Attribute meta-properties for formal ontological analysis

Valentina Tamma and Trevor J.M. Bench Capon

Department of Computer Science, University of Liverpool,
Chadwick Building, Liverpool L69 7ZF, UK,
{valli, tbc}@csc.liv.ac.uk

Abstract. Formal ontological analysis is a methodology that uses ideas from philosophy in order to guide the process of building ontologies with a correct and as untangled a structure as possible.
This paper presents an ontology model that aims to facilitate formal ontological analysis, by providing a set of *meta-properties* which characterise the behaviour of concept properties in a concept definition, to provide a richer semantics of the concept. We describe concepts in terms of their attributes (characterising features) and we also describe the role played by these features in the concept definition: whether they are prototypical or exceptional; whether they are permitted to change over time, and if so, how often this happens; how likely is a concept to show these features, etc. We show that these meta-properties, besides enriching concept descriptions, can be used to determine whether the notions of *identity* and *rigidity* hold, thus supporting in part the OntoClean [31] methodology.

1 Introduction

Many current applications such as e-commerce and the semantic web rely on the ability of different resources or agents to interoperate with each other and with users. In some cases, interoperation becomes quite complex, because agents may have been independently developed, and so the assumption that agents use the same communication language and the same terminology in a consistent way cannot be made. When dealing with independently developed agents, their interoperability with humans and others depends on their ability to understand each other.

Ontologies are an explicit, formal specification of a shared conceptualisation, where *a 'conceptualisation' refers to an abstract model of some phenomenon in the world by having identified the relevant concepts of that phenomenon. 'Explicit' means that the type of concepts used, and the constraints on their use are explicitly defined. · 'Formal' refers to the fact that the ontology should be machine-readable. 'Shared' reflects the notion that an ontology captures consensual knowledge, that is it is not private to some individual, but accepted by a group* [27]. That is, ontologies provide a formally defined specification of the meaning of those terms that are used by agents to interoperate.

Agents can differ in their understanding of the world surrounding them, in their goals, and their capabilities, but they can still interoperate in order to perform a task, provided they can reach agreement on a shared understanding, mainly obtained by reconciling the differences. This kind of reconciliation might be accomplished by *merging* the ontologies to which the agents refer, that is, by building a single ontology that is the merged version of the different ontologies, which often cover similar or overlapping domains [4].

Ontology merging starts with the attempt to find the places in which the source ontologies overlap [18], that is the coalescence of two semantically identical terms in different ontologies so that they can be referred to by the same name in the resulting ontology. This is the only step of the merge process which is relevant to the scope of this article. The coalescence of terms in diverse ontologies has to be accomplished despite heterogeneous agent ontologies and heterogeneity has to be reconciled in order to share knowledge. Defining the different types of heterogeneity is out of the scope of this article, although we recognise that it can hinder attempts to coalesce terms, especially when it is the semantics that is heterogeneous. Ontology or semantic heterogeneity occurs when different ontological assumptions about overlapping domains are made [30].

The ontologies involved in the merging process, be they heterogeneous or not, are usually assumed to be either built according to some kind of engineering methodology, such as Methontology [3], or their ontology taxonomic structures are validated according to some methodology such as OntoClean [31]. Both methodologies aim to ensure that the ontology obtained after applying them is correct, that it does not contain cycles or recursive definitions, and it has a taxonomic structure that is as untangled as possible.

Methontology and OntoClean are complementary methodologies in that the former provides the guidelines for building or re engineering ontologies, whereas the latter can be used either in the validation step (when ontologies are engineered or restructured) or simultaneously with the ontology construction (when ontologies are built from scratch). These two methodologies are currently undergoing an integration process [2] as part of the activities of the OntoWeb special interest group on Enterprise-standards Ontology Environments (SIG's home page: `http://delicias.dia.fi.upm.es/ontoweb/sig-tools/index.html`).

Methodologies to obtain well-built ontologies, however, are not sufficient to support a semi-automatic coalescence process. In fact we cannot recognise whether two concepts (that can be heterogeneous) are similar, only on the basis of the terms denoting them, the relationships with other terms, and their descriptions, but we need to have a full understanding of the concepts. As noted by McGuinness [17], an explicit representation of the semantics of terms would be a step towards understanding whether two concepts are similar. It emerges that the current ontology models are not expressive enough to provide such an explicit representation. Even when heavyweight ontologies are used (that is, concepts described in terms of attributes, linked by relations, organised into an Is-a relationship and constrained by axioms), their expressiveness does not allow a full account of the semantics of the concepts described. The ontology model we present in this paper is enriched by attribute meta-properties which account for the behaviour of attributes in the concept definition.

This paper is organised as follows: Section 2 presents the OntoClean methodology and the notions of formal ontological analysis, while Section 3 introduces our proposal for an ontology model encompassing a set of meta-properties for attributes which are then discussed in the following subsections. This ontology model was also presented in [29]: in this paper we do not discuss any implementation issues and we concentrate on the meta-properties, clarifying the relationship with the concept meta-properties used in OntoClean and the role attribute meta-properties play in associating senses to concepts. Section 4 discusses the attribute meta-properties and relates them with two notions (identity and rigidity) of formal ontological analysis and with roles. Finally, Section 5 draws conclusions.

2 Identity, Unity, Essence, and Dependence and their use in OntoClean

OntoClean [31] is a methodology to perform a *formal ontological analysis* on taxonomies in order to verify which formal meta-properties hold, thus making clear and explicit the modelling assumptions made while designing the ontologies. The clarification and explication of the modelling assumptions is a necessary step in evaluating ontologies, since it permits knowledge engineers to detect and reconcile ontological conflicts that may affect one or more ontologies. Ontological conflicts may become apparent when two ontologies are compared in order to coalesce terms, and they reveal cases of ontological heterogeneity. For example, two well known ontologies [1] (Wordnet [19] and Panglos [11]), present the following conflict: one models Physical Object as subconcept of Amount of matter wheres the other models Amount of matter as subconcept of Physical object. This is a case of ontology heterogeneity due to different modelling of the concepts. Such ontological conflicts need to be detected and resolved if terms are to be coalesced.

OntoClean is firmly based on the philosophical notions of *identity, unity, essence (rigidity),* and *dependence.* The attribute meta-properties we present in this paper are related to these notions, and we discuss them below.

Identity: Identity is the logical relation of numerical sameness, in which a thing stands only to itself. Based on the idea that everything is what it is and not another thing, philosophy has tried for a long time to identify the criteria which allow a thing to be identified for what it is even when it is cognised in two different forms, by two different descriptions and/or at two different times [32, 9]. This comprises both aspects of finding constitutive criteria (which features a thing must have in order to be what it is), and of finding re-identification criteria (which features a thing has to have in order to be recognised as itself by a cognitive agent).

---

[1] Strictly speaking, neither Wordnet nor Panglos are ontologies. However, they are often used and referred to as ontologies.

OntoClean does not make any difference between identity and re-identification, but we believe that these are distinct, although equally important aspects of identity. It happens to be the case the fingerprints are unique to individuals. This means that, in the actual world, fingerprints can serve as re-identification criteria. But it is possible that everyone had the same fingerprints: in such a possible world, fingerprints would not provide re-identification criteria. Moreover, fingerprints cannot be used to discover identity across possible worlds, and cannot be a criterion *constitutive* of such identity.

Although the problem of *identifying* what features an entity should have in order to be what it is and recognised as such has been central to philosophy, it has not had the same impact in conceptual modelling and more generally AI. The ability to identify individuals is central to the modelling process; more precisely, it is not the mere problem of identifying an entity in the world that is central to the ontological representation of the world, but the ability to *re-identify an entity in all its possible forms*, or more formally, *re-identification in all possible worlds*. That is, the problem is related to distinguishing a specific instance of a concept from its siblings on the basis of certain *characteristic properties* which are unique and intrinsic to *that instance*. Intrinsic properties are usually modelled as *attributes*.

Identity is, of course inherently time dependent, since time gives rise to a particular system of possible worlds where it is highly likely that the same instance of a concept exhibits different features This problem is known as *identity through change*: an instance of a concept may remain the same while exhibiting different properties at different instants of time. Therefore it becomes important to understand not only which features or properties can change and which cannot [31], but also, we add, the situations that can trigger such changes.

Identity is an absolute notion (whereas re-identification is not), although we recognise that applying identity to certain concepts, such as those representing artifacts, is not always straightforward.

Unity: the notion of *unity* is often included in a more generalised notion of identity, although these two notions are different. While identity aims to characterise what is unique for an entity of the world when considered as a whole, the goal of unity is that of *distinguishing the* parts *of an instance from the rest of the world by means of a* unifying relation *that binds them together (not involving anything else)* [31]. For example, the question 'Is this my car?' represents a problem of identity, whereas the question 'Is the steering wheel part of my car?' is a problem of unity. Also the notion of unity is affected by the notion of time; for example, can the parts of an instance be different at different instants of time?

Essence: The notion of *essence* is strictly related to the notion of *necessity* [10]. An *essential property* is a property that is necessary for an object, that is, a property that is true in every possible world [15]. Based on the notion of *essence*, Guarino and colleagues [8] have introduced the notion of *rigidity*. A rigid property is a property that is necessary to all instances in any instant of time, that is a property ϕ such that: $\Box(\forall x,\ t\phi(x,\ t) \rightarrow \Box\forall t'\ \phi(x,\ t'))$. For this formula, and in the remainder of this paper, we use the modal notions of *necessity* $\Box$ and *possibility* $\Diamond$ quantified over possible worlds (in Kripke's semantics [13]), meaning that the extension of predicates concerns what exists in any possible world. We use these operators according to the following meanings: $\Box\ \phi$ means that ϕ holds in *all* possible worlds $\Diamond\ \phi$ means that ϕ is possible, i.e. that ϕ holds in *at least* one possible world, which might be accessible from the actual world. Rigidity strictly depends on the notions of *time* and *modality* [29]; this point is further elaborated in Section 4.2. It is important,however, not to confuse modal necessity with temporal permanence. Modal necessity means that the property is true in every possible world. Time is undoubtedly one partition of these worlds, but temporal permanence means that the property is true in that world (time), with no information concerning the other possible worlds, and this might happen by pure chance. For example, fingerprints are temporally permanent, but might differ in other possible worlds.

Dependence: In OntoClean [31], the notion of dependence is considered related to concept properties. In this context, dependence permits us to distinguish between *extrinsic* and *intrinsic* properties based on whether they depend on objects other than the one they are ascribed to or not.

In order to establish which of these meta-properties hold, OntoClean is supported by a description logic based system that can help knowledge engineers to assign the meta-properties to concepts and to verify the taxonomic structure on the grounds of the modelling methodology. In this paper we focus our attention on the process of assigning the meta-properties. OntoClean guides knowledge engineers in this process by asking them to answer some questions such as "Does the property carry identity". Knowledge engineers can answer yes, no or unsure, in this latter case more specific questions can be asked, such as "Are instances of the property countable?".

The OntoClean methodology depends on the knowledge engineer's understanding of the ontologies being analysed and can thus be problematic if used to evaluate independently designed ontologies. Moreover, OntoClean does not take into account the structure of concept definitions, as it does not consider the characteristic features (or *attributes*) that might have been used to define concepts. This work proposes an enriched ontology model whose aim is to complement the OntoClean methodology, by providing an additional way to determine meta-properties to concepts. In our proposal we describe concepts in terms of their attributes, which are in turn described not only in terms of their structural features (such as range, domain, cardinality etc.), but also in terms of their meta-properties, which describe the contribution given by the attributes to the concept definition. We describe the enriched ontology model and the meta-properties for attributes in the next sections.

3 Enriched ontology model

The ontology conceptual model [2] we propose comprises *concepts*, *attributes*, *relations*, and *instances*. We do not consider here axioms. Concepts represent the entities of the domain and the tasks we want to model in the ontology. Concepts are described in terms of defining properties, which are represented by associating an *attribute* with either a single value or a set of values. Concepts are organised into an Is-a hierarchy, so that a concept attributes and their values are inherited by subconcepts. Multiple inheritance is permitted, so attributes and their values can be inherited from multiple parents. The values associated with an attribute can be restricted in order to provide a more specific definition of a concept [14].

Attributes can be described in terms of their structural characteristics, such as the concepts that they are defining, their allowed values, the type of the values (string, integer, etc.), and the maximum and minimum values (if attributes are numeric). Attributes can also be described in term of the following meta-properties:

- *Attribute's behaviour over time*: The meta-properties *Mutability, Mutability Frequency, Event Mutability* and *Reversible Mutability* provide a better description of attributes by characterising their behaviour over time, that is, whether they are allowed to change their value during the concept lifetime (*Mutability*); and how often the change occurs (*Mutability Frequency*); whether the change is reversible (*Reversible Mutability*); and what triggers change (*Event Mutability*);
- *Modality*: this meta-property is a qualitative description of the degree of inheritability of a concept property by its subconcepts;
- *Prototypes* and *Exceptions*: the meta-properties *Prototypical* and *Exceptional* aim to describe properties that are prototypical for a concept, that is the properties that obtain for the *prototypical* (from a cognitive viewpoint, following Rosch [21]) instances of a concept. Exceptions are those properties which can be ascribed to a concept although being highly unusual;
- *Inheritance* and *Distinction*: *inherited* meta-properties regard those properties that hold because inherited from an ancestor concept, although they may be overruled in the more specific concept in order to accommodate inheritance with exceptions. *Distinguishing* properties are those that permit us to distinguish among siblings of a same concept. In other words a distinguishing property ϕ is a property such that $\diamond\exists x\ \phi(x) \ \wedge\ \diamond\exists x\ \neg\phi(x)$, that is there is possibly something for which the property ϕ holds, and there is possibly something for which the property does not hold, and these are neither tautological nor vacuous [31]. Distinguishing properties can lead to disjoint concepts in the ontology's taxonomic structure.

---

[2] by conceptual model we mean the knowledge engineer's evolving conception of the domain knowledge. It is the knowledge that actually determines the construction of a formal knowledge base. A conceptual model is an intermediate design construct, a template to begin to constrain and codify human skill, it is neither formal nor directly executable on a computer [16]

These meta-properties provide means to distinguish between *necessary* and *sufficient* conditions for class membership. Indeed, the modality meta-property and those describing the behaviour over time permit the identification of essential (or rigid) properties and necessary properties are those that are essential to all instances of a concept. Prototypical properties are good candidates to identify sufficient conditions, as discussed in Section 3.3.

Relations between concepts are supported by the model as are instances. Finally, the ontology model supports roles. Concepts are also used to represent *roles*, which can be thought of describing the *part played* by a concept in a context, (a more complete discussion on roles is postponed to Section 4.3). Roles are described in terms of their context, and the formal role relationship holds, that is, roles are related to concepts by a 'Role-of' relations.

This ontology model has been used to model a medical condition *Disseminated Intravascular Coagulation* (DIC) [28], whose evolution depends on the changes over time of its symptoms. This ontology model is proving quite promising since it permits physicians to fully capture the changes in the attribute values, how these affect the hierarchy formed by the different types of DIC, and to make explicit most of the modelling assumptions. However, its use is not restricted to medical domains.

This ontology model enriches the traditional model proposed initially by Gruber [6], in that it permits the characterisation of the properties of a concept. From this viewpoint it should be considered more expressive. The solution of adding information characterising concept properties is a controversial one. Indeed, any number of meta-properties could be used to characterise attribute's behaviour. Here we focused our attention on those meta-properties that support formal ontological analysis.

Although we do realise that often it is not true that 'more is better', this work claims that an ontology model which include this type of property characterisation is helpful to deal with ontology heterogeneity problems in two ways. On the one hand the model complements the set of formal ontological properties proposed in [31], and can guide in assigning them to concepts in a way which depends on concept definitions in terms of attributes. This is particularly useful when knowledge engineers need to assign formal properties to ontologies they have not designed.

Additionally, this conceptual model for ontologies facilitates a better understanding of the concepts' semantics. Currently ontology merging is performed by hand based on the expertise of the knowledge engineers and on the ontology documentation. Even in this case the ontology model we propose can prove useful by providing a characterisation of the properties, which can help to identify semantically related terms. The following subsections describe all the meta-properties for attributes except Inheritance and Distinction (which are trivial) in more detail:

3.1 Behaviour over time

The meta-properties which model the behaviour of the attributes over time are:

- *Mutability*, which models the liability of a concept property to change. A property is mutable if it can change during the concept's lifetime;
- *Mutability Frequency*, which models the frequency with which a property can change in a concept description;
- *Event Mutability*, which models the reasons why a property may change;
- *Reversible Mutability*, which models reversible changes of the property.

These meta-properties describe the behaviour of *fluents* over time, where the term *fluent* is borrowed from situation calculus to denote a property of the world that can change over time. Modelling the behaviour of fluents corresponds to modelling the changes in properties that are permitted in a concept's description without changing the essence of the concept.

Fluents are used to characterise time dependency in processes. Hence, here and in [28] we take the view that changes in concept properties can be modelled as *processes* [25].

Describing the behaviour over time also involves distinguishing properties whose change is *reversible* from those whose change is *irreversible*.

Property changes over time are caused either by the natural passage of time or are triggered by specific event occurrences, and so, they need to be represented by a suitable temporal framework that permits us to reason with time and events. In [29] we chose *Event Calculus* [12] to accommodate the representation of changes. Event calculus deals with local event and time periods and provides the ability to reason about change in properties caused by a specific event and also the ability to reason with incomplete information.

We mentioned above that processes model changes in concept properties (which correspond to changes in the values associated with attributes). Processes can be described in terms of their starting and ending points and of the changes that happen in between. We can distinguish between *continuous* and *discrete changes*, the former describing incremental changes that take place continuously while the latter describe changes occurring in discrete steps called *events*. Analogously we can define *continuous properties* as those changing regularly over time, such as the age of a person, versus *discrete properties* which are characterised by an event which causes the property to change. If a property mutability frequency is *regular* (that is it changes regularly), then the process is continuous, if it is *volatile* the process is discrete, and if it changes *once only* in the concept's lifetime, then the process is considered discrete and the triggering event is set equal to *time-point=T*.

Any regular occurrence over time can be, however, expressed in form of an event, since most of the forms of reasoning for continuous properties require discrete approximations. Therefore in the ontology model we present here, continuous properties are thought of as discrete properties where the event triggering the change in property is the passing of time from the instant t to the instant t'. Events are always thought of as *point events*, and we consider *durational events* (events which have a duration) as being a collection of *point events* in which the property whose mutability is modelled by the set of meta-properties hold as long as the event lasts.

3.2 Modality: Weighing the validity of attribute properties

The term modality is used to express the way in which a statement is true or false, which permits us to establish whether a statement constitutes a *necessary truth* and to distinguish necessity from possibility [13]. The term can be extended to qualitatively measure the way in which a statement is true by trying to estimate the number of possible worlds in which such a truth holds. This is the view we take, by denoting the degree of confidence that we can associate with the property holding in a given world with the meta-property *modality.* This notion is analogous to the *rankings* defined by Goldszmidt and Pearl [5]: *Each world is ranked by a non-negative integer* κ *representing the degree of surprise associated with finding such a world* (in which the property does not hold).

Here we use the term modality to denote the degree of surprise in finding a world where the property P holding for a concept C does not hold for one of its subconcepts C'. The additional semantics encompassed in this meta-property is important to account for statements that have different degrees of credibility. Indeed there is a difference in asserting facts such as "Cats are pets" and "All felines are pets", the former is generally more believable than the latter, for which many more counterexamples can be found. The ability to distinguish facts whose truth holds in more or less possible worlds is important in order to find which facts are true in *every* possible world and therefore constitute *necessary truth*, which permits us to establish *rigidity.*

Furthermore, the ability to evaluate the degree of confidence in a property describing a concept is also related to the problem of reasoning with ontologies obtained by merging. In such a case, mismatches can arise if a concept inherits conflicting properties. In order to be able to reason with these conflicts some assumptions have to be made, concerning on how likely it is that a certain property holds. In case of conflict the property degree of credibility can be used to apply some forms of non monotonic reasoning or belief revision. For example, we could rank the possible alternatives on the grounds of the degree of credibility following an approach similar to the one presented in [5].

3.3 Prototypes, exceptions, and concepts

A full understanding of a concept includes not only the set of properties generally recognised as describing a typical instance of the concept, but also the known exceptions. In this way, we partially follow the cognitive view of prototypes and graded structures, which is also reflected by the information modelled in the meta-property *modality.* In this view all cognitive categories show gradients of membership which describe how well a particular subclass fits people's idea or image of the category to which the subclass belong [21]. Prototypes are the subconcepts which best represent a category, while exceptions are those which are considered exceptional although still belonging to the category.

Prototypes show all the sufficient conditions for class membership. For example, let us consider the biological category *mammal*: a *monotreme* (a mammal who does not give birth to live young) is an example of an exception with respect to the property of giving birth to live young. Prototypes depend on the context;

there is no universal prototype but there are several prototypes depending on the context, therefore a prototype for the category *mammal* could be *cat* if the context taken is that of *animals that can play the role of pets* but it is *lion* if the assumed context is *animals that can play the role of circus animals*.

The context is in part determined by the task for which the ontology is built, even in those cases where the ontology is intended to be task neutral, because of the *interaction problem* [1]. Thus, attributes considered prototypical are very likely to differ in ontologies constructed for different tasks.

The ability to distinguish between prototypes and exceptions helps to determine which properties are necessary and sufficient conditions for concept membership. In fact a property which is prototypical and that is also inherited by all the subconcepts becomes a natural candidate for a necessary condition. Prototypes, therefore, permit the identification of the subconcepts that best fit the cognitive category represented by the concept *in the specific context given by the ontology*. On the other hand, by describing which properties are exceptional, we provide a better description of the membership criteria in that it permits us to determine which properties, although rarely holding for that concept, are still possible properties describing the cognitive category.

Prototypes and exceptions can prove useful in dealing with conflicts arising from ontology merging. When no specific information is made available about a concept and it inherits conflicting properties, then we can assume that the prototypical properties hold for it.

In the ontology model presented above the context can be partially described by the roles applicable to the concept for which prototypical and exceptional properties are modelled. Ontologies typically presuppose context and this feature is a major source of difficulty when merging them, since information about context is not always made explicit.

4 Discussion

The ontology model presented in previous section could be implemented in any kind of ontology representation formalisms. In [29] we presented an implementation of the ontology model above in a frame-based representation formalism, and so attributes were described by associating values to slots, and their structural description and meta-properties were modelled by the slot's facets.

By adding the meta-properties to the ontology model, we provide an explicit representation of the attributes' behaviour over time, their prototypicality and exceptionality, and their degree of applicability to subconcepts. This explicit representation may be used to support and complement the OntoClean methodology [31], in that they can help in determining which meta-properties hold for concepts, as we will illustrate in the sub-sections of this section.

Furthermore, the enriched ontology model we propose forces knowledge engineers to make ontological commitments, that is the agreement as to the meaning of the terms used to describe a domain [7] explicit. The extent of knowledge shared depends on the extent of the different agents' ontological commitment made explicit. Real situations are information-rich events, whose context is so rich that, as it has been argued by Searle [23], it can never be fully specified. When dealing with real situations one makes many assumptions about meaning and context [22], and these are rarely formalised. But when dealing with ontologies these assumptions must be formalised since they are part of the ontological commitments that have to be made explicit. Enriching the semantics of the attribute descriptions with things such as the behaviour of attributes over time or how properties are shared by the subconcepts makes some important assumptions explicit.

The enriched semantics helps to recognise and reconcile cases of ontology heterogeneity. By adding information on the attributes we are also aiming to measure the similarity between concepts more precisely and to disambiguate between concepts that *seem* similar while they are not. Indeed, two concepts are to be considered similar if they have similar names, if they are described by similar attributes and *if these attributes show the same behaviour in the concept description* [28].

A possible drawback of enriching the ontology model is that knowledge engineers are required a deeper analysis of a domain. We realise that it makes the process of building an ontology even more time consuming but we believe that a more precise ontological characterisation of the domain at least balances the increased complexity of the task. Indeed, in order to include the attribute meta-properties to the ontology model, knowledge engineers need to have a full understanding not only of the concept they are describing, but also of the context in which the concept is used. Arguably, they need such knowledge if they are to perform the modelling task thoroughly.

The evaluation of the price to pay for this enriched expressiveness and of the kind of reasoning inferences permitted by this model are strictly dependent on the domain and the task at hand. We can imagine that the automatic coalescence of terms might require more sophisticated inferences whose cost we cannot evaluate *a priori*. In some other cases, the simple matching between properties' charactersiations might help in establishing or ruling out the possiblity of semantic relatedness. For example, if two concepts are described by the same properties but with different characterisations, this might indicate that the concepts have been conceptualised differently.

4.1 Identity

The idea of modelling the permitted changes for a property is strictly related to the philosophical notion of *identity*. The meta-properties modelling the behaviour over time are, thus, relevant for establishing the *identity* of concepts [31], since the proposed ontology model addresses the problem of modelling identity when time is involved, namely *identity through change*, which is based on the common sense notion that an individual may remain the same while showing different properties at different times [10]. The knowledge model we propose explicitly distinguishes the properties that can change from those which cannot, and describes the changes in properties that an individual can be subjected to, while still being recognised as an instance of a certain concept. Properties that do not change over time are those that are good candidates to become re-identification criteria.

Prototypical and exceptional properties and the modality meta-properties describing how the property is inherited in the hierarchy can all contribute to determine what are the necessary and sufficient conditions for class membership. Necessary and sufficient conditions are ultimately the conditions that permit us to define the properties constitutive of identity and to distinguish them from those that permit re-identification.

In order to find suitable identity criteria (which permit to identify a concept), knowledge engineer should look at *essential properties*, that is those properties which hold for an individual in every possible circumstance in which the individual exists. It is important to note that essential properties should also be intrinsic if they are to be used to determine identity.

Also inheritance and distinction contribute to identify identity criteria, in that identity criteria have to be looked for among the distinguishing properties.

4.2 Essence and rigidity

Identity through change is also relevant to determine *rigidity*, which derives from the notion of *essence* we defined in Section 2. There we defined a *rigid property* as *a property that is essential to* all *its instances.*

In [29] and in [28] we have related the notion of *rigidity* to those of *time* and *modality*; and, by including in our ontology model a meta-property *modality* and those concerning the behaviour over time, we can precisely identify rigidity in the subset of the set of possible worlds. Indeed, since an ontology defines a vocabulary, we can restrict ourselves to the set of possible worlds which is defined as the set of maximal descriptions obtainable using the vocabulary defined by the ontology [20]. By characterising the rigidity of a property in this subset of possible worlds we aim to provide knowledge engineers the means to reach a better understanding of the *necessary* and *sufficient* conditions for the class membership. However, this does not mean that the rigidity of a property depends on any account of whether the property is used to determine class membership or not. That is, the final aim is to try to separate the properties constitutive of identity from those that permit re-identification. Under the assumption of restricting the discourse to this set of possible worlds, *rigid properties* are those properties which are inherited by all subconcepts, and thus which have a certain degree of belief associated with the meta-property *modality* and that cannot change in time.

It is important to note that, although in [29] we have modelled this information as a facet which can take value in the set {*All, Almost all, Most, Possible, A Few, Almost none, None*}, the choice of such a set is totally arbitrary, and it is intended only as an example of a possible way to represent this meta-property. Alternatively, knowledge engineers should be able to associate with this meta-property either a probability value, if they know the probability with which the property is inherited by subconcepts, or a degree of belief (such as a κ-value, as in [5], which depends on an ϵ whose value can be changed according to the knowledge available, thus causing the κ function to change), if the probability function is not available.

4.3 Roles dependence on identity and rigidity

Rigidity is not only central in order to distinguish necessary truth but also to distinguish *roles* from concepts.

A definition of role that makes use of the formal meta-properties and includes also the definition given by Sowa [24] is provided by Guarino and Welty. In [31] they define a role as: ‘ *the properties expressing the* part played *by one entity in an event, often exemplifying a particular relationship between two or more entities. All roles are* anti-rigid *and* dependent... *A property* ϕ *is said to be anti-rigid if it is not essential to* all *its instances, i.e.* $\Box(\forall x,\ t\phi(x,\ t) \rightarrow \Diamond\exists\ t'\ \neg\ \phi(x,t'))$... *A property* ϕ *is* (externally) dependent *on a property* ψ *if, for all its instances x, necessarily some instance of* ψ *must exist, which is not a part nor a constituent of x, i.e.* $\forall x\Box(\phi(x) \rightarrow \exists y\psi(y) \wedge \neg P(y,x) \wedge \neg C(y,x))$ ’, where $P(y,x)$ denotes that y is a *part* of x while $C(y,x)$ denotes that y is a *constituent* of x. In other words a concept is a role if its individuals stand in relation to other individuals, and they can enter or leave the extension of the concept without losing their identity. From this definition it emerges that the ability to recognise whether rigidity holds for some property ϕ is essential in order to distinguish whether ϕ is a role.

Roles may be ‘naturally’ determined when social context is taken into account, and the social context determines the way in which a role is acquired and relinquished. For example, the role of *President of the country* is relinquished differently depending on the context provided by the country. So, for example, in Italy the role may be acquired and relinquished only once in the lifetime of an individual, whereas if the country is the United Sates, the role can be acquired and relinquished twice, because a president can be re-elected. Social conventions may also determine that once a role is acquired it cannot be relinquished at all. For example, the role *Priest* in a catholic context is relinquished only with the death of the person playing the role. The ability to distinguish roles gives also a deeper understanding of the possible contexts in which a concept can be used. Recognising a role can be equivalent to defining a context, and the notion of context is the basis on which prototypes and exceptions are defined.

In [26] Steimann compares the different characteristics that have been associated in the literature with roles. From this comparison it emerges that the notion of role is inherently temporal, and roles are acquired and relinquished dependent on either time or a specific event. For example the object *person* acquires the role *teenager* if the person is between 13 and 19 years old, whereas a person becomes *student* when they enroll for a degree course. Moreover, from the list of features in [26] it follows that many of the characteristics of roles are time or event related, such as: an object may acquire and abandon roles dynamically, may play different roles simultaneously, or may play the same role several time, simultaneously, and the sequence in which roles may be acquired and relinquished can be subjected to restrictions. Indeed, what distinguishes a role from a concept, in the modelling process, is that a role holds during a specific span of time in which some property holds. For example, the role 'Student' is applicable only if the property of being registered to a university holds. Therefore, the meta-properties that model the behaviour over time permits the representation of the acquisition and relinquishment of a role.

For the aforementioned reasons, ways of representing roles must be supported by some kind of explicit representation of time and events. Indeed the proposed model provides a way to model roles as fluents; moreover, by modelling the reason for which a property change, we provide knowledge engineers the ability to model the events that constrain the acquisition or the relinquishment of a role.

5 Conclusions

Sharing ontologies independently developed is a burning issue that needs to be resolved. This paper presents a set of meta-properties describing concept's characteristic features (attributes) that can be used to support both the process of building correct ontologies (by complementing and supporting the formal ontological analysis performed by the OntoClean methodology [31]) and the disambiguation of cases of ontology heterogeneity. Formal ontological analysis is usually demanding to perform and we believe that the set of meta-properties for attributes we propose can support knowledge engineers in determining the meta-properties holding for the concepts by forcing them to make the ontological commitments explicit.

The meta-properties we propose, namely Mutability, Mutability Frequency, Reversible Mutability, Event Mutability, Modality, Prototypicality, Exceptionality, Inheritance and Distinction encompass semantic information aiming to characterise the behaviour of properties in the concept description. We have argued that such a precise characterisation can help to disambiguate among concepts that only seem similar, and in turn can support mappings across the structure of multiple shared ontologies that we have devised as alternative to the current approaches to knowledge sharing. We claim that this characterisation of the concept properties is also very important in order to provide a precise specification of the semantics of the concepts. Such characterisation is essential if we want to perform a formal ontological analysis, in which knowledge engineers can precisely determine which formal tools they can use in order to build an ontology which has a taxonomy that is clean and not very tangled.

The novelty of this characterisation is that it explicitly represents the behaviour of attributes over time by describing the permitted changes in a property used to describe a concept. It also explicitly represents the class membership mechanism by associating with each attribute (represented in a slot) a qualitative quantifier representing how properties are inherited by subconcepts. Finally, the model does not only describe the prototypical properties holding for a concept but also the exceptional ones. By providing this explicit characterisation, we are asking knowledge engineers to make more hidden assumptions explicit, thus providing a better understanding not only of the domain in general, but also of the role a concept plays in describing a specific domain.

Acknowledgements

We wish to express our gratitude to Asunción Gómez-Pérez for the many discussions and valuable comments on the PhD thesis from which this paper is derived. We have also benefitted from the discussion with Mariano Fernández López and we would like to thank him for his thought provoking comments. The PhD presented in this paper was funded by BT plc.

References

[1] T. Bylander and B. Chandrasekaran. Generic tasks in knowledge-based reasoning: The right level of abstraction for knowledge acquisition. In B. gaines and J. Boose, editors, *Knowledge acquisition for knowledge bases*, volume 1, pages 65–77. Academic Press, London, 1988.

[2] M. Fernández-López, A. Gómez-Pérez, and N. Guarino. The methontology & ontoClean merge. Technical report, OntoWeb special interest group on Enterprise-standards Ontology Environments, 2001.

[3] M. Fernández-López, A. Gómez-Pérez, A. Pazos-Sierra, and J. Pazos-Sierra. Building a chemical ontology using METHONTOLOGY and the ontology design environment. *IEEE Intelligent Systems and their applications*, January/February:37–46, 1999.

[4] N. Fridman Noy and M.A. Musen. SMART: Automated support for ontology merging and alignment. In *Proceedings of the 12th Workshop on Knowledge Acquisition, Modeling and Management (KAW)*, Banff, Alberta, Canada, 1999. University of Calgary.

[5] M. Goldszmidt and J. Pearl. Qualitative probabilisties for default reasoning, belief revision, and causal modelling. *Artificial Intelligence*, 84(1-2):57–112, 1996.

[6] T. R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199–220, 1993.

[7] N. Guarino. Formal ontologies and information systems. In N. Guarino, editor, *Proceedings of FOIS'98*, Amsterdam, 1998. IOS Press.

[8] N. Guarino, M. Carrara, and P. Giaretta. An ontology of meta-level-categories. In *Principles of Knowledge representation and reasoning: Proceedings of the fourth international conference (KR94)*, pages 270–280, San Mateo, CA, 1994. Morgan Kaufmann.

[9] E. Hirsch. *The concept of identity.* Oxford University Press, New York, 1982.

[10] I. Kant. *Critique of pure reason.* St. Martin's press, New York, 1965. Translation by N. Kemp Smith from *Kritik der reinen Vernunft*, 1787.

[11] K. Knight and S. Luk. Building a large knowledge base for machine translation. In *Proceedings of the American Association of Artificial Intelligence Conference, AAAI-94*, Seattle, WA, 1994.

[12] R. Kowalski and M. Sergot. A logic-based calculus of events. *New Generation Computing*, 4:67–95, 1986.

[13] S.A. Kripke. *Naming and necessity.* Harvard University Press, Cambridge, Massachusetts, USA, 1980.

[14] O. Lassila and D. McGuinness. The role of frame-based representation on the semantic web. *Electronic Transactions on Artificial Intelligence (ETAI) Journal: area The Semantic Web*, To appear, 2001.

[15] E.J. Lowe. *Kinds of being. A study of individuation, identity and the logic of sortal terms.* Basil Blackwell, Oxford, UK, 1989.

[16] G.F. Luger. *Artificial intelligence. Structures and strategies for complex problem solving.* Addison Wesley-Pearson Education, Harlow, England, fourth edition, 2002.

[17] D.L. McGuinness. Conceptual modelling for distributed ontology environments. In B. Ganter and G.W. Mineau, editors, *Proceedings of the Eighth International Conference on Conceptual Structures Logical, Linguistic, and Computational Issues (ICCS 2000)*, volume LNAI 1867, 2000.

[18] D.L. McGuinness, R.E. Fikes, J. Rice, and S. Wilder. An environment for merging and testing large ontologies. In A.G. Cohn, F. Giunchiglia, and B. Selman, editors, *Principles of Knowledge Representation and Reasoning. Proceedings of the seventh international conference (KR'2000)*, pages 483–493, San Francisco, CA, 2000. Morgan Kaufmann.

[19] G.A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. Miller. Introduction to wordnet: An on line lexical database. Technical report, Cognitive Science Laboratory, Princeton University, 1993.

[20] A. Plantiga. *The nature of necessity.* Clarendon Library of logic and philosophy. Clarendon Press, New York, 1989.

[21] E.H. Rosch. Cognitive representations of semantic categories. *Journal of Experimental Psychology: General*, 104:192–233, 1975.

[22] E.H. Rosch. Reclaiming concepts. *Journal of Consciousness Studies*, 6(11-12):61–77, 1999.

[23] J.R. Searle. *Intentionality.* Cambridge University Press, Cambridge, 1983.

[24] J.F. Sowa. *Conceptual Structures: Information Processing in Mind and Machine.* Addison-Wesley, Reading, MA, 1984.

[25] J.F. Sowa. *Knowledge Representation: Logical, Philosophical, and Computational Foundations.* Brooks Cole Publishing Co., Pacific Grove, CA, 2000.

[26] F. Steimann. On the representation of roles in object-oriented and conceptual modelling. *Data and Knowledge Engineering*, 35:83–106, 2000.

[27] R. Studer, V.R. Benjamins, and D. Fensel. Knowledge engineering, principles and methods. *Data and Knowledge Engineering*, 25(1-2):161–197, 1998.

[28] V. Tamma. *An ontology model supporting multiple ontologies for knowledge sharing.* PhD thesis, University of Liverpool, 2002.

[29] V.A.M. Tamma and T.J.M. Bench-Capon. An enriched knowledge model for formal ontological analysis. In C. Welty and B. Smith, editors, *Proceedings of the international conference on formal ontology and information systems (FOIS'01)*, New York, 2001. ACM press.

[30] P.R.S. Visser, D.M. Jones, T.J.M. Bench-Capon, and M.J.R. Shave. Assessing heterogeneity by classifying ontology mismatches. In N. Guarino, editor, *Formal Ontology in Information Systems. Proceedings FOIS'98, Trento, Italy*, pages 148–182. IOS Press, 1998.

[31] C. Welty and N. Guarino. Supporting ontological analysis of taxonomical relationships. *Data and knowledge engineering*, 39(1):51–74, 2001.

[32] D. Wiggins. *Identity and Spatio-Temporal continuity.* Basil Blackwell, Oxford, 1967.