Concepts, Attributes, and Arbitrary Relations
Some Linguistic and Ontological Criteria for Structuring Knowledge Bases

Nicola Guarino
Italian National Research Council,
Institute for Systems Dynamics and Bioengineering (LADSEB-CNR),
Corso Stati Uniti 4, I-35020 Padova, Italy
phone: +39 49 8295751, fax: +39 49 8295649, email: guarino@ladseb.pd.cnr.it

ABSTRACT

There is a subtle risk of ambiguity in the choice between concepts and roles forced by current KL-ONE-like languages, since many roles may be concepts as well. In this paper we explore the ontological foundations of the role/concept relationship, and analyze its implications on the practice of knowledge engineering. We criticize the current interpretation of KL-ONE roles as *arbitrary* relations, which vanishes their original meaning and makes them identical to slots. We suggest to call *attributes* those *concepts* which actually act as conceptual components, and propose a formal semantics which binds these concepts to their corresponding relations.

Keywords: knowledge representation, terminological logics, knowledge engineering, knowledge acquisition, formal ontology, KL-ONE

Final Draft

To appear on *Data and Knowledge Engineering* 8 (1992)

Concepts, Attributes, and Arbitrary Relations

Some Linguistic and Ontological Criteria for Structuring Knowledge Bases

Nicola Guarino
Italian National Research Council,
Institute for Systems Dynamics and Bioengineering (LADSEB-CNR),
Corso Stati Uniti 4, I-35020 Padova, Italy

1 Introduction

In his seminal paper "On the epistemological status of semantic networks" Ron Brachman gave the following definition of a role [4]:

> The roles represent the various kinds of attributes, parts, etc, that things in the world are considered to "have". These include, for example, such things as parts (e.g., fingers of a hand), inherent attributes of objects and substances (e.g. color), arguments of functions (e.g. multiplier and multiplicand of a multiplication), and "cases" of verbs in sentences (e.g. "agent"). Any generalized attribute of this sort has two important pieces (1) the particular entity that becomes the value for the attribute in an instance of the Concept, and (2) the functional role which that entity fills in the conceptual complex. A Role is a formal entity that captures both of these aspects in a structured way, by packaging up information about both the role filler and the functional role itself.

Some years later, this definition changed slightly ([6], our italics):

> (...) the Role is the primary *component* of a Concept. A Role acts like a generalized attribute description, representing potential relationships between individuals of the type denoted by the Concept and other individuals. In other words, Roles are the KL-ONE *equivalent of two-place predicates*.

Indeed, in current KL-ONE-style languages like CLASSIC [2,5], LOOM [15], BACK [17] the semantics of a role is that of an *arbitrary* binary relation. John Sowa, however, introduces the notion of role in a different context [21]:

> Subtypes of ENTITY are of two kinds: *natural types,* which have no required set of linguistic associations; and *role types,* which are subtypes of natural types in some particular pattern of relationships. PERSON, for example, is a natural type, and TEACHER is a subtype of PERSON in the role of teaching.

The difference is basically in the fact that a role, for Sowa, captures a *particular pattern of* *relationships,* and not a single (arbitrary) relation as for Brachman. According to this view, roles are *associated* to relationships, but are concepts, not relations. We may observe therefore that Sowa's roles have almost nothing to do with KL-ONE roles; rather, we think that this comparison sheds some light on the cognitive meaning of roles as important knowledge representation primitives: there is a subtle, yet intimate connection between roles-as-concepts and roles-as-relations. An evident sign of this connection is the fact that many commonly used KL-ONE role names are actually common nouns like *son* or *age*, which may also be used as names of concepts.

Yet, current systems force us to make a radical choice between roles and concepts, which has a great influence on the knowledge engineering process of building a knowledge base. Once we have split our domain into roles and concepts, we have only two possible choices. The first is to definitely give up the possibility to capture the intrinsic meaning of our roles, using them just as "flat" relations. Paradoxically, some roles have a meaning which is more defined than that of the concepts they belong to: as noticed by Wilensky [23], *age* may have a well-defined meaning, more so than does *person*. The second possibility is to create a number of concepts which roughly parallel role names: for instance, *has-age/age*. Some systems like NIKL [16] encourage this practice (which has been recently re-formulated by Brachman and colleagues [5]) by allowing a separate hierarchy of roles. As noticed again by Wilensky, although this choice may improve the overall expressive adequacy, it leads to a duplication which does not account in a systematic way of the a-priori relationship existing between roles and concepts, and leaves the burden of its maintenance to the user. The dangerous effects of this duplication have been noticed for instance by Haimowitz [13].

A possibility to account in some way for the role/concept connection has been explored in the early KL-ONE networks, with the proposal of the so-called "QUA link" [9]. Its purpose was to define a concept in terms of the RoleSet of another concept, but its semantics was never completely clarified, and the construct has been now abandoned.

In the rest of this paper we shall try to explore the ontological foundations of the role/concept relationship, and to analyze its implications on the current practice of knowledge engineering. In the next section we propose to use the term "role" only in Sowa's sense; bearing on Husserl's theory of foundation [18], we distinguish between roles and natural concepts, and define a role as a concept which implies some particular "pattern of relationships", but does not necessarily act as a conceptual component of something. In section 3 we define *attributes* as concepts having an associate relational interpretation, allowing them to act as conceptual components as well as concepts on their own; we propose a formal semantics which binds these concepts to their corresponding relations, and a linguistic criterion to distinguish attributes from *slots,* i.e. from those relations which cannot be considered as conceptual components. Moreover, we show how the choice of considering attributes as concepts enforces *discipline* in conceptual analysis as well as *uniformity* in knowledge representation.

In section 4 we show that the set of roles and that of attributes, as defined above, do actually have a large intersection, represented by all those concepts like *mother* which, besides implying a pattern of relationships, act also as conceptual components. However, there are at the same time examples of attributes which cannot easily be called roles: they are for instance *qualities* like *color* and *part-names* like *wheel*. We propose a basic ontology of attributes, which uses the ontological notions of *foundedness* and *semantic rigidity* as classification features.

Finally, in the discussion we propose a possible methodology to decide whether a given name may actually be used as attribute of a certain object, which integrates the linguistic and ontological considerations made in the paper.

2 Roles and concepts

Let us try first to propose a standard definition for the term "role". We take Sowa's interpretation as a starting point, since it seems to fit well with the general meaning of the term. For instance, Websters' International Dictionary reports the following meanings for the word "role":

- a character assigned to or assumed by someone
- a socially prescribed pattern of behaviour corresponding to an individual's status in a particular society
- a part played by an actor
- a function performed by someone or something in a particular situation, process, or operation.

The proper meaning of a role is therefore that of a *concept*, whose instances are involved in "some particular pattern of relationships". Sowa proposes a simple test to decide whether a concept is a role [21]: *τ is a role type if something can only be identified as type τ by considering some other entity, action, or state*. This test is however too vague to capture our intended meaning. For instance, a car may be classified as a role since in order to recognize something as a car somebody has to check whether it has (at least) an engine and some wheels.

2.1 The notion of foundation

In order to be able to propose a more adequate condition, we have first to introduce the notion of *foundation* among concepts, typical of Husserl's ontology [20]. Foundation has been formalized in [18] by the modal formula below, where ε is the membership relation[1], $\leq$ the part-of relation and **nec** stands for the modal necessity (box) operator:

Definition 2.1 The concept α is *founded on* β (written $\alpha \Downarrow \beta$) if:

$$\textbf{nec}\ \forall x\ (x\ \varepsilon\ \alpha \supset \exists y\ (y\ \varepsilon\ \beta \wedge x \not\leq y \wedge y \not\leq x)).$$

α is called *founded* (written $\alpha \Downarrow$) if there exists a β such that $\alpha \Downarrow \beta$. α is called *essentially independent* (written $I(\alpha)$) if $\neg (\alpha \Downarrow)$, and *self-founding* if $\alpha \Downarrow \alpha$.

The meaning of the above definition is that, in order α to be founded on another concept β, any instance x of α has to be necessarily associated to an instance y of β *which is not related to x by a part-of relation*. In other words, the instances of α cannot exist as such except *in a more comprehensive unity* where they are associated to some other object. For example, *son* is founded since sons *as such* exist only within the framework of a family, where they are associated to their parents. On the other side, *car* is essentially independent since, although the existence of a car implies the existence of its engine, this engine does not necessarily belong to a more comprehensive unity of which the car is a part, but is instead a part of the car. Finally, *spouse* is an example of a self-founding concept.

2.2 Roles and natural concepts

The notion of foundation defined above seems to be very appealing in order to give a formal definition to Sowa's roles: if we stipulate that a role is a founded concept, not only concepts denoted by *relational nouns* like *son* or *spouse* [8] fall under this category, but also concepts like *pedestrian,* exactly as argued by Sowa. Indeed, both categories are associated to some "pattern of relationships": the association is *explicit* for the former and *implicit* for the latter.

However, there is a class of relational nouns not mentioned by Sowa, which would fit very hard with the natural meaning of "role" and still fit our definition of founded concepts. They usually denote *qualities* like *color, weight, velocity*, or *position*. Take for instance *color*: if *blue* is a color, then necessarily exists an object whose color is *blue,* which is not related to the object *blue* by a part-of relation[2]. The concept *color* is therefore founded, but it does not seem to fit the dictionary definition of a role reported above. The proper ontological classification of these concepts will be given in section 4; for the time being, we simply want to exclude them from

[1] It should better called *predication* relationship, but for our purposes may be equally intended as a membership.
[2] Unless we accept the (somewhat unnatural) view that properties of an object like its color are also parts of that object.

being roles. We propose therefore to introduce a further condition, which, together with foundedness, gives us a more adequate characterization of the meaning of roles, distinguishing them from other kind of concepts.

Definition 2.2 A concept α is called *semantically rigid* (written $R(\alpha)$) if

$$\forall x\ (x\ \varepsilon\ \alpha \supset \mathbf{nec}\ (x\ \varepsilon\ \alpha)).$$

Intuitively, we can think that a concept α is semantically rigid if it contributes to the very identity of its instances, in such a way that, if x is an α in a particular situation, it has to keep to be an α in any possible situation in order to keep its identity. For instance, an animal can cease to be a pup while still being a dog: *animal* and *dog* are semantically rigid, *pup* is not.

Definition 2.3 A concept α is called a *role* if it is founded but not semantically rigid, that is, $\alpha \Downarrow \wedge \neg R(\alpha)$.

According to this definition, qualities cannot be roles, since they are semantically rigid: if what is a color would (by absurd) cease to be a color, it would be something *else*. On the other side, all the other concepts considered so far as plausible roles appear to be not semantically rigid. We see therefore that two independent notions contribute to the meaning of a role: the former is the notion of some "pattern of relationships" which defines the meaning of a role only within a more general unity; the latter is bound to the Aristotelian distinction between substance and accident, and guarantees that (using Sowa's words) this "pattern of relationships" is a "particular" one.

The two notions of foundations and semantical rigidity can be used to introduce an important set of concepts which is disjoint from that of roles and which is usually not formally defined: the set of *natural concepts*.

Definition 2.4. A concept α is called a *natural concept* if it is essentially independent and semantically rigid, that is, $I(\alpha) \wedge R(\alpha)$.

Example 2.1 *dog* is an example of a natural concept. Observe that, according to the above definitions, *color* is neither a role nor a natural concept, since it is semantically rigid but not essentially independent. The same is true for *pup* or *lame-dog,* since they may be considered as essentially independent but not semantically rigid.

3 Slots and attributes

After having re-established the natural meaning of the term “role”, we shall investigate in this section on the relationship between this meaning and that of the entities called “roles” in KL-ONE. Provisionally, we shall call these entities “KL-ONE roles”.

The first observation is that the latter of Brachman’s definitions cited in the introduction is a dangerous generalization of the former, since it presents a subtle internal contradiction[3]: if something has to be a *component* of a concept it cannot be an *arbitrary* two-place predicate! To see that, let us recall one of Woods’ examples [25]:

JOHN
    HEIGHT: 6 FEET
    HIT: MARY

Woods argues that, in this case, "no longer do the link names stand for attributes of a node, but rather arbitrary relations between the node and other nodes", and suggests to use a different notation for the two types of knowledge. The choice made with KL-ONE is in this direction; however, as long as KL-ONE roles are *arbitrary* binary relations, there is no *a priori* criteria to distinguish among the two different kinds of knowledge. For instance, Brachman ([3], p. 137) tackles Woods’ example by conceptualizing the *hit* relation and therefore representing *mary* as the filler of an *object* role for *hit*, but nothing in his formalism prevents the user from adopting another formalization, by representing *mary* as the filler of the *hit* “role”. What we want to stress is that *hit* should be *forbidden* to be a KL-ONE role: the intended meaning of KL-ONE roles – as results from the former of the two definitions above – is *something else*. It is this difference which should differentiate them from slots. We suggest to call *attributes* those KL-ONE roles which capture the original intended meaning of conceptual components, while adopting the term *slot* for generic equivalents of two-place predicates.

3.1 Woods' linguistic test

We think that we can find the right meaning of attributes by looking at the linguistic interpretation suggested by Woods:

Y is a value of the attribute A of X if we can say that *Y is a A of X* (or *the A of X*).

If we cannot find a Y that fits this expression, A *cannot be an attribute*. There is a simple constraint that must be satisfied in order to fit the above interpretation: *an attribute name has to denote a concept* (that is, it has to be a noun). It is here that we face the connection between

[3] There is another serious problem in this definition, namely that the intensionality present in the functional meaning of roles cited in the former definition has disappeared. We discuss of the intensionality of KL-ONE roles elsewhere [11].

concepts and binary relations we have mentioned above: the kind of relations we would like to use as conceptual components of an object have *names* which may easily be names of *concepts*.

3.2Attributes as concepts

The solution we propose is to represent attributes as concepts, i.e. unary relations, which have an *associated* binary relation. We call this binary relation the *relational interpretation* of the attribute. The two kinds of relations can be easily distinguished at the syntactic level on the basis of their different arity, and therefore they can have the same functor *name;* the implicit connection between them can be accounted by an *enforced semantics*. This choice will be clearer while considering the following (super-simplified) assertional language AL:

Definition 3.1 The *alphabet* of AL consists of the following disjoint sets of symbols:

(a) a set O of *objects;*
(b) a set C of *concepts,* with a distinguished subset A of *attributes*.

Definition 3.2 An *atomic formula* of AL is defined as follows:

<atomic_formula> ::= <object_description> | <relation_description>
<object_description> ::= (*concept object*).
<relation_description> ::= (*attribute object object*).

Non-atomic formulas of AL are defined as usual.

Notice that, since attributes are concepts, there may exist some relation descriptions which have the same functor of object descriptions.

Definition 3.3 An *interpretation* of AL is an ordered tuple $\mathbf{I} = <U, \delta, \rho>$, where:

(a) U is an arbitrary set called the *universe of discourse;*
(b) δ is a function from $O \cup C$ into $U \cup 2^U$ called *denotation function*, such that for each $x \in O$ and $\alpha \in C$, $\delta(x) \in U$ and $\delta(\alpha) \in 2^U$;
(c) ρ a function from A into $2^{U \times U}$ called *relational interpretation function,* such that, for each attribute α and each $<x, y> \in \rho(\alpha)$, $y \in \delta(\alpha)$.

Definition 3.4 An atomic formula ϕ of AL is *satisfied* by an interpretation $\mathbf{I}$ (written $\mathbf{I} \models \phi$) iff it satisfies one of the following conditions:

(a) $\mathbf{I} \models (\alpha\ x)$ iff $\delta(x) \in \delta(\alpha)$;
(b) $\mathbf{I} \models (\alpha\ x_1\ x_2)$ iff $<\delta(x_1), \delta(x_2)> \in \rho(\alpha)$.

We see that the *denotation* of a given attribute α is still unique and identical to that of a concept, while the interpretation of a relation description with functor α is made with the help of the *auxiliary* function ρ, which gives the relation *associated* to α. The connection existing between the two meanings is expressed by the constraint which forces the range of $\rho(\alpha)$ to be included in $\delta(\alpha)$. This semantic constraint corresponds, at the syntactic level, to the following (second-order)

Axiom 3.5 *(Attribute Consistency Postulate)***.** Any value of an attribute is also an instance of the concept corresponding to that attribute. In AL, this means that $\forall_{\alpha \in A} \forall_{x,y \in O} (\alpha\ x\ y) \supset (\alpha\ y)$.

In conclusion, we can formulate the following general definition:

Definition 3.6. An *attribute* is a concept which, in the domain of interest, has a unique relational interpretation, satisfying the Attribute Consistency Postulate.

Let us assume, for the time being, that this relational interpretation is given by the user, as we have seen for the language AL. We shall remove this assumption in the next section, discussing the ontological status of attributes. First, we would like to discuss the implications of the choice we have made. With the uniformity between attributes and concepts, we eliminate a source of confusion and redundancy within current applications of frame-based languages. Separate attribute hierarchies are no more necessary, and range declarations like *(all age Age)* become superfluous. Moreover, we capture Brachman's original intuition (abandoned in subsequent formalizations) on the "abstract commonality" among KL-ONE roles with the same name ([3], p. 141, our italics)

> (...) while these roles [i.e., those with the same name] on the surface appear disparate, there is a strong common sense between them. (...) While I do not as yet have a proposal on how this might be done, it does appear to be a fruitful research area. (...) The abstraction of the commonalities between locally defined roles *separates this notation from one which uses role names merely as convenient names for slots*.

We see that it is exactly this "abstract commonality" which should account for the difference between KL-ONE roles and ordinary slots. But current terminological languages do not exhibit this conceptual distinction as far as their "roles" are interpreted as *arbitrary* binary relations. Only the entities we have called *attributes* are really different from slots.

3.3. Naming discipline

We have seen that, in order to suit our interpretation, attribute names have to pass "Woods' linguistic test". The first consequence of this is the necessity of a strong *discipline* in choosing attribute names. For instance, all commonly used slot names containing prepositions (*childOf, connectedTo*), verbs (*hasPart, ifNeeded*) or plurals (*parts, instances*) cannot be considered as attributes, and – if possible – they have to be substituted with *father, connection, part, acquisitionProcedure, instance,* and so on. We think that this constraint is definitely not a disadvantage: often what distinguishes a good piece of software from a bad one is the choice of the right names. Moreover, in the effort for finding right attribute names we often learn more about the conceptual structure of our domain, and improve the *value* of our knowledge base. In our opinion, this value is strictly bound to the easiness of *knowledge integration*, which in turns depends on the *granularity* of the representation and on its *cognitive transparency*.

Of course, not always is this substitution immediate, natural, or convenient; we argue that this situation is for us a "signal" that something must (or may be) represented in a totally different way. To see that, let us consider again Woods' example. We have seen that it refers to two different kinds of knowledge; however, within current hybrid systems belonging to KRYPTON's mainstream, both pieces of knowledge have to be put into the Abox, without any possibility of a conceptual distinction. As discussed in greater detail in [10, 12], we propose instead to consider as terminological knowledge all the knowledge *about* (the internal structure of) terms, i.e. about *structural* relationships between terms. On the other side, *relational knowledge* is knowledge about *arbitrary* relationships between terms. Notice that individuals *are* terms, both from the logic and from the linguistic point of view. Therefore, the proposition "the height of John is 6 feet" is a piece of terminological knowledge about the term *john*, while "John hit Mary" is a piece of relational knowledge about a particular relationship existing between the two terms *john* and *mary*. Now, since *hit* cannot be an attribute of *john*, the latter statement has to be expressed as relational knowledge. Our characterization of attributes enforces in this way an important conceptual distinction.

As an intriguing example of the danger of interpreting KL-ONE roles as arbitrary relations, consider the KL-ONE role *self,* whose denotation is the identity relation, introduced by Levesque and Brachman [14] in order to prove the intractability of subsumption for FL. In the light of the above discussion, *self* should not be an attribute, since it is not a concept. By the way, we feel that it is exactly the nature of attributes as concepts which allows us to use them to define concepts in terms of *something else:* how can we think of defining a concept in terms of the property of being identical to itself?

4. A basic ontology for attributes

The definition of attributes we have given in the previous section leaves a number of questions still open. Given that attributes are concepts, which are their ontological properties? Is there a criterion to decide whether a concept can be an attribute and to define its relational interpretation? How are attributes related to roles and natural concepts? In this section we try to answer these questions by proposing a schematic (meta)ontology of concepts, which distinguishes among natural concepts, roles and attributes (Fig. 1).

Definition 4.1. A *domain scheme* is a tuple **D** = $<U, C, R, \delta>$, where U is an arbitrary set called the *universe of discourse*, C a set of symbols called *concepts*, R a set of binary relations on UxU called *relevant relations*, and δ a function from U into 2^U called *denotation function*.

Given a domain scheme, we present in the following the conditions for a concept $\alpha \in C$ to be an attribute, removing the assumption made in the previous section, that the relational interpretation of an attribute is given by the user.

4.1. Relational attributes.

Let us first consider relational nouns such as *son* or *spouse*. They have a "natural" relational interpretation which is directly associated with their name, and therefore they should satisfy Def. 3.6. We want to introduce an ontological criterion which forces them to be attributes, such that their relational interpretation is the obvious one. The key idea we explore below is that the relation we are looking for can be defined in terms of the *founding* relation.

Definition 4.2. Let **D** = $<U, C, R, \delta>$ be a domain scheme and $\alpha \in C$ a concept in **D**. A binary relation $R \in R$ is called a *partial relational interpretation* of α in **D** if there exists a concept $\beta \in C$ such that $\alpha \Downarrow \beta \wedge dom(R) \subseteq \delta(\beta) \wedge range(R) \subseteq \delta(\alpha)$.

Definition 4.3. Let **D** = $<U, C, R, \delta>$ be a domain scheme, $\alpha \in C$ a concept in **D**, and $R_\alpha \subseteq R$ the set of its partial relational interpretations. The concept α is called a *relational attribute* in **D** if R_α is non-empty. Its *relational interpretation* $\rho(\alpha)$ is given by $\rho(\alpha) = \bigcup_{R_i \in R_\alpha} R_i$. A relational attribute is called *non-ambiguous* in **D** if the domains of its relational interpretations are mutually disjoint.

Example 4.1. Suppose that the concepts *teacher, subject* and *student* are defined in a domain **D** (with δ satisfying their obvious meaning), as well as the two relations R_1: δ(*subject*) $\rightarrow \delta$(*teacher*) and R_2: δ(*student*) $\rightarrow \delta$(*teacher*). Since *teacher* is founded both on *subject* and on *student*, R_1 and R_2 are both partial relational interpretations of *teacher*, with

ρ(*teacher*)=$R_1 \cup R_2$. *teacher* is therefore a relational attribute. It is non-ambiguous since *subject* and *student* are disjoint.

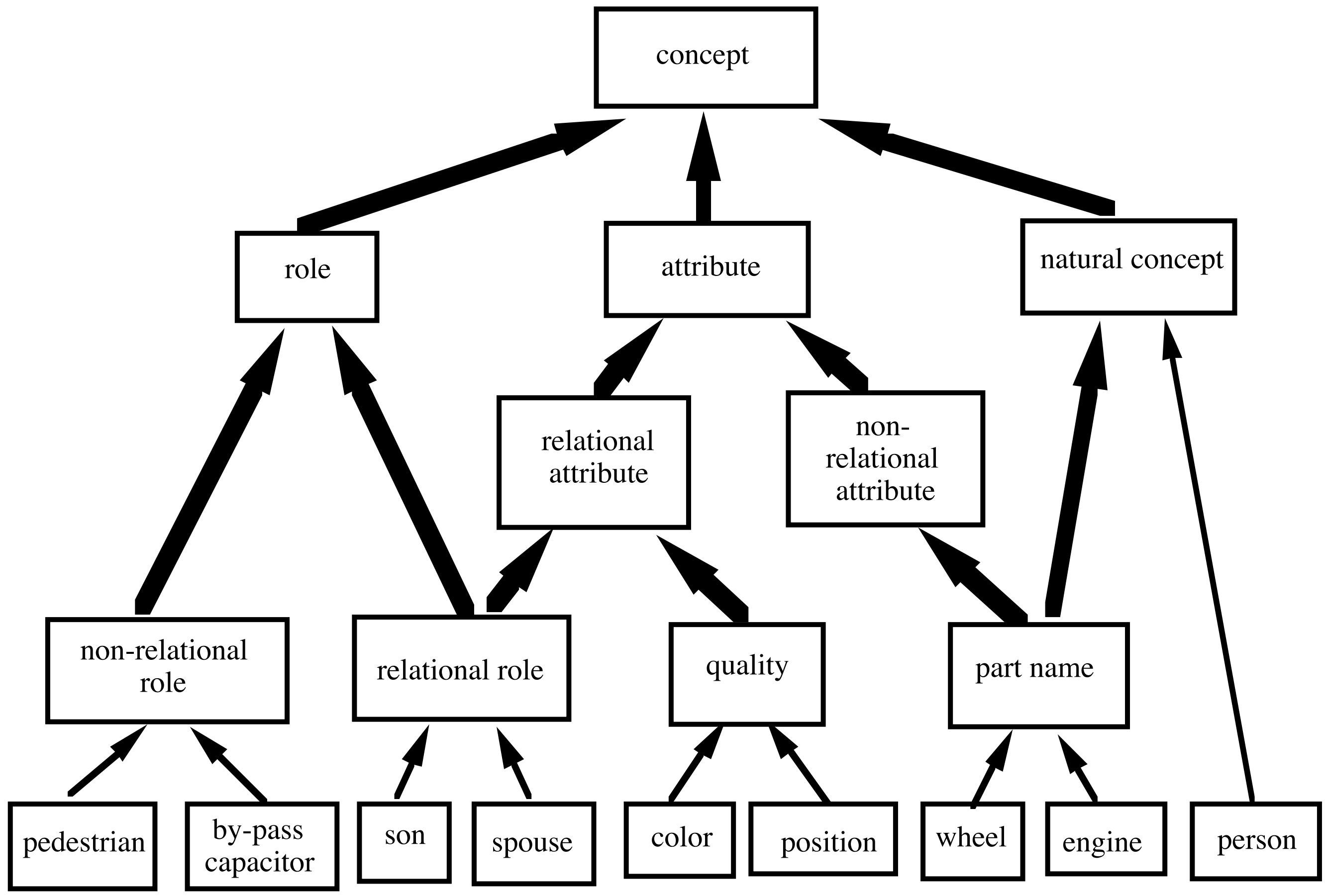

Figure 1. A basic ontology of attributes, showing their relationships with roles and natural concepts. Thick arrows represent inclusion relationships,while thin arrows membership relationships.

Let us consider now the left side of Fig. 1. Relational roles represent those concepts which are founded, non-rigid, and have a relational interpretation: they are therefore the intersection between roles and attributes. Qualities are relational attributes since they are founded and have a relational interpretation; they are however disjoint from roles because of their semantic rigidity. We would like to stress that the distinction between qualities and relational roles is more than a mere subtlety, since qualities present two important peculiarities with respect to relational roles. First, their instances are *properties*, i.e. *predicable entities* [20]; indeed, *red* or *on* may well be names of predicates, but the same is not true for *john* as an instance of the relational role *son*. Second, their relational interpretation is a *function*; that is, features are single-valued attributes. The latter aspect has important computational implications.

4.2. Non-relational attributes

Coming now to the right side of Fig. 1, let us notice that there is a whole category of concept-structuring terms which we want to consider as attributes but are not relational nouns. The most important are those related with *part-names*. Consider for instance an AL statement like *(wheel car#467 wheel#7279)*. The term *wheel* contributes to the characterization of *car#467,* but it is not a role, since it does not satisfy Def. 2.3. The reason is that a wheel is always a wheel, independently of its participation to the structure of a more complex entity. Notice that it cannot be a relational attribute, since it is not founded. Still, it satisfies Def. 3.6 for attributes, since we could easily conceive a relation named *wheel* (satisfying the Attribute Consistency Postulate) mapping cars (or whatever) into particular wheels. However, this relation would not be part of the nature of a wheel.

Differently from previous cases, the attribute *wheel* should be therefore linked to its relational interpretation in an indirect way. It is important to notice that, again, this should not be an *arbitrary* relation, since it seems to have a precise ontological nature, imposing a formal constraint to the relation: it has to be a specialization of a *part* relation, whose range is restricted to be a *wheel*. The Attribute Consistency Postulate is still valid.

But who tells us that we are dealing with a part? If the context was different, say *(wheel john wheel#7279),* the preferred interpretation might be, for instance, that *wheel#7279* is one of the wheels John is playing with, or one of the wheels of his car: in both cases, not a *part* of John. We can observe then the second difference with respect to other attributes: the ontological nature of the relation *wheel* is (indirectly) linked to cannot be predicted *a priori* on the basis of its *name*. That is, it is a property of the *link* between *wheel* and the object it refers to. The conclusion is that while the semantics of relational attributes may be in general easily understandable from the context (except those cases of ambiguity mentioned in Def. 4.3), this is not true for non-relational attributes.[4] As discussed in [11], we need a granularity finer than that of current frame-based languages in order to express this kind of knowledge.

A possibility to limit the potential ambiguities of non-relational attributes without resorting to more expressive languages is to limit their use to (generalized) *parts*[5], as appears in Fig. 1. This

[4] The only possible interpretation is a general "belonging" relationship between the attribute and the object it refers to, but this is too weak for any practical use. This situation has been called by Wilensky the "belonging fallacy" of frame based languages [23]. Wilensky however does not make our distinction between relational and non-relational attributes.

[5] We acknowledge that the *part* relationship has not been adequately characterized, since we do not distinguish among different kinds of parts. A thorough analysis of the problems with parts can be found in [24, 7]. The intuitive meaning of parts is however sufficient to support the present discussion. A deeper ontology for parts and features has been explored in [1].

means that, for instance, the interpretation of attributes as *possessions* should be avoided. Consider for instance the two following statements:

(book padovaPublicLibrary divinaCommedia)
(book john divinaCommedia).

We argue that *book* should not be used as an attribute of John in the second case, since its ontological role may be ambiguous: for instance, the interpretation of *Divina Commedia* as a *part* of John should be excluded, but there is no possibility to formally force that. However, the restriction of attributes to either relational nouns or parts may lead us to a better way to express the same information. For instance, we can use the relational role *possession* to state *(possession john divinaCommedia).*

In conclusion, we can complete our ontological formalization of attributes as follows:

Definition 4.4. Let **D**=$<U, C, R, \delta>$ be a domain scheme. A concept $\alpha \in C$ is called a *non-relational attribute* in **D** if: (i) α is a natural concept; (ii) there exists a relation $< \in R$ to which it has been given the meaning of a *proper part* relation; (iii) the *relational interpretation* $\rho(\alpha) = \{<x, y> \mid y < x \wedge y \in \delta(\alpha)\}$ is not empty.

Definition 4.5. Let **D**=$<U, C, R, \delta>$ be a domain scheme. A concept $\alpha \in C$ is called an *attribute* in **D** if it is either a relational or a non-relational attribute in **D**.

5. Discussion and conclusions

First of all, we would like to stress that the conceptual distinction between "right" and "non right" attribute names does not depend on the formal semantics we have presented, nor on the choice to give the same name to attributes-as-relationships and attributes-as-concepts: the first result of our analysis is an ontological characterization of attributes as the main components of a concept, which can immediately be applied as a criterion to discriminate between attributes, other concepts and arbitrary relations. A possible methodology to decide whether A can be an attribute for an object X could be the following:

1. **Check** whether A and X satisfy Woods' linguistic test: "something is the/an A of (some) X". This test ensures that A is a concept, and that it may be actually related with X.
2. **If** the linguistic test succeeds **then**
   if A is founded on X **then**
      return true (A is a relational attribute).

else **if** A is the name of a part of X **then**
return true (A is a non-relational attribute)
else
return false.

Let us briefly compare our methodology with the one recently proposed by Brachman and colleagues for the CLASSIC language [5]. These authors propose a general principle for distinguishing concepts from roles (in the KL-ONE sense) which is very simple and works well in many cases. The principle is that *objects which depend on other objects for their existence must be roles*. In our terminology, this means that foundation is the only criterion used to decide about attributes. The excessive simplicity of this criterion forces the authors to admit exceptions to their rule: for instance, although they acknowledge that *grape* may have an independent existence (and therefore is not founded), they use it as a role for *wine* because it is natural to refer to a "wine's grape". In this way Brachman and colleagues implicitly admit the importance of Woods' linguistic test, but they do not include it among their principles.

Now, according to our discussion, *grape* is not a good attribute for *wine* since, although it fits Woods' linguistic test, it is not founded, nor is it the name of a part of a wine. A better attribute may be *constituent,* with value restriction *grape*. Our methodology gives a criterion to check whether an attribute is the right one *for a given object,* and, in case of failure, forces us to reformulate knowledge in terms of more *basic* relations. Moreover, it can be easily specialized to discriminate between relational roles, features and parts.

However, the real limitation of CLASSIC's methodology is that the principles they introduce are only used to *suggest* possible roles, but not to *exclude* some relations from being roles. This means that roles standing for arbitrary relations like *hit* or *likes* are explicitly admitted: "roles are relationships between individuals".

Coming now to the semantics we have presented, it may be observed that it actually assigns more than one interpretation to those concepts which are also attributes. We have already observed that, technically, the denotation of any symbol of the alphabet is still unique. We would like to add however that this kind of situation is frequent in natural language: it is the way of reference to a symbol (i.e., in many cases, the syntax) which characterizes its actual semantics. In the case of KL-ONE roles, Trost and Steinacker [22] distinguish between definitional and functional references (e.g. a hammer as an artifact and a hammer as instrument of an action), but reach a conclusion which is the opposite of ours: they do not admit any necessary common meaning between the two interpretations, while we have presented some ontological motivations for their mutual dependence.

Acknowledgements

This research has been made in the framework of a special National project on Hybrid Systems, supported by the "Progetto Finalizzato Informatica e Calcolo Parallelo" of the Italian National Research Council. I would like to thank Luca Boldrin, Daniele Giaretta and Dario Maguolo for their comments on earlier versions of this paper.

Bibliography

[1] L. Boldrin, Formalismi logici per la rappresentazione della relazione parte-tutto nell'ambito della realizzazione di ontologie condivise, Thesis, University of Milan, Corso di laurea in Scienze dell'Informazione, 1991.

[2] A. Borgida, R. J. Brachman, D. L. McGuinness, and L. A. Resnick, CLASSIC: a structural data model for objects, in: *Proc. of the 1989 ACM SIGMOD Int. Conf. on Management of Data* (1989) 59-67.

[3] R. J. Brachman, What's in a concept: structural foundations for semantic networks, *Int. J. Man-Machine Studies* (1977) 127-152 .

[4] R. J. Brachman, On the epistemological status of semantic networks, in: N. Findler, ed., *Associative networks: representation and use of knowledge by computers* (Academic Press, 1979).

[5] R. J. Brachman, D. L. McGuinness, P. F. Patel-Schneider, L. A. Resnick, and A. Borgida, Living with CLASSIC: When and How to Use a KL-ONE-like Language, in: J. Sowa, ed., *Principles of Semantic Networks* (Morgan Kaufmann 1990), 401-456.

[6] R. J. Brachman and J. G. Schmolze, An Overview of the KL-ONE Knowledge Representation System, *Cognitive Science* 9 (1985), 171-216.

[7] R. Chaffin and D. J. Herrmann, The nature of semantic relations: a comparison of two approaches, in: M. W. Evens, ed., *Relational Models of the Lexicon* (Cambridge University Press 1988).

[8] J. de Bruin and R. Scha, The Interpretation of Relational Nouns, in: *Proc. of ACL 88* (Buffalo, New York, 1988).

[9] M. W. Freeman, The qua link, in: J. G. Schmolze and R. J. Brachman, eds., *Proc. of the 1981 KL-ONE Workshop* (Jackson, New Hampshire, 1981) 54-64.

[10] N. Guarino, Nature and structure of terminological knowledge, in: *Proc. 1st Conf. of the Italian Association for Artificial Intelligence (AI*IA 89)* (Trento, Italy, December 1989).

[11] N. Guarino, What's in a role: towards an ontological foundation for terminological logics. Italian National Research Council, LADSEB-CNR Int. Rep. 06/90, 1990.

[12] N. Guarino, A concise presentation of ITL, *ACM SIGART Bulletin,* special issue on Implemented Knowledge Representation and Reasoning Systems, 2 (1991), 61-69. A

revised and extended version appears in: M. M. Richter and H. Boley, eds., *Proc. of Int. Workshop on Processing Declarative Knowledge (PDK '91)* (Lectures Notes in Artificial Intelligence, Springer Verlag 1992).

[13] I. J. Haimowitz, Using NIKL in a Large Medical Knowledge Base. MIT Technical Memo LCS/TM-348, 1988.

[14] H. Levesque and R. J. Brachman, Expressiveness and tractability in knowledge representation and reasoning, *Computational Intelligence* 3 (1987), 78-93.

[15] R. M. McGregor, A deductive Pattern Matcher, in: *Proc. of AAAI-88* (Morgan Kaufmann 1988)

[16] M. G. Moser, An Overview of NIKL, the New Implementation of KL-ONE. BBN Rep. 5420, 1983.

[17] C. Peltason, A. Schmiedel, C. Kindermann, and J. Quantz, The BACK system revisited. Dept. of Computer Science, Technical University of Berlin, Technical Report KIT-75, 1989.

[18] P. M. Simons, Three Essays in Formal Ontology: I - The Formalization of Husserl's Theory of Wholes and Parts, in: B. Smith, ed., *Parts and Moments: Studies in Logic and Formal Ontology* (Philosophia Verlag, München, 1982), 113-159.

[19] B. Smith, ed., *Parts and Moments: Studies in Logic and Formal Ontology* (Philosophia Verlag, München, 1982).

[20] B. Smith and K. Mulligan, Pieces of a theory, in: B. Smith, ed., *Parts and Moments: Studies in Logic and Formal Ontology* (Philosophia Verlag, München, 1982), 15-109.

[21] J. F. Sowa, Using a lexicon of canonical graphs in a semantic interpreter, in: M. W. Evens, ed., *Relational Models of the Lexicon* (Cambridge University Press, 1988), 113-137.

[22] H. Trost and I. Steinacker, The Role of Roles: Some Aspects of the Real World Knowledge Representation, in: *Proc. of IJCAI-81*, 1981.

[23] R. Wilensky, Some problems and proposals for knowledge representation. University of California, Berkeley, Rep. UCB/CSD 87/351, 1987.

[24] M. E. Winston, R. Chaffin, and D. Herrmann, A Taxonomy of Part-Whole Relations, *Cognitive Science* 11 (1987), 417-444.

[25] W. A. Woods, What's in a Link: Foundations for Semantic Networks, in: D. G. Bobrow and A. M. Collins, eds., *Representation and Understanding: Studies in Cognitive Science*, (Academic Press 1975). Also in R. Brachman and H. Levesque, eds., *Readings in Knowledge Representation* (Morgan Kaufmann 1985).