Using the Formal Representations of "Elementary Events" to Set Up Computational Models of Full "Narratives"

Gian Piero Zarri

Abstract In this chapter, we describe the conceptual tools that, in an NKRL context (NKRL = Narrative Knowledge Representation Language), allow us to obtain a (computer-usable) description of full "narratives" as logically structured associations of the constituting (and duly formalized) "elementary events." Dealing with this problem means, in practice, being able to formalize those "connectivity phenomena"—denoted, at "surface level," by logico-semantic coherence links like causality, goal, co-ordination, subordination, indirect speech, etc.—that assure the conceptual unity of a whole narrative. The second-order, unification based solutions adopted by NKRL in this context, "completive construction" and "binding occurrences," allow us to take into account the connectivity phenomena by "reifying" the formal representations used to model the constitutive elementary events. These solutions, which are of interest from a general digital humanities point of view, are explained in some depth making use of several illustrating examples.

Introduction

NKRL, the "Narrative Knowledge Representation Language," is both a *conceptual modeling tool* (Zarri 2009) and a (fully implemented) *computer science environment* (Zarri 2009: Appendix A, 2010), created for dealing with "*narratives*" in an innovative way. In a nutshell a narrative—see, e.g., Bal (1997), Jahn (2005)—is a general unifying framework used for relating real-life or fictional stories involving concrete or imaginary characters and their relationships. A narrative materializes actually as (multimedia) work of speech, writing, song, film, television, video game, photography, theater, etc.

Even if the conceptual structures and the procedures used in NKRL for dealing with narratives are quite general, the concrete applications of this language have concerned mainly *non-fictional narratives*. While fictional narratives have prin-

G.P. Zarri (✉)
Sorbonne University, STIH Laboratory, Maison de la Recherche – 28, rue Serpente, 75006 Paris, France
e-mail: zarri@noos.fr

S. Hai-Jew (ed.), *Data Analytics in Digital Humanities*, Multimedia Systems and Applications, DOI 10.1007/978-3-319-54499-1_2

cipally an entertainment value and represent a narrator's account of a story that happened in an imaginary world (a novel is a typical example of fictional narrative), non-fictional narratives are deeply rooted in the everyday world. They are conveyed, e.g., by NL supports under the form of news stories, corporate memory documents (memos, reports, minutes, etc.), normative and legal texts, medical records, etc. But they can also be represented by multimedia documents like audio and video records, surveillance videos, actuality photos for newspapers and magazines, etc. A photo representing President Obama addressing the Congress, or a short video showing three nice girls on a beach, can be considered as "non-fictional narrative" documents even if they are not, of course, NL documents. We can note immediately the *ubiquitous character* of this sort of (non-fictional) narrative resources and their *general economic importance*.

In agreement with the most recent theoretical developments in the "narratology" domain, NKRL understands a (fictional or non-fictional) "narrative" under the form of *a "coherent" (i.e., logically connected) stream of spatio-temporally constrained "elementary events."* It is then evident that, in an NKRL context, a fundamental step for the modeling of (whole) narratives concerns the possibility of finding a complete, logically correct, and computer-exploitable *formal representation* of the different elementary events that makes up the stream. This topic has been dealt with in-depth in several recent publications, see Zarri (2009, 2010, 2011a, 2015) for example, and will only be alluded to in passing in this paper.[1] This last focuses, instead, on the description of *two specific mechanisms* used in NKRL for formalizing some *relational phenomena* that are particularly relevant in the context of the *logical coherence of the narrative stream* evoked above—and that are of interest, in general, from a cognitive point of view. They are (1) *the need to refer* to an elementary event (or a full narrative) *as an argument of another event* (see, e.g., an event *X* where someone speaks about *Y*, where *Y* is itself an elementary event or a logically coherent set of events), and (2) *the need for associating together through some sort of logico-semantic relationships* elementary events or narratives that could also be regarded as *independent entities* (as an elementary event or full narrative *X* being linked to another event or narrative *Y* by causality, goal, coordination,

[1]We will only mention that, in an NKRL context, each elementary event is recognized—as usual, see Matsuyoshi et al. (2010) for example—thanks to the detection of "generalized predicates" within the natural language (NL) formulation of the whole stream. These predicates correspond then to the usual tensed/untensed "verbs," but also to "adjectives" ("... worth several dollars ...", "...a dormant volcano...", nouns ("...Jane's amble along the park...", "...a possible attack..."), etc., when they have a predicative function. Let us look, e.g., at two simple narratives proper to a recent NKRL application concerning the conceptual analysis of accident messages in an industrial context, see Zarri (2011b), like: "The control room operator recognizes an alarm" and "The control room operator presses a button to initialize a new start-up sequence." In the first example, the whole narrative is formed of a unique elementary event, detected via the presence of the predicate "recognize." In the second, the narrative is formed of two elementary events, identified thanks to the occurrence of the two predicates "press" and "initialize." The whole narrative is eventually fully formalized by using a "second order operator" in the GOAL style to link together the formal expressions of the two elementary events, see later in this chapter.

alternative, etc. relationships). In an NKRL context, the first relational phenomenon is called "*completive construction*" and the second "*binding occurrences*"; the two are collectively denoted as "*connectivity phenomena*." Passing from the *deep, conceptual level* proper to NKRL to a *surface, linguistic level*, they can then evoke the classical "*textual cohesion*" aspects described, among many others, by Halliday and Hasan (1976) and Morris and Hirst (1991) and the "*contingency phenomena*" recently analyzed, e.g., by Hu et al. (2013) in a film scene descriptions context. At surface level, the presence of connectivity phenomena is recognized through the existence of "*cues,*" i.e., *syntactic/semantic features* like causality, goal, co-ordination, subordination, indirect speech, etc.

In the following, we will present first, section "Basic Notions About NKRL", a quick recall of some fundamental principles about NKRL, and the associate terminology. Section "Linking Elementary Events" is the central component of the paper, showing how the basic building blocks corresponding to the elementary events can be associated within wider structures in order to take the "connectivity phenomena" into account. Section "Querying/Inference Procedures" mentions briefly the querying/inference mechanisms of NKRL, referring the reader to other NKRL publications for additional details. Section "Related Work" concerns some comparisons with work related to the specific NKRL's approach; section "Conclusion" supplies, eventually, a short "Conclusion."

Basic Notions About NKRL

NKRL innovates with respect to the current ontological paradigms—e.g., those developed in a Semantic Web (SW) context, see Bechhofer et al. (2004), W3C OWL Working Group (2012)—by adding an "ontology of elementary events" to the usual "ontology of concepts."

The ontology of concepts is called HClass ("hierarchy of classes") in an NKRL context and includes presently (February 2016) more than 7500 "standard" concepts—"standard" meaning here that the "properties" or "attributes" used to define a given concept are simply expressed as *binary (i.e., linking only two arguments) relationships* of the "property/value" type. From a purely "formal" point of view HClass—see Zarri (2009, pp. 43–55, 123–137)—is not fundamentally different, then, from the ontologies that we can build up by using the frame version of Protégé (Noy et al. 2000).

The ontology of elementary events is, by contrast, *a new sort of hierarchical organization* where the nodes correspond to *n*-ary structures called "*templates,*" represented schematically according to the syntax of Eq. (1) below. This ontology is then denoted as HTemp (hierarchy of templates) in NKRL. Templates, in opposition to the "*static/basic*" notions (like "human being," "amount," "color," "artefact," "control room," "valve," "level of temperature," etc.) denoted by the HClass concepts, take into account the "*dynamic/structured*" component of the narrative information. They can be conceived, in fact, as *the canonical, formal representation* *of generic classes of spatio-temporally characterized elementary events* like “move a physical object,” “be present in a place,” “having a specific attitude towards someone/something,” “produce a service,” “asking/receiving an advice,” etc.

$$\left(L_i \left(P_j \left(R_1\ a_1\right)\ \left(R_2\ a_2\right) \ldots \left(R_n\ a_n\right)\right)\right) \quad (1)$$

In Eq. (1), L_i is the “*symbolic label*” identifying (*reifying*) the particular *n*-ary structure corresponding to a specific template—as we will see in the following, these reification operations are of a fundamental importance in the context of the association of (formalized) elementary events. P_j is a “*conceptual predicate*.” R_k is a generic “*functional role*” (Zarri 2011a) used to specify the logico-semantic function of its “filler” a_k with respect to the predicate. a_k is then a “*predicate argument*” introduced by the role R_k.

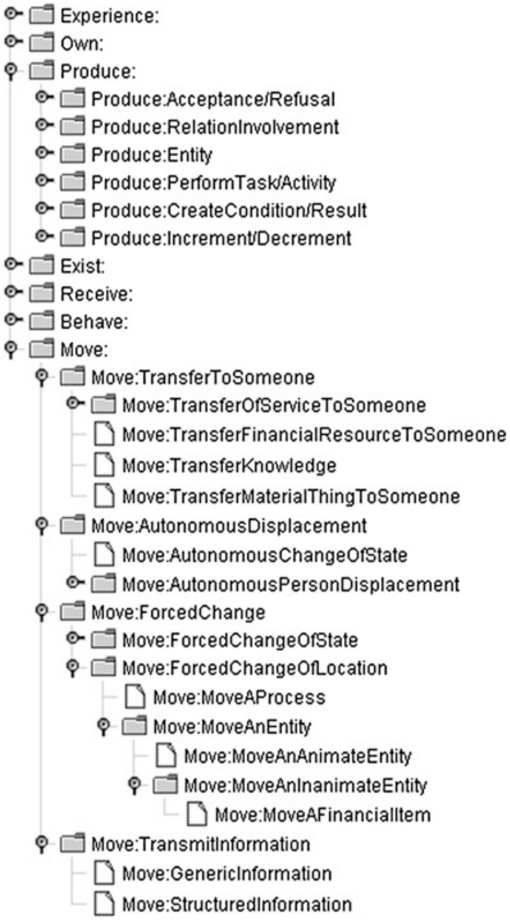
When a template following the general syntax of Eq. (1) and denoted as Move:TransferMaterialThingsToSomeone in NKRL is *instantiated* to provide the representation of a simple elementary event like “Bill gives a book to Mary,” the predicate P_j (MOVE) will introduce its three arguments a_k, JOHN_, MARY_, and BOOK_1 (“*individuals,*” i.e., instances of HClass concepts) through the three functional relationships (R_k roles) SUBJ(ect), BEN(e)F(iciary), and OBJECT. The global *n*-ary construction is then *reified* through the symbolic label L_i *and necessarily managed as a coherent block at the same time*. The instances of templates are called “*predicative occurrences*” and correspond then to the representation of specific elementary events, see the examples in the following section.

Note that, to avoid the ambiguities of natural language and any possible combinatorial explosion problem—see Zarri (2009, pp. 56–61)—both the conceptual predicate of Eq. (1) and the associated functional roles are “*primitives*.” Predicates P_j pertain then to the set {BEHAVE, EXIST, EXPERIENCE, MOVE, OWN, PRODUCE, RECEIVE}, and the roles R_k to the set {SUBJ(ect), OBJ(ect), SOURCE, BEN(e)F(iciary), MODAL(ity), TOPIC, CONTEXT}. Figure 1 reproduces a fragment of the HTemp hierarchy that displays, in particular, the conceptual labels of some off-springs of the Move: (and Produce:) sub-hierarchies. As it appears from this figure, HTemp is structured into *seven branches*, where each branch includes only the templates created—according to the general syntax of Eq. (1)—around one of the seven predicates (P_j) admitted by the NKRL language.

For the sake of clarity, we reproduce in Table 1 the full formalism corresponding to the template Move:TransferMaterialThingsToSomeone (see also Fig. 1) used to produce the predicative occurrence formalizing the elementary event “Bill gives a book to Mary” of the above example. The constituents (as SOURCE, MODAL, (*var2*), etc. in Table 1) included in square brackets are *optional*. HTemp includes presently (February 2016) more than 150 templates, very easy to specialize and customize, see, e.g., Zarri (2009, pp. 137–177, 2014).

As we can see from Table 1, the arguments of the predicate (the a_k terms in Eq. 1) are actually represented by variables (var_i) with *associated constraints*. These are expressed as *concepts or combinations of concepts*, i.e., using the terms of the NKRL standard ontology of concepts (HClass). When creating a

Fig. 1 Partial image of HTemp, with the Produce: and Move: branches partly unfolded



predicative occurrence as an instance of a given template, the constraints linked to the variables are used to specify the *legal sets of HClass terms (concepts or individuals) that can be substituted for these variables within the occurrence*. In the predicative occurrence corresponding to the above example, we must then verify that JOHN_ and MARY_ are real HClass instances of individual_person, a specific term of human_being_or_social_body, see the constraints on the SUBJ and BENF functional roles of the template in Table 1. BOOK_1, as an instance of the HClass concept book_, verifies in turn the constraint artefact_ associated with the filler of the OBJ role. book_ is, in fact, a specific term of artefact_ through intermediate HClass concepts like, e.g., information_support.[2]

---

[2]We can also note that "*determiners*" (or "*attributes*") can be added to templates or predicative occurrences to introduce *further details* about the basic core, "symbolic label/predicate/functional roles/arguments of the predicate," see Eq. (1), of their formal representation (Zarri 2009, pp.

Table 1 A template of the Move: branch of HTemp

name: Move:TransferMaterialThingToSomeone
father: Move: TransferToSomeone
position: 4.21
NL description: 'Transfer a Material Thing (e.g., a Product, a Letter…) to Someone'

MOVE	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	=	artefact_
var4	=	human_being_or_social_body
var6	=	human_being_or_social_body
var8	=	sector_specific_activity, service_
var9	=	sortal_concept
var10	=	situation_, symbolic_label
var2, *var5*, *var7*	=	location_

Linking Elementary Events

In section "Introduction", we have mentioned those "*connectivity phenomena*"—signaled, at "*surface linguistic level,*" by the presence of NL syntactic/semantic features like causality, goal, indirect speech, co-ordination, subordination, etc.—that assure the logical coherence among the components (elementary events) of a specific narrative.

In NKRL, the connectivity phenomena are dealt with making use of *Higher Order Logic (HOL) structures*—according to HOL, a predicate can take one or more other predicates as arguments—obtained from the *reification* of generic (i.e., not only predicative, see below) occurrences. Concretely, the reification is based on the use of the *symbolic labels* denoted by the L_i terms in Eq. (1) above. "Reification" is intended here—as usual in a Knowledge Representation context—as the possibility

70–86). In particular, determiners/attributes of the "*location*" type—represented in general by *lists of instances* of the HClass location_ concept and of its specialization terms—can be associated through the "colon" operator, ":", with the *arguments of the predicate* (i.e., the fillers) introduced by the SUBJ, OBJ, SOURCE, and BENF functional roles of a template, see Table 1. Another important category of determiners/attributes associated, in this case, to a *full, well-formed template or predicative occurrence* to particularize its meaning are constants of the "*modulator*" type. Modulators are classed into three categories: *temporal* (begin, end, obs(erve)), *deontic* (oblig(ation), fac(ulty), interd(iction), perm(ission)), and *modal modulators* (for, against, wish, ment(al), etc.). See the examples in the sections below for some additional information about the determiners/attributes.

of creating new objects ("*first class citizens*") out of already existing entities and to "*say something*" about them without making explicit reference to the original entities.

Completive Construction

A first example of HOL connection between elementary events is represented by the "completive construction." This consists in using as *filler* of a functional role in a predicative occurrence pc_i the *symbolic label* L_j of another (*generic*) occurrence c_j. We can note immediately that the c_j (*indirectly*) used as fillers can correspond not only to predicative occurrences pc_i, but also to those "binding occurrences" bc_i we will introduce in the next sub-section. Constraints proper to the "completive construction" category of NKRL HOL constructions are:

- *Only* the OBJ, MODAL, TOPIC, and CONTEXT functional roles of pc_i can accept as filler the symbolic label L_j of a c_j, and *only one of these four roles* can be utilized in the context of a *specific instantiation* of the completive construction mechanism.
- L_j must denote a *single* symbolic label, i.e., any "structured filler" represented under the form of an association of labels *cannot be used* in a completive construction framework.
- For (software) implementation reasons, this *single* label L_j is prefixed, in the "external" NKRL format used in the examples of this chapter, by a "*sharp,*" "#", code. The general format of a completive construction filler corresponds then, actually, to #symbolic_label, see the examples below. Note that symbolic_label is a *regular concept* of HClass, the standard NKRL ontology of concepts. This concept has then as *specific instances* all the *actual labels* used to denote (predicative and binding) occurrences in a specific NKRL application.

As a first example, we reproduce in Table 2 a fragment of a scenario concerning a recent Ambient Assisted Living (AAL) application of NKRL. In this fragment, a robot reminds John, an ageing person, of the obligation to lock the front door. The modulator oblig(ation), see Note 2, has been used in aal9.c12 to denote the *absolute necessity* of locking the front door. The "temporal determiners/attributes" date-1/date-2 are used in association with predicative occurrences to introduce the temporal information proper to the original elementary event, see, e.g., Zarri (2009, pp. 80–86, 194–201).[3]

[3]With respect to the *(semi-)automatic* synthesis of predicative occurrences like aal9.c11 and aal9.c12 in Table 2 and all the others mentioned in this paper—more in general, with respect to the *(semi-)automatic "translation"* from Natural Language (NL) into NKRL—several prototypes exist. All of them derive, basically, from the algorithms developed in the eighties in the framework of the RESEDA (in French, *Reseau Sémantique Documentaire*) project, an NKRL's ancestor, see Zarri (1983). Very in short, an up-to-date syntactic parser in the style of the well-known Stanford

Table 2 An example of completive construction

```
aal9.c11)   MOVE   SUBJ      ROBOT_1
                   OBJ       #aal9.c12
                   BENF      JOHN_
                   MODAL     audio_warning
                   date-1:   11/4/2011/17:35
                   date-2:
```

Move:StructuredInformation (4.42)

On 11/4/2011, at 17h35, the robot reminds John through an audio message of what is described in the predicative occurrence aal9.c12.

```
aal9.c12)   MOVE   SUBJ      JOHN_
                   OBJ       FRONT_DOOR_1: (unlocked_, locked_)
                   { oblig }
                   date-1:   11/4/2011/17:35
                   date-2:
```

Move:ForcedChangeOfState (4.12)

*On 11/4/2011, at 17h35, John must necessarily, modulator "*oblig*(ation)", lock the front door.*

We can note, in the formal encoding of Table 2, the use of predicative occurrences corresponding to *two different types* of Move: templates. The first, Move:StructuredInformation (a specialization of the Move:TransmitInformation template) is necessarily used in NKRL to represent, according to the "completive construction" modalities, the *transmission of some complex information* whose content is described by one or more predicative occurrences. The second, Move:ForcedChangeOfState, a specialization of Move:ForcedChange, is used when an agent (SUBJ) moves an entity (OBJ = physical object, animate entity, process, etc.) from an initial state to a final one. In this case, the *initial state* is represented by the *first position* of the location list associated (through the ":" operator, see again Note 2 above) with the filler of the OBJ role in the predicative occurrence (aal9.c12 in Table 2) that represents the moving. The *final state* is represented by the *second position* of the same list. Possible *intermediary states* can be symbolized as the *ordered sequence* of locations included between the first and last position of the list. Note also that the procedure used in aal9.c12 to denote a forced change of location

parser (Klein and Manning 2003) is used for a preliminary syntactic analysis of the NL text corresponding to the NKRL structures to be generated. A set of generalized "if-then" rules is then activated, where the "antecedents" of the rules denote fragments of the syntactic analysis able to "trigger" NKRL template-like structures (represented by the "consequents" of the rules) if some specific lexico-syntactic conditions are recognized. HClass and lexico-semantic resources like WordNet, VerbNet, Roget Thesaurus, etc. are used to complete the "translation" operations. A recent system in this style is described, e.g., in Ayari et al. (2013).

is valid in general, i.e., also when the elements of the vector associated with the OBJ filler correspond to *concrete "physical" locations* and not to "abstract" states.

Binding Occurrences

A second, more general way of linking together NKRL elementary events within the scope of a full narrative consists in making use of "*binding occurrences*." These are lists labelled with specific "*binding operators*" Bn_i whose arguments arg_i are represented (*reification*) by symbolic labels L_j of (*predicative or binding*) c_j occurrences. The general expression of a binding occurrence bc_i is then:

$$(\mathrm{Lb}_k \ \ (\mathrm{Bn}_i \ \mathrm{L}_1 \ \mathrm{L}_2 \ldots \mathrm{L}_n))\,, \tag{2}$$

where $\mathrm{Lb_k}$ is now the symbolic label identifying the whole (*autonomous*) binding structure. Unlike templates and predicative occurrences, binding occurrences are then characterized by the absence of any predicate or functional role. The eight binding operators are listed (and defined) in Table 3.

The binding occurrences bc_i must necessarily conform to the following mandatory restrictions to be considered as *well formed*:

- Each term (argument) L_j that, in a binding list, is associated with one of the operators of Table 3, denotes exactly a *single* predicative or binding occurrence c_j *described externally to the list*. Therefore, the arguments L_j are always *single terms* and cannot consist of lists of symbolic labels associated in turn with binding operators.
- In the binding occurrences of the ALTERN, COORD, and ENUM type, *no restriction is imposed on the cardinality of the list*, i.e., on the possible number of terms (arguments) L_j.
- In the binding occurrences labelled with CAUSE, REFER, GOAL, MOTIV, and COND, on the contrary, *only two arguments Lm and Ln are admitted*, see Table 3. The binding occurrences labelled with these five binding operators are then simply of the form: $(\mathrm{Lb}_k \ (\mathrm{Bn}_i \ \mathrm{L}_m \ \mathrm{L}_n))$. In these lists, the arguments L_m and L_n can denote, in general, *either a predicative or a binding occurrence*: an exception is represented by the COND binding occurrences, where the first argument, L_m, *must correspond necessarily to a predicative occurrence* pc_i, see again Table 3.

To supply now a first idea of the modalities of use of the "binding occurrences" tools let us suppose, see Table 4, we would like to formalize in NKRL terms the following situation: "From the main control room of the GP1Z plant, the production activities leader pushes the SEQ1 button in order to start the auxiliary lubrication pump M202." According to what was explained in Note 1 above, recognizing the presence of two surface predicates like "push" and "start" implies the creation, at "*deep level,*" of two different elementary events (two predicative occurrences). Moreover, the presence of "in order of" denotes the existence of some

Table 3 Binding operators of NKRL

Operator	Acronym	Mnemonic description
Alternative	ALTERN	The "*disjunctive*" operator. *Only a specific elementary event* corresponding to one of the terms included in the list of the associated L_j labels must be taken into account, but this term is not known *a priori*.
Co-ordination	COORD	The "*collective*" operator. *All the elementary events* corresponding to all the L_j terms of the list must *necessarily* be considered *together*.
Enumeration	ENUM	The "*distributive*" operator. *Each elementary event corresponding to each* L_j *term* of the list must be taken into account, but they are dealt with in a separate way.
Cause	CAUSE	*Only two* L_j *terms* can appear in a CAUSE binding occurrence (and in all the binding occurrences designated by one of the following binding operators of this Table). CAUSE is the "*strict causality*" operator, introducing a *necessary and sufficient causal relationship between the elementary events denoted by the first,* L_m*, and the second,* L_n*, arguments of the list*, the latter event explaining the former.
Reference	REFER	The "*weak causality*" operator, introducing a necessary *but not sufficient* causal relationship between the elementary events denoted by the first, L_m, and the second, L_n, arguments of the list.
Goal	GOAL	The "*strict intentionality*" operator: the elementary event denoted by the first argument L_m is *necessary* to bring about the event denoted by the second argument, L_n, and this second event is *sufficient* to explain the first. The predicative occurrence(s) corresponding to the second argument is/are marked as "*uncertain,*" operator "*", see Zarri (2009, p. 71).
Motivation	MOTIV	The "*weak intentionality*" operator: the event denoted by the first argument L_m is *not necessary* to bring about the event denoted by L_n, but this last is *sufficient* to explain the first. The predicative occurrence(s) denoted by the second argument is/are marked as "uncertain," operator "*".
Condition	COND	The (single) *predicative occurrence* pc_i denoted by L_m represents an event that *could occur* if the predicative or binding occurrence c_j denoted by L_n *should take place*. pc_i is necessarily associated with a modal modulator "poss(ibility)"; the (single or multiple) predicative occurrence(s) corresponding to L_n is/are necessarily marked as "uncertain."

connectivity phenomena that brings together the two events. The first occurrence of Table 4, virt2.c32, corresponds then to the action of "pushing": button_pushing is an HClass concept, specialization of another (high level) concept, activity_, through device_use and other HClass terms. Note that the TOPIC role has the general meaning of "apropos of," "concerning," "with reference to," etc. The second occurrence, virt2.c33, represents the (possible) result of the action of "pushing," i.e., the shift of the auxiliary lubrication pump from an "idle" to a "running" state. Note, in this case, *the assimilation of the two states to "locations,"* with the original state

Table 4 Binding and predicative occurrences

```
virt2.c32) PRODUCE    SUBJ   INDIVIDUAL_PERSON_102: (GP1Z_MAIN_CONTROL_ROOM)
                      OBJ    button_pushing
                      TOPIC  SEQ1_BUTTON
                      date-1:  16/10/2008/08:26
                      date-2:

Produce:PerformTask/Activity (6.3)

*virt2.c33)  MOVE   SUBJ      AUXILIARY_LUBRICATION_PUMP_M202: (idle_)
                    OBJ       AUXILIARY_LUBRICATION_PUMP_M202: (running_)
                    date-1:   16/10/2008/08:26
                    date-2:

Move:AutonomousChangeofState (4.32)

virt2.c30)   (GOAL  virt2.c32  virt2.c33)
```

occupying the first place of the location list associated with the SUBJ's filler and the final state occupying the second position of this list (see also the location list associated with the OBJ's filler in the occurrence aal9.c12 of Table 2 above).

To encode now the "connectivity phenomena" information, we must introduce a *binding occurrence* virt2.c30 to link together the conceptual labels virt2.c32 (denoting the planning activity) and virt2.c33 (denoting the intended result). This binding occurrence will be labelled using the GOAL operator introduced in Table 3 and involving, as already stated, *only two arguments*. The global meaning of virt2.c30 is then: "The activity described in virt2.c32 is focalized towards (GOAL) the realization of virt2.c33." In agreement with the semantics of the GOAL operator (see Table 3) virt2.c33, the "*result*," is characterized by the presence of an *uncertainty attribute code*, "*", to indicate that, at the moment of "pushing," the real instantiation of a situation corresponding to "pump running" *cannot be categorically affirmed* (Zarri 2009, p. 71).

Note that, in Table 4, we have used a Move:AutonomousChangeOfState template instead of the template Move:ForcedChangeOfState that appears in Table 2. In NKRL, each elementary event is, in fact, *autonomously modelled*. Should virt2.c33 really take place, we will see the pump starting to move without any apparent human participation. In contrast, in Table 2, John must explicitly step in to carry out the locking of the door.

NKRL Modelling of Full Narratives

The second order (HOL) structures of NKRL, completive construction and binding occurrences, allow us to take correctly into account the connectivity phenomena; accordingly, they play also a crucial role in the modelling of *full narratives* (or scenarios, complex events, knotty circumstances, etc.). As an example, we supply in Table 5 the NKRL representation of a full narrative proper to the context of the "accident messages" application already mentioned: "On November 1st, 2008, at 10h15, the start-up procedure of the GP1Z turbine was stopped by the production activities leader, given that he had been informed by a field operator of the presence of an oil leakage concerning an auxiliary lubrication pump."

The (*mandatory*) starting point for the creation of the NKRL formal representation of any sort of complete narrative consists in the set-up of *a binding occurrence listing the main topics dealt within this narrative*. This "upper level" occurrence corresponds frequently, as in the present case (see virt3.c1), to a binding occurrence of the COORD(ination) type (COORD is one of the "binding operators" listed in Table 3 above). We have then assumed here that the narrative was formed of *three independent but strictly connected items*, relating the first (virt3.c2) the narrative's core, i.e., the specific causes of the turbine's stop, and giving the second (virt3.c3) and the third (virt3.c4) auxiliary information about the jobs of the two involved people. But the upper level binding occurrence could also consist, e.g., of an ENUM(eration) relationship, of a CAUSE binding occurrence, etc.: *all the operators listed in Table 3 can then be used in this role*. Having set-up the top level of the conceptual representation, the different blocks listed in this binding occurrence are *successively expanded* and the corresponding elementary events *separately encoded*.

Let us consider, e.g., the binding occurrence virt3.c6 that illustrates the two (*strictly associated*, COORD) precise reasons of the stop. The first is described in the completive construction formed by the *indirect* inclusion of virt3.c8, the "*message*" signaling the leakage, as *OBJ(ect)* of the *transmission of information* between the two individuals INDIVIDUAL_PERSON_104 and INDIVIDUAL_PERSON_106 represented by the predicative occurrence virt3.c7. Note that, thanks to the completive construction mechanism, the two occurrences virt3.c7/virt3.c8 perform actually as a *unique conceptual unit*. Note also that the insertion of the symbolic label #virt3.c8 within the arguments of the binding occurrence virt3.c6 concerns only, once again, *some coherence controls proper to the NKRL software*, and does not alter at all the *actual cardinality (two)* of the COORD's arguments in virt3.c6. The second reason of the stop is described in virt3.c9: when the leakage is detected we can note, temporal modulator obs(serve), that *the auxiliary pump is linked to the turbine* (coupled_with is an HClass concept, specialization of binary_relational_property). "obs"—see Note 2 and Zarri (2009, pp. 71–75)—is a "*temporal modulator,*" used to indicate that the situation described in the associated predicative occurrence is *certainly true* at the specific date stored in the date-1 temporal attribute of the occurrence (see also the two "status" occurrences virt3.c3 and virt3.c4). We do not care then, for lack of interest, lack of information or for the sake of conformity with the original wording of the narrative, about the *real duration* of this situation, which surely extends in time before and after the given date.

Table 5 NKRL modelling of a full narrative

```
virt3.c1)    (COORD  virt3.c2  virt3.c3  virt3.c4)
```

The conceptual model of the narrative is formed of three components.

```
virt3.c2)    (CAUSE  virt3.c5  virt3.c6)
```

The first component consists of a CAUSE *binding relationship.*

```
virt3.c5)    PRODUCE   SUBJ      INDIVIDUAL_PERSON_102: (GP1Z_MAIN_CONTROL_ROOM)
                       OBJ       activity_stop
                       TOPIC     (SPECIF turbine_startup GP1Z_TURBINE)
                       date-1:   1/11/2008/10:15, (1/11/2008/10:30)
                       date-2:
Produce:PerformTask/Activity (6.3)
```

On November 1st, 2008, INDIVIDUAL_PERSON_102 *ends the start-up of the* GP1Z_TURBINE.

```
virt3.c6)    (COORD  virt3.c7  #virt3.c8  virt3.c9)
```

The second term of the CAUSE *relationship consists of a* COORD *binding occurrence.*

```
virt3.c7)    MOVE    SUBJ      INDIVIDUAL_PERSON_104: (GP1Z_COMPLEX)
                     OBJ       #virt3.c8
                     BENF      INDIVIDUAL_PERSON_102: (GP1Z_MAIN_CONTROL_ROOM)
                     MODAL     vhf_audio_transmitter
                     date-1:   1/11/2008/10:15
                     date-2:
Move:StructuredInformation (4.42)
```

INDIVIDUAL_PERSON_104 *sends to* INDIVIDUAL_PERSON_102 *the* virt3.c8 *message.*

```
virt3.c8)    PRODUCE   SUBJ      INDIVIDUAL_PERSON_104: (GP1Z_COMPLEX)
                       OBJ       detection_
                       TOPIC   (SPECIF lubrication_oil_leakage (SPECIF around_
                                         AUXILIARY_LUBRICATION_PUMP_M202))
                       date-1:   1/11/2008/10:02
                       date-2:   1/11/2008/10:15
Produce:PerformTask/Activity (6.3)
```

INDIVIDUAL_PERSON_104 *has discovered the presence of an oil leakage around the lubrication pump M202.*

```
virt3.c9)    OWN    SUBJ      AUXILIARY_LUBRICATION_PUMP_M202
                    OBJ       property_
                    TOPIC     (SPECIF coupled_with GP1Z_TURBINE)
                    { obs }
                    date-1:   1/11/2008/10:02
                    date-2:
Own:CompoundProperty (5.42)
```

On November 1st, 2008, at 10h02, we can observe that the lubrication pump is related to the GP1Z_TURBINE.

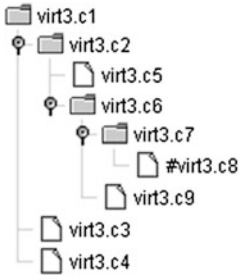
```
virt3.c3)    BEHAVE   SUBJ      INDIVIDUAL_PERSON_102: (GP1Z_MAIN_CONTROL_ROOM)
                      MODAL     production_activities_leader
                      { obs }
                      date-1:   1/11/2008/10:15
                      date-2:
Behave:Role (1.11)
```

We can remark that INDIVIDUAL_PERSON_102 *fulfils the function of production activities leader.*

```
virt3.c4)    BEHAVE   SUBJ      INDIVIDUAL_PERSON_104: (GP1Z_COMPLEX)
                      MODAL     field_operator
                      { obs }
                      date-1:   1/11/2008/10:15
                      date-2:
Behave:Role (1.11)
```

We can remark that INDIVIDUAL_PERSON_104 *fulfils the function of field operator at the GPIZ complex.*

Fig. 2 Tree structure corresponding to the narrative of Table 5



Eventually, we can note that the logical arrangement of a narrative (like that of Table 5) can always be represented as some sort of complex *tree structure*, see Fig. 2. This remark is not new, and can be considered as valid in general independently from the formalization adopted, see, e.g., the "story trees" of Mani and Pustejovsky (2004).

Querying/Inference Procedures

Reasoning in NKRL ranges from the *direct questioning* of a knowledge base of NKRL formal structures to the execution of *high-level inference procedures*. These issues have been dealt with in some detail in Zarri (2005, 2009, pp. 183–243, 2013). We will then limit us, here, to supply some essential information about these topics.

Search Patterns

Direct questioning of NKRL knowledge bases is implemented by means of search patterns p_j (formal queries) that unify information in the base thanks to the use of a Filtering Unification Module (*Fum*).

Formally, search patterns correspond to *specialized/partially instantiated* templates where the "*explicit variables*" that characterize the templates (var_i, see Table 1 above) have been replaced by concepts/individuals compatible with the constraints imposed on these variables in the original HTemp structures. In a search pattern, the concepts are used then as "*implicit variables*." When trying to unify a search pattern p_j, manually built up from the user or automatically created by an *InferenceEngine* (see below) with the predicative occurrences pc_i of the knowledge base, a p_j concept can match (1) the individuals included in pc_i that represent *its own instances*, and (2) all its pc_i *subsumed concepts* (according to the HClass' structure) *along with their own instances*. This, inheritance-based, way of operating corresponds then to a sort of semantic/conceptual expansion of the original pattern.

"Transformation" Inference Rules

A first class of NKRL high-level inference procedures is implemented through the use of the so-called *transformation* rules. These rules try to "adapt," from a semantic point of view, a search pattern p_j that "failed" (that was unable to find a unification within the knowledge base) to the real contents of this base making use of a sort of *analogical reasoning*. Transformations attempt then to automatically "*transform*" p_j into one or more different $p_1, p_2 \ \ldots \ p_n$ that are not strictly "*equivalent*" but only "*semantically close*" (analogical reasoning) to the original one.

A transformation rule is composed of a left-hand side, the "*antecedent,*" and of one or more right-hand sides, the "*consequent(s)*." The antecedent corresponds to the formulation, in search pattern format, of the "query" to be transformed, while the consequent(s) denote(s) the representation(s) of one or more search patterns to be substituted for the given one. Indicating with *A* the antecedent and with Cs_i all the possible consequents, these rules can be expressed as:

$$A\left(var_i\right) \Rightarrow Cs_i\left(var_j\right), \quad var_i \subseteq var_j \tag{3}$$

The restriction $var_i \subseteq var_j$ corresponds to the usual "*safety condition*" constraint that assures the logical congruence of the rules, stating that a transformation rule is *well-formed* when all the variables declared in the antecedent *A* appear also in the consequent Cs_i accompanied, in case, by additional variables.

Let us now see a concrete example: we want to ask whether, within the particular knowledge base where are stored all the NKRL-encoded events concerning the activation of a gas turbine, we can retrieve the information that a given oil extractor is running. In the absence of a direct answer we can reply by supplying, thanks to a rule like *t11* of Table 6, other related information stating, e.g., that the site leader has heard the working noise of the oil extractor. Expressed in natural language, this last result could be paraphrased as: "The system cannot assert that the oil extractor is running, but it can certify that the site leader has heard the working noise of this extractor."

From Table 6 we can note that the *atoms* of the NKRL rules are expressed using the *usual* NKRL knowledge representation tools, i.e., *as n-ary complex data structures centered on the notion of "functional role"* (Zarri 2011a). This implies the possibility to implement and manage *highly expressive* inference rules whose *atoms* can *directly represent* complex situations, actions, etc. In the context of the NKRL's rule system we are no more restricted, then, to the set-up of rules under the form of ordinary (and *scarcely expressive*) *binary clauses*. An exhaustive paper on this topic is Zarri (2013).

Table 6 An example of transformation rule

t11: *"working noise/condition" transformation*

antecedent:

OWN	SUBJ	*var1*
	OBJ	property_
	TOPIC	running_

var1 = consumer_electronics, hardware_, surgical_tool, diagnostic_tool/system, small_portable_equipment, technical/industrial_tool

first consequent schema (*conseq1*):

EXPERIENCE	SUBJ	*var2*
	OBJ	evidence_
	TOPIC	(SPECIF *var3 var1*)

var2 = individual_person
var3 = working_noise, working_condition

second consequent schema (*conseq2*):

BEHAVE	SUBJ	*var2*
	MODAL	industrial_site_operator

Being unable to demonstrate directly that an industrial apparatus is running, the fact that an operator hears its working noise or notes its working aspect can be considered as a proof of its running status.

"Hypothesis" Inference Rules

The "*hypothesis*" rules represent a second important class of NKRL inference rules. They allow us to build up automatically a sort of "*causal explanation*" for an elementary event (a predicative occurrence) retrieved by direct query within an NKRL knowledge base. These rules can be expressed as *biconditionals* of the type:

$$X \text{ iff } Y_1 \text{ and } Y_2 \ \ldots \text{ and } Y_n, \quad (4)$$

where the *head X* of the rule corresponds to a predicative occurrence pc_i to be "explained" and the *reasoning steps* Y_i must all be satisfied; Y_i are called "*condition schemata*" in a hypothesis context. This means that, for each of them, at least a search patterns p_j must be (*automatically* in this case) built up by *InferenceEngine* in order to find, using *Fum* (see section "Search Patterns"), a *successful* unification with some information of the base. In this case, the set of pc_1, pc_2 … pc_n predicative occurrences retrieved by the condition schemata Y_i, thanks to their conversion into p_j, can be interpreted as a context/causal explanation of the original occurrence pc_i (X). A generalization of the safety condition introduced above is used in a hypothesis rules context.

To mention now a well-known NKRL example, let us suppose we have directly retrieved, in a querying-answering mode, the information: "Pharmacopeia, a USA biotechnology company, has received 64,000,000 dollars from the German company Schering in the context of its R&D activities"; this information corresponds then to pc_i (X). We can then be able to automatically construct, using a "hypothesis" rule, a sort of "*causal explanation*" for this event by retrieving in the knowledge base information like: (1) "Pharmacopeia and Schering have signed an agreement concerning the production by Pharmacopeia of a new compound," pc_1 (Y_1) and (2) "in the framework of this agreement, Pharmacopeia has actually produced the new compound," pc_2 (Y_2). Of course, as usual in a "hypothesis" context, the explication proposed by this rule *corresponds to only one of all the possible reasons that can be interpreted as the "cause" of the original event*. A particular hypothesis rule must always be conceived as a member of a "family" of possible explication statements.

Note, moreover, that an interesting feature of the NKRL rule system concerns the possibility of making use of "transformations" when working in a "hypothesis" context—i.e., of utilizing these two modalities of inference in an "*integrated*" way. This means in practice that, whenever a search pattern p_j is derived from a condition schema Y_i of a hypothesis to implement a step of the reasoning process, we can use this pattern as it has been *automatically built up by InferenceEngine from its "father" condition schema*, but also in a "*transformed*" form if the appropriate transformation rules exist. In this way, a hypothesis that was deemed to fail because of the impossibility of deriving a "successful" p_j from one of its condition schemata Y_i can now continue if a new p_j, *obtained using a transformation rule*, will find a successful unification within the base, getting then new values for the hypothesis variables. Moreover, this strategy can also be used to discover all the possible *implicit* relationships among the stored data, see Zarri (2005) for further details.

Related Work

In this section, we will mention some approaches to the solution of the *"connectivity phenomena"* problems that have been suggested in an Artificial Intelligence (AI) and Computational Linguistics (CL) context and that have some relationships with the NKRL's procedures described in section "Linking Elementary Events".

Proposals Derived from an "Artificial Intelligence" Context

An *n*-ary knowledge representation model used to encode *narrative-like structures* that was very popular in the seventies is the "Conceptual Dependency" theory of Roger Schank (Schank 1973, Schank and Abelson 1977). In this, the underlying meaning ("*conceptualization*") of a given narrative was expressed as the association of semantic predicates chosen from a set of twelve formal "*primitive actions*" (like INGEST, MOVE, ATRANS, abstract relationship transfer, PTRANS, physical transfer, etc.) with seven *role relationships* ("deep cases") in the Case Grammar style (Fillmore 1968). The seven roles were Object (in a state), Object (of a change of state), Object (of an action), Actor, Recipient/Donor, From/To, and Instrument. Unfortunately, Schank's theory was, on the one hand, *insufficiently specified* and, on the other, *unnecessarily complicated* because of the influence of "psychological" (introspective) considerations according to a characteristic trend of AI in those years. Nevertheless, Schank's work had a particularly strong influence on the development of formalized (and at least partly computerized) systems for the representation and management of *storylines and connectivity phenomena* making use of all sorts of scripts, scenarios, TAUs (Thematic Abstraction Units), MOPs (Memory Organization Packets), etc., see, e.g., (Dyer 1983; Kolodner 1984).

The SnePS (Semantic Network Processing System) proposal of Stuart Shapiro (1979) belongs roughly to the same period and allows us, e.g., to annotate "*narrative*" situations like "Sue thinks that Bob believes that a dog is eating a bone" *by associating labeled graphs* in a way not too different from the NKRL's "completive construction" approach, see section "Completive Construction". Interestingly, solutions of this type have been re-discovered recently in the framework of the "Interoperable Knowledge Representation for Intelligence Support" (IKRIS) project, financed between in 2005–2006 by the US DTO (Disruptive Technology Office). IKRIS' main result is represented by the specifications of IKL, the "IKRIS Knowledge Language" (Hayes 2006; Hayes and Menzel 2006). IKL is an extension of Common Logic (ISO 2007) that, although still dealing with, fundamentally, first-order logic structures, includes *some HOL improvements in the NKRL style*. For example, IKL's formal structures called "*proposition name*" and introduced by the reserved symbol "that" allows us to "reify" the content of a sentence that can then be freely referred to from inside different contexts—the similarity with the *completive construction approach* is then evident. Going back in time to the fifties-sixties we can also note that, among the "*correlators*" introduced by Silvio Ceccato in a Mechanical Translation (MT) context *to represent "narratives" as recursive networks of triadic structures* (Ceccato 1961, 1964), some concerned the representation of "*connectivity phenomena*" like coordination and subordination, apposition, subject–predicate relationships, etc.

Among the recent suggestions for representing phenomena of this kind, we must discuss in particular some mechanisms used in a *Conceptual Graph's environment* for dealing with "*contexts*." John Sowa's Conceptual Graphs (CGs), see (Sowa 1984, 1999), are based on a powerful graph-based representation scheme. A conceptual graph is a finite, connected, bipartite graph that makes use of two kinds of nodes, "*concepts*" and "*conceptual relations*" (these last corresponding to NKRL's functional roles). For example, a CG corresponding to the narrative "John is going to Boston by bus" is represented by a conceptual structure where a "*concept node,*" "Go" (having a function similar to that of an NKRL conceptual predicate, but denoted by an NL term) is associated with three "*relation nodes*" (roles) like Ag(e)nt, Dest(ination), and Instr(ument). These relations introduce the three *arguments of the predicate*, i.e., three new concept nodes representing, respectively, the *constant* John (the "agent") as an instance of the *concept* Person, the *constant* Boston (the "destination") as an instance of the *concept* City, and the *concept* Bus (the "instrument"). The resemblance to HTemp and to the NKRL representation of elementary events is evident. Moreover, for any CGs system, it is assumed that there exists a *pre-defined type hierarchy of "concept-types,"* different according to the domain to formalize and similar then to HClass.

Contexts in CGs are dealt with making use of the second order (nested graphs) extensions that bear some resemblance to NKRL's constructs like completive construction and binding occurrences, as we can see from Sowa's analysis (1999, pp. 485–486) of the complex narrative "Tom believes that Mary wants to marry a sailor." This last is decomposed, as in NKRL, in two parts. In a first one, "Tom believes that . . . " Tom is modeled as an "experiencer" (Expr role) that Believe(s) a "proposition" (an OBJ(ect) filler according to the NKRL's formalism). The second part corresponds to the representation of the proposition/filler " . . . Mary wants to marry . . . ", where the two elementary events signaled by the presence of the two predicates "want" and "marry" are linked together by the fact that the "situation" corresponding to the marriage is the Th(e)me of Mary's wishes. Other similarities between CGs and NKRL concern some specific algorithmic aspects of the querying/inference procedures, see, e.g., (Ellis 1995, Corbett 2003).

We can also find, however, some important differences between the NKRL and the CGs approaches. They are related, e.g., to the organization of the "standard" hierarchy of concepts (quite simple in a CGs context with respect to the sophistication of the HClass hierarchy in NKRL), the choice of the deep cases (functional roles in NKRL terms) or the *general theoretical background* proper to the inference procedures. But the central point of any discussion about the relationships between CGs and NKRL concerns John Sowa's choice of leaving *completely free*, for the sake of generality, the selection of those "*predicates*" that, in CGs as in NKRL, represents the focal element of the formal representation of an elementary event. In the CGs representation of the "John is going to Boston . . . " event, see above, the predicate can then be *simply represented by the surface element "Go"*—it would be a primitive like MOVE in NKRL. Note that Sowa emphasizes (1984, p. 14) that a CGs' predicate can be either a primitive or an NL term, but it is normally the second (simpler) solution that is chosen. As a consequence, *it is practically impossible to create an exhaustive and authoritative list of CGs "canonical graphs,"* roughly equivalent to NKRL's "templates," *for evident reasons of combinatorial explosion* (e.g., 3100 English verbs are included in the well-known Levin's classification (1993), which is notoriously incomplete). A tool like HTemp—extremely important for the set-up of concrete NKRL applications and, in practice, part and parcel of the definition of the NKRL language—is not really conceivable, then, in a CGs context.

Other general knowledge representation systems that share with CGs some ambitions of "universality" are CYC (Lenat and Guha 1990; Lenat et al. 1990) and Topic Maps (Pepper 2000).

CYC concerns one of the most controversial endeavors in the history of Artificial Intelligence. Started in the early 1980, the project ended about 15 years later with the set-up of an enormous knowledge base containing *about a million of hand-*

entered "logical assertions" including both simple statements of facts and rules about what conclusions could be inferred if certain statements of facts were satisfied. The "upper level" of the CYC ontology is now freely accessible on the Web, see http://www.cyc.com/cyc/opencyc. A criticism often addressed to CycL—the *n*-ary knowledge representation language of CYC—concerns its uniform use of the same representation model (substantially, a frame system rewritten in logical form) to represent phenomena *conceptually very different* (the "*one and only syndrome*"). In NKRL, on the contrary, concepts are represented in the (usual) binary way, elementary events/situations (and general classes of events/situations) like *n*-ary predicate/roles-based structures, connectivity phenomena as labelled lists with reified arguments, special conceptual structures have been conceived to take the temporal phenomena into account, etc.

With respect now to Topic Maps (TMs), a TMs "*topic*" is used to represent *any possible specific notion that could be interesting to speak about*, like the play Hamlet, the playwright William Shakespeare, or the "authorship" relationship. A topic *reifies* then a subject, making it "real" for a computer system. Topics can have "*names,*" and each individual topic is an instance of one or more classes of topics ("*topic types*"). They can also have "*occurrences,*" that is, information resources, specified as a text string that is part of the Topic Map itself, or as a link to an external resource, which are considered as relevant to the subjects the topic reify. Topics can participate in relationships with other topics, called "*associations*": an association consists in a number of "*association roles*" each of which has a topic attached as a "*role player*." In spite of the introduction of *associations*, Topic Maps do not seem to present really new insights into the connectivity phenomena issues. More in general, it must be noticed that TMs have been often considered as a downgraded version of other (more structured and powerful) conceptual proposals, like Semantic Networks (Lehmann 1992), Conceptual Graphs, or NKRL.

Eventually, we can note that the now popular "Semantic Web" (SW) tools and languages (see http://semanticweb.org/wiki/Tools for an overview) *cannot represent* a viable alternative for an effective management of the "connectivity phenomena" and the related, *high-level knowledge representation problems*. This is linked to the difficulties that concern the set-up of *complete and effective formal description of complex information structures* like spatio-temporal data, contexts, reified situations, human intentions and behaviors, etc. making use only of the *quite limited "binary" knowledge representation tools* proper to the SW languages. As already stated, properties in the binary model are simply expressed, in fact, as a *binary (i.e., accepting only two arguments) relationship linking two individuals or an individual and a value*. The resulting, well-known lack of expressiveness of the SW languages is described, to give only few examples, in papers like Mizoguchi et al. (2007), Salguero et al. (2009), and Liu et al. (2010). Dealing with the above high-level representation problems requires, on the contrary, to make use of *high-level knowledge representation tools* in the NKRL, CGs, CycL, etc. style, able then too take care of *higher arity relations*. Note also, in this context, some perplexities in the knowledge representation milieus about recent proposals of SW origin suggesting to deal with all sort of very complex problems—from the representation of temporal data (Scherp et al. 2009) to the modelling of the user context—by exploiting *fragments of existing SW ontologies* under the form of "*Ontology Design Patterns*" (ODPs) that, at least in principle, we could freely compose, specialize, and reutilize. Unfortunately, existing ODPs—see, e.g., those collected within the ODP portal (http://ontologydesignpatterns.org/wiki/Ontology_Design_Patterns_._org_%28ODP%29)—are characterized by a *high level of heterogeneity and the lack of shared theoretical principles for their construction and use*. They are not without evoking, then, those "*idiosyncratic patterns*" whose development, according to Kozaki et al. (2007), can lead "... to a decrease of the semantic interoperability of ontologies because ... such patterns will lack compatibility with others."

Proposals Derived from a "Linguistics/Computational Linguistics" Context

Looking now at a broad Computational Linguistics/Natural Language Processing (NLP) context, we can note immediately that *interesting similarities* can be found between the use of NKRL for the *modelling of the inner meaning of narrative documents* and the use of tools as VerbNet, PropBank, and FrameNet for the *surface semantic/thematic role labeling of NL texts* in a "post-case grammars" framework (Palmer et al. 2010).

However, the main objectives of any possible kind of NLP procedures concern, firstly, the implementation of *linguistically motivated*, surface analyses of NL documents aiming at discovering *syntactic/semantic relationships between NL items expressed in a specific language*. Therefore, these objectives coincide only in part with those concerning the execution of *deep "conceptual" procedures* in an NKRL' style. Look, e.g., at the NKRL's "*functional roles*": even if they are labelled with terms borrowed from research on case grammar (Fillmore 1968) and thematic roles, they are in fact "*deep cases,*" used to link together "*conceptual entities*" (concepts, concept instances, semantic predicates, spatio-temporal abstractions, etc.) instead of "*words*." Pure *surface phenomena* like the idiosyncrasies in the lexical choices, the active/passive alternation, the morphology, etc. *are then totally ignored*. This means that the *formal expressions* dealt with in an NKRL context *are independent from any particular natural language formulation*—even if they are drafted in a sort of "basic English" for human understanding in their "external" formulation, a choice shared with other conceptual approaches like CGs, Schank's Conceptual Dependency, etc.—and that NKRL follows then a sort of "*interlingua*" (i.e., language independent) approach. All the above can be summed up by saying that NKRL, as all the formal systems mentioned in the previous sub-section, addresses the problem of encoding the "*meaning*" of generic (not only NL) multimedia documents through the development of an *a priori* formal notation for expressing conceptual contents that is *independent* from the search for an optimal *form of correspondence*—see Jackendoff's (1990) "θ-Criterion"—with the "*surface*" (syntactic) form these contents can assume. Obviously, in an NKRL, etc. approach, the *correspondence problem still exists*, but can be tackled *a posteriori* in a very pragmatic way, see, e.g., Note 3 above.

We briefly mention below, nevertheless, some linguistic/NLP systems/theories that can be considered as *particularly significant* from a *"semantic/conceptual"* point of view.

Episodic Logic (EL) (Schubert and Hwang 2000) is an *NL-like*, highly formalized logical representation for narrative understanding allowing, among other things, the expression of sentence and predicate reification, of intensional predicates (corresponding to wanting, believing, making, etc.) of episodes, events, states of affairs, etc. "Episodes" can be *explicitly related* in terms of part-whole, temporal, and causal relations. Interesting solutions for the connectivity phenomena management can also be found in the *Discourse Representation Theory*, DRT (Kamp and Reyle 1993), a semantic theory developed for representing and computing *trans-sentential anaphora and other forms of text cohesion*. See, for example, the specific DRT procedures—that make use, among other things, of "embedding functions" similar, at the surface level, to the context solutions proposed by Sowa, etc., see above—that have been suggested for managing all sort of context-related problems. The *Text Meaning Representation* model (TRM) is part of the OntoSem environment (Nirenburg and Raskin 2004). It consists of an (at least partially) implemented theory of *natural language processing* that aims at automatically deriving *structured meaning* (in TMR terms) *from unstructured texts*. The central piece of TMR is a language-independent *single* ontology structured as a DAG (Direct Acyclic Graph) where the arcs represent IsA relationships. The ontology includes about 8500 concepts represented according to a plain frame-like format. Detailed analyses of the advantages and weaknesses of TRM are presented in Sowa (2005) and Zarri (2009, pp. 146–149).

Conclusion

This chapter focuses on the problem of finding a *complete and coherent way* of representing the "*global meaning*" of complex (multimedia) "*narratives*" by properly associating its constituent basic entities represented as a set of formalized "*elementary events*." Solving this problem means, in practice, being able to formalize those "*connectivity phenomena*"—denoted, at *linguistic surface level*, by logico-semantic cohesion links like causality, goal, co-ordination, subordination, indirect speech, etc.—that assure the conceptual unity of the narratives, scenarios, situations, etc. Note that the problem of finding *reasonable solutions* for dealing with this sort of phenomena is, at the same time, far from being trivial from a Computer Science point of view (see also the "State of the Art" in the previous section) and of a strong interest from a general *Cognitive Science/Digital Humanities* perspective. It is part, in fact, of a wider problem that concerns finding *reasonable solutions* for dealing with "*contexts*"; representing contexts in full is still a largely unsettled problem. See

McCarthy (1993) for the most cited formal theory about representing contexts as abstract mathematical, first class objects,[4] a theory that goes back to more than 20 years ago.

Specifically, the solutions adopted by NKRL, "*completive construction*" and "*binding occurrences,*" allow us to model the connectivity phenomena by "*reifying*" the formal representations associated with the constitutive elementary events; these solutions have been explained making use of several illustrating examples. In particular, the NKRL representation of a complex, structured narrative that involves the occurrence of several elementary events has been presented in full and commented in some detail.

We can conclude the chapter by noticing that, apart from being a knowledge representation language, NKRL is also a *fully operational computer science environment*, implemented in Java and built up, thanks, at least partly, to the support of several European projects; a detailed description of the NKRL software can be found in Zarri (2009: Appendix A). The environment exists in two versions, a (standard) SQL-based version and a (lighter) file-oriented one, to be used mainly as a "demonstration" version. The environment includes also powerful "inference engines" able to carry out complex inference procedures based, e.g., on "analogical" and "causal" reasoning principles, see again Zarri (2005, 2013) in this context.

References

N. Ayari, A. Chibani, Y. Amirat, in *Proceedings of the 2013 IEEE International Conference on Robotics and Automation (ICRA)*. Semantic Management of Human-Robot Interaction in Ambient Intelligence Environments Using N-Ary Ontologies (IEEEXplore, Piscataway, 2013), pp. 1164–1171

M. Bal, *Narratology: Introduction to the Theory of Narrative*, 2nd edn. (University Press, Toronto, 1997)

S. Bechhofer, F. van Harmelen, J. Hendler, I. Horrocks, D.L. McGuinness, P.F. Patel-Schneider, L.A. Stein (eds.), *OWL Web Ontology Language Reference* (W3C Recommendation 10 February 2004). W3C (2004), http://www.w3.org/TR/owl-ref/. Accessed 28 Feb 2016

S. Ceccato (ed.), *Linguistic Analysis and Programming for Mechanical Translation* (Technical Report RADC-TR-60-18) (Feltrinelli, Milano, 1961)

S. Ceccato, Automatic translation of languages. Inf. Storage Retr. **2**, 105–158 (1964)

D. Corbett, *Reasoning and Unification Over Conceptual Graphs* (Kluwer Academic/Plenum Publishers, New York, 2003)

M.G. Dyer, *In-Depth Understanding* (The MIT Press, Cambridge, 1983)

G. Ellis, Compiling conceptual graph. IEEE Trans. Knowl. Data Eng. **7**, 68–81 (1995)

[4]In McCarthy's theory, the main formulas are sentences of the form **ist**(*c*, *p*), which are to be taken as assertions that the proposition *p is true in (ist) the context c*, itself asserted in an outer context *c'*. A well-known concrete implementation of McCarthy's theory is represented by the "microtheories," introduced by Ramanathan V. Guha and largely used in a CYC framework (Guha and Lenat 1994); a recent re-interpretation of McCarthy's ideas in Description Logics terms is (Klarman and Gutiérrez-Basulto 2011).

C.J. Fillmore, in *Universals in Linguistic Theory*, ed. by E. Bach, R.T. Harms. The Case for Case (Holt, Rinehart and Winston, New York, 1968), pp. 1–88

R.V. Guha, D.B. Lenat, Enabling agents to work together. Commun. ACM **37**(7), 127–142 (1994)

M.A.K. Halliday, R. Hasan, *Cohesion in English* (Longman, London, 1976)

P. Hayes, *IKL Guide* (Florida Institute for Human & Machine Cognition (IHMC), Pensacola, 2006), http://www.ihmc.us/users/phayes/IKL/GUIDE/GUIDE.html. Accessed 28 Feb 2016

P. Hayes, C. Menzel, *IKL Specification Document* (Florida Institute for Human & Machine Cognition (IHMC), Pensacola, 2006), http://www.ihmc.us/users/phayes/IKL/SPEC/SPEC.html. Accessed 28 Feb 2016

Z. Hu, E. Rahimtoroghi, L. Munishkina, R. Swanson, M.A. Walker, in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP '13)*, ed. by T. Baldwin, A. Korhonen. Unsupervised Induction of Contingent Event Pairs from Film Scenes (ACL, Stroudsburg, 2013), pp. 369–379

International Organization for Standardization, ISO, *Information Technology-Common Logic (CL): A Framework for a Family of Logic-based Languages* (ISO/IEC 24707:2007) (ISO, Geneva, 2007)

R. Jackendoff, *Semantic Structures* (The MIT Press, Cambridge, 1990)

M. Jahn, *Narratology: A Guide to the Theory of Narrative* (version 1.8) (English Department of the University, Cologne, 2005), http://www.uni-koeln.de/~ame02/pppn.htm. Accessed 28 Feb 2016

H. Kamp, U. Reyle, *From Discourse to Logic. Introduction to Modeltheoretic Semantics of Natural Language, Formal Logic and Discourse Representation* (Kluwer, Dordrecht, 1993)

S. Klarman, V. Gutiérrez-Basulto, in *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence*, ed. by W. Burgard, D. Roth. Two-Dimensional Description Logics for Context-Based Semantic Interoperability (AAAI Press, Menlo Park, 2011), pp. 215–220

D. Klein, C.D. Manning, in *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, ed. by E. W. Hinrichs, D. Roth. Accurate Unlexicalized Parsing (ACL, Stroudsburg, 2003), pp. 423–430

J.L. Kolodner, *Retrieval and Organizational Strategies in Conceptual Memory: A Computer Model* (Lawrence Erlbaum Associates, Hillsdale, 1984)

K. Kozaki, E. Sunagawa, Y. Kitamura, R. Mizoguchi, in *Proceedings of the 2nd Workshop on Roles and Relationships in Object Oriented Programming, Multiagent Systems and Ontologies, co-located with ECOOP 2007 (Technical Report 2007-9)*, ed. by G. Boella, S. Goebel, F. Steimann, S. Zschaler, M. Cebulla. Role Representation Model Using OWL and SWRL (Technische Universität Berlin, Berlin, 2007), pp. 39–46

F. Lehmann (ed.), *Semantic Networks in Artificial Intelligence* (Pergamon Press, Oxford, 1992)

D.B. Lenat, R.V. Guha, *Building Large Knowledge Based Systems* (Addison-Wesley, Reading, 1990)

D.B. Lenat, R.V. Guha, K. Pittman, D. Pratt, M. Shepherd, CYC: toward programs with common sense. Commun. ACM **33**(8), 30–49 (1990)

B. Levin, *English Verb Classes and Alternation, A Preliminary Investigation* (University Press, Chicago, 1993)

W. Liu, Z. Liu, J. Fu, R. Hu, Z. Zhong, in *Proceedings of the 2010 International Conference on Complex, Intelligent and Software Intensive Systems*, ed. by L. Barolli, F. Xhafa, S. Vitabile, H.-H. Hsu. Extending OWL for Modeling Event-Oriented Ontology (IEEE Computer Society Press, Los Alamitos, 2010), pp. 581–586

I. Mani, J. Pustejovsky, in *Proceedings of the ACL Workshop on Discourse Annotation*, ed. by B. Webber, D. Byron. Temporal Discourse Models for Narrative Structure (Stroudsburg, ACL, 2004), pp. 57–64

S. Matsuyoshi, M. Eguchi, C. Sao, K. Murakami, K. Inui, Y. Matsumoto, in *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2010*, ed. by N. Calzolari, K. Choukri, B. Maegaard, J. Mariani, J. Odijk, S. Piperidis, M. Rosner, D. Tapias. Annotating Event Mentions in Text with Modality, Focus, and Source Information (European Language Resources Association (ELRA), Paris, 2010), pp. 1456–1463

J. McCarthy, in *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence – IJCAI/93*, ed. by R. Bajcsy. Notes on Formalizing Context (Morgan Kaufmann, San Francisco, 1993), pp. 555–562

R. Mizoguchi, E. Sunagawa, K. Kozaki, Y. Kitamura, A model of roles within an ontology development tool: Hozo. J. Appl. Ontol. **2**, 159–179 (2007)

Morris, J. and G. Hirst, G. (1991). Lexical cohesion computed by Thesaural relations as an indicator of the structure of text. Comput. Linguist. 17: 21-48.

S. Nirenburg, V. Raskin, *Ontological Semantics* (The MIT Press, Cambridge, 2004)

N.F. Noy, R.W. Fergerson, M.A. Musen, in *Knowledge Acquisition, Modeling, and Management – Proceedings of EKAW 2000*, ed. by R. Dieng, O. Corby. The Knowledge Model of Protégé-2000: Combining Interoperability and Flexibility, vol 1937 (Springer LNCS, Berlin, 2000), pp. 17–32

M. Palmer, G. Gildea, N. Xue, *Semantic Role Labeling* (Morgan and Claypool Publishers, San Rafael, 2010)

S. Pepper, *The TAO of Topic Maps: Finding the Way in the Age of Infoglut* (Ontopia AS, Oslo, 2000), http://www.ontopia.net/topicmaps/materials/tao.html. Accessed 28 Feb 2016

A.G. Salguero, C. Delgado, F. Araque, in *Computer Aided Systems Theory, 12th International Conference, EUROCAST 2009*, ed. by R. Moreno Díaz, F. Pichler, A. Quesada Arencibia. Easing the Definition of N-Ary Relations for Supporting Spatio-Temporal Models in OWL, vol 5717 (Springer LNCS, Berlin, 2009), pp. 271–278

R.C. Schank, in *Computer Models of Thought and Language*, ed. by R. C. Schank, K. M. Colby. Identification of Conceptualizations Underlying Natural Language (W.H. Freeman and Co., San Francisco, 1973), pp. 187–247

R.C. Schank, R.P. Abelson, *Scripts, Plans, Goals and Understanding: An Inquiry into Human Knowledge Structures* (Lawrence Erlbaum, Oxford, 1977)

A. Scherp, T. Franz, C. Saathoff, S. Staab, in *Proceedings of the Fifth International Conference on Knowledge Capture, K-CAP '09*, ed. by Y. Gil, N. Noy. F – A Model of Events Based on the Foundational Ontology DOLCE+DnS Ultralite (ACM, New York, 2009), pp. 137–144

L.K. Schubert, C.H. Hwang, in *Natural Language Processing and Knowledge Representation: Language for Knowledge and Knowledge for Language*, ed. by L. Iwanska, S. C. Shapiro. Episodic Logic Meets Little Red Riding Hood: A Comprehensive, Natural Representation for Language Understanding (MIT/AAAI Press, Cambridge & Menlo Park, 2000), pp. 111–174

S.C. Shapiro, in *Associative Networks: Representation and Use of Knowledge by Computers*, ed. by N. V. Findler. The SNePS Semantic Network Processing System (Academic Press, New York, 1979), pp. 179–203

J.F. Sowa, *Conceptual Structures: Information Processing in Mind and Machine* (Addison-Wesley, Reading, 1984)

J.F. Sowa, *Knowledge Representation: Logical, Philosophical, and Computational Foundations* (Brooks Cole Publishing Co., Pacific Grove, 1999)

J.F. Sowa, Review of "Computational Semantics" by Sergei Niremburg and Victor Raskin. Comput. Linguist. **31**, 147–152 (2005)

W3C OWL Working Group (eds.), *OWL 2 Web Ontology Language Document Overview*, 2nd edn (W3C Recommendation 11 December 2012). W3C (2012), http://www.w3.org/TR/owl2-overview/. Accessed 28 Feb 2016

G.P. Zarri, in *Proceedings of the First International Conference on Applied Natural Language Processing*, ed. by I. M. Kameny, B. T. Oshika. Automatic Representation of the Semantic Relationships Corresponding to a French Surface Expression (ACL, Stroudsburg, 1983), pp. 143–147

G.P. Zarri, Integrating the two main inference modes of NKRL, transformations and hypotheses. J. Data Semant. (JoDS) **4**, 304–340 (2005)

G.P. Zarri, *Representation and Management of Narrative Information, Theoretical Principles and Implementation* (Springer, London, 2009)

G.P. Zarri, in *Computational Models of Narratives – Papers from the AAAI 2010 Fall Symposium (Technical Report FS-10-04)*, ed. by M. A. Finlayson, P. Gervás, E. Mueller, S. Narayanan, P. Winston. Representing and Managing Narratives in a Computer-Suitable Form (AAAI Press, Menlo Park, 2010), pp. 73–80

G.P. Zarri, in *Proceedings of the 24th International Florida AI Research Society Conference, FLAIRS-24*, ed. by R. C. Murray, P. M. McCarthy. Differentiating Between "Functional" and "Semantic" Roles in a High-Level Conceptual Data Modeling Language (AAAI Press, Menlo Park, 2011a), pp. 75–80

G.P. Zarri, Knowledge representation and inference techniques to improve the management of gas and oil facilities. Knowl.-Based Syst. (KNOSYS) **24**, 989–1003 (2011b)

G.P. Zarri, Advanced computational reasoning based on the NKRL conceptual model. Expert Syst. Appl. (ESWA) **40**, 2872–2888 (2013)

G.P. Zarri, in *Special Issue on Affective Neural Networks and Cognitive Learning Systems for Big Data Analysis*, ed. by A. Hussain, E. Cambria, B. Schuller, N. Howard. Sentiments Analysis at Conceptual Level Making Use of the Narrative Knowledge Representation Language, vol 58 (Neural Networks (NEUNET), 2014), pp. 82–97

G.P. Zarri, in *Proceedings of the Twenty-Eighth International Florida Artificial Intelligence Research Society Conference, FLAIRS-28*, ed. by I. Russell, W. Eberle. The "Qua-Entities" Paradigm versus the Notion of "Role" in NKRL (Narrative Knowledge Representation Language) (AAAI Press, Menlo Park, 2015), pp. 97–102

Springer

http://www.springer.com/978-3-319-54498-4

Data Analytics in Digital Humanities
Hai-Jew, S. (Ed.)
2017, XXII, 295 p. 37 illus., 23 illus. in color., Hardcover
ISBN: 978-3-319-54498-4