date2 :
```

Pharmacopeia, a USA biotechnology company, has received 64,000,000 dollars by Schering, a German pharmaceutical company, in relation to R&D activities.

The result for level 1 :

```
conc13.c3)  PRODUCE  SUBJ   (COORD1 PHARMACOPEIA_ SCHERING_)
                     OBJ    (COORD1 r_and_d_agreement sale_agreement)
                     BENF   (COORD1 PHARMACOPEIA_ SCHERING_)
                     TOPIC  (SPECIF synthesis_ (SPECIF COMPOUND_1 new_))
                     date1 :
                     date2 :
```

Pharmacopeia and Schering have signed two agreements (have produced two agreements having themselves as beneficiaries) concerning the production of a new compound .

The result for level 2 :

```
conc13.c7)  PRODUCE  SUBJ      PHARMACOPEIA_
                     OBJ       COMPOUND_1
                     MODAL     biotechnology_process
                     CONTEXT   r_and_d_agreement
                     date1 :
                     date2 :
```

In the framework of an R&D agreement, Pharmacopeia has actually produced the new compound .

Table 3. Final results for hypothesis *h1*.

Suppose, e.g., we ask, in a “Southern Philippines terrorism” context: “Search for the existence of some links between ObL (a well known international ‘terrorist’) and Abubakar Abdurajak Janjalani, the leader of the Abu Sayyaf group” – this group is one of the Muslim independence movements in Southern Philippines. In the absence of a direct answer, the query can be transformed into: “Search for the attestation of the transfer of economic/financial items between the two”, which could lead to retrieve: “During 1998/1999, Abubakar Abdurajak Janjalani has received an undetermined amount of money from ObL through an intermediate agent”.

From a formal point of view, transformation rules are made up of a left-hand side, the ‘*antecedent*’ – i.e. the formulation, in search pattern format, of the ‘query’ to be transformed – and one or more right-hand sides, the ‘*consequent(s)*’ – the representation(s) of one or more queries that must be substituted for the given one. A transformation rule can be expressed as: *A* (antecedent, left-hand side) ⇒ *B* (consequent(s), right-hand side). The ‘transformation arrow’, ‘⇒’, has a double meaning:

- operationally speaking, the arrow indicates the *direction* of the transformation: the left-hand side *A* (the original search pattern) is removed and replaced by the right-hand side *B* (one or more new search patterns);
- the ‘semantic’ meaning of the arrow is that information obtained through *B implies* (in a weak meaning) the information we should have obtained from *A*.

Note that many of the transformation rules used in NKRL are characterized by a very simply format implying only one ‘consequent’ schema. An example of ‘multi-consequent’ transformation is given by this specific “Southern Philippines terrorism” rule: “In a context of ransom kidnapping, the certification that a given character is wealthy or has a professional role can be substituted by the certification that i) this character has a tight kinship link with another person (first consequent schema, *conseq1*), and ii) this second person is a wealthy person or a professional people (second consequent schema, *conseq2*)". Let us then suppose that, during the search for all the possible information items linked with the Robustiniano Hablo's kidnapping, see occurrence mod3.c5 in Table 1b above, we ask to the system whether Robustiano Hablo is wealthy. In the absence of a direct answer, the system will automatically 'transform' the original query making use of the above 'kinship' rule. The result consists in saying that we do not know if Robustiano Hablo is wealthy, but we can say that his father is a wealthy businessperson, see (Zarri, 2005) for more details.

Recent developments

'Hypotheses' and 'transformations' can now work in an 'integrated' way. Integrating these two inferencing modalities corresponds to:

- From a practical point of view, transformations can now be used to try to find successful unifications with information in the knowledge base when the search patterns *derived directly* from the condition schemata of a hypothesis fail. A hypothesis deemed then to fall short could continue successfully until its normal end.
- From a more general point of view, transformations can be used to modify in an *unpredictable way* the reasoning steps (condition schemata) to be executed within a 'hypothesis' context, independently from the fact that these steps have been successful or not. This is equivalent to 'break' the *strictly predefined scenarios* proper to the hypothesis rules, and *to augment then the possibility of discovering 'implicit information' within the knowledge base and to deal with 'fragmentary' knowledge.*

A recent, detailed paper on this topic is (Zarri, 2005). We will limit ourselves to supply here some general information about the integration and an informal example.

Let us suppose that, as one of the possible answers to a question concerning the kidnapping events in Southern Philippines during 1999, we have retrieved the information: "Lieven de la Paille and Eric Drum have been kidnapped by a group of people on June 13, 1999". Making use of a hypothesis rule like that of Table 4 to 'explain' the kidnapping as a more precise 'ransom kidnapping' will give rise to a failure because of the impossibility of satisfying directly the 'intermediate' steps Cond1, Cond2 and Cond3 of *h2*, i.e., of founding *direct matches* of the search patterns derived from these condition schemata with information in the knowledge base.

- (Cond1) The kidnappers are part of a separatist movement or of a terrorist organization.
- (Cond2) This separatist movement or terrorist organization currently practices ransom kidnapping of particular categories of people.
- (Cond3) In particular, executives or assimilated categories are concerned (other rules deal with civil servants, servicemen, members of the clergy etc.).
- (Cond4) It can be proved that the kidnapped is really a businessperson or assimilated.

Table 4. Inference steps for the 'ransom kidnapping' hypothesis.

Making use of transformations in a hypothesis context means activating a hypothesis having a format *potentially equivalent* to that of Table 5. For example, the proof that the kidnappers are part of a terrorist group or separatist organization can be now obtained *indirectly*, transformation T3, by checking whether they are members of a specific subset of this group or organization. We can see that there is a whole family of transformations corresponding to the condition schemata Cond2 of *h2*. They represent variants of this general scheme: the separatist movement or the terrorist organization, or some group or single persons affiliated with them, have requested/received money for the ransom of the kidnapped. Note that, in this family of transformations, transformation T2 implies only one 'consequent' schema, whereas all the residual transformations are 'multi-consequent'.

Figure 2 illustrates the concrete use of rule T5 to satisfy the requirements of the condition schema Cond1: it is impossible to demonstrate directly that the kidnappers are part of a separatist movement, but we can show that they are part of the renegades of the Moro Islamic Liberation Front and that, at the moment of the kidnapping, the Moro Islamic (MILF) was still in control of its renegades.

CONCLUSION

In this paper, we have supplied some details about NKRL – a fully implemented, up-to-date knowledge representation and inferencing system especially created for an 'intelligent' exploitation of narrative knowledge –

trying to show how this language can also be usefully used in 'crisis management' and 'terrorism' contexts. We recall here that the main innovation of NKRL consists in associating with the traditional ontologies of concepts an 'ontology of events', i.e., a new hierarchical organization where the nodes correspond to *n*-ary structures called 'templates'.

- (Cond1) The kidnappers are part of a separatist movement or of a terrorist organization.
  - (**Rule T3, Consequent1**) *Try to verify whether a given separatist movement or terrorist organization is in strict control of a specific sub-group and, in this case,*
  - (**Rule T3, Consequent2**) *check if the kidnappers are members of this sub-group. We will then assimilate the kidnappers to 'members' of the movement or organization.*
- (Cond2) This movement or organization practices ransom kidnapping of given categories of people.
  - (**Rule T2, Consequent**) *The family of the kidnapped has received a ransom request from the separatist movement or terrorist organization.*
  - (**Rule T4, Consequent1**) *The family of the kidnapped has received a ransom request from a group or an individual person, and*
  - (**Rule T4, Consequent2**) *this second group or individual person is part of the separatist movement or terrorist organization.*
  - (**Rule T5, Consequent1**) *Try to verify if a particular sub-group of the separatist movement or terrorist organization exists, and*
  - (**Rule T5, Consequent2**) *check whether this particular sub-group practices ransom kidnapping of particular categories of people.*
  - …
- (Cond3) In particular, executives or assimilated categories are concerned.
  - (**Rule T0, Consequent1**) *In a 'ransom kidnapping' context, we can check whether the kidnapped person has a strict kinship relationship with a second person, and*
  - (**Rule T0, Consequent2**) *(in the same context) check if this second person is a businessperson or assimilated.*
- (Cond4) It can be proved that the kidnapped person is really an executive or assimilated.
  - (**Rule T6, Consequent**) *In a 'ransom kidnapping' context, 'personalities' like consultants, physicians, journalists, artists etc. can be assimilated to businesspersons.*

Table 5. 'Ransom kidnapping' rule in the presence of transformations.

REFERENCES

1. (2002). *National Strategy for Homeland Security*. Washington (DC): Office of Homeland Security.
2. Chen, H. (2006). *Intelligence and Security Informatics for International Security: Information Sharing and Data Mining*. New York: Springer.
3. Chen, H., and Wang, F.-Y. (2005). Artificial Intelligence for Homeland Security. *IEEE Intelligent Systems* **20**(5): 12-16.
4. Güven, S., Podlaseck, M., and Pingali, G. (2005). PICASSO: Pervasive Information Chronicling, Access, Search, and Sharing for Organizations. In: *Proceedings of the IEEE 2005 PerCom Conference*. Los Alamitos (CA): IEEE Computer Society Press.
5. Hobbs, R.L., and Potts, C. (2000). Hyperscenarios: A Framework for Active Narratives. In: *Proceedings of the 38th Annual ACM Southeast Regional Conference*. New York: ACM.
6. Jahn, M. (2005). *Narratology: A Guide to the Theory of Narrative* (version 1.8). Cologne: English Department of the University (http://www.uni-koeln.de/~ame02/pppn.htm).
7. Soulier, E., ed. (2006). *Le Storytelling, concepts, outils et applications*. Paris : Lavoisier.

9\. Zarri, G.P. (2003). A Conceptual Model for Representing Narratives. In: *Innovations in Knowledge Engineering*. Adelaide: Advanced Knowledge International.

0\. Zarri, G.P. (2005a). Integrating the Two Main Inference Modes of NKRL, Transformations and Hypotheses. *Journal on Data Semantics (JoDS)* **4**: 304-340.

10\. Zarri, G.P. (2005b). An *n*-ary Language for Representing Narrative Information on the Web. In: *SWAP 2005, Semantic Web Applications and Perspectives – Proceedings of the 2nd Italian Semantic Web Workshop* (CEUR-WS.org/Vol-166). Aachen: Sun SITE Central Europe.

11\. Zarri, G.P. (2008, in press). *Representation and Management of Narrative Information.* London: Springer.

```
            **********  the result for condition 1   ****************
*********************************************************************************************
***Entering an internal transformation module : internal level 1 ***********************
*********************************************************************************************
***                            The model to transform
***
***:
***      ]  BEHAVE
***      SUBJ(ect)  : INDIVIDUAL_PERSON_68 :
***      MODAL(ity) : part_of
***      TOPIC      : separatist_movement
***      {}
***      date-1     :null
***      date-2     :null
***      is instance of:
***
***            **********  the result for consequent 1   ****************
***mod57.c17:
***      ]  OWN
***      SUBJ(ect)  : MORO_ISLAMIC_LIBERATION_FRONT :
***      OBJ(ect)   : control_ :
***      TOPIC      : MORO_ISLAMIC_RENEGADE
***      {poss }
***      date-1     :24/3/1999
***      date-2     :null
***      is instance of:Own:Control
***Natural language description :
***On March 24, 1999, it is possible that the MILF is still in control of its renegades.
***
***            **********  the result for consequent 2   ****************
***mod33.c10:
***      ]  BEHAVE
***      SUBJ(ect)  : ( SPECIF INDIVIDUAL_PERSON_68 ( SPECIF cardinality_ several_ ) ) :
***      MODAL(ity) : part_of
***      TOPIC      : MORO_ISLAMIC_RENEGADE
***      {obs }
***      date-1     :13/6/1999
***      date-2     :null
***      is instance of:Behave:Member
***Natural language description :
***The kidnappers are member of a group of renegades of the Moro Islamic Liberation Front
```

Figure 2. Application of the transformation rule about the 'renegades'.