Applied Ontology 0 (2007) 1
IOS Press

Representing and Reasoning over a Taxonomy of Part-Whole Relations

C. Maria Keet * and Alessandro Artale
KRDB Research Centre, Faculty of Computer Science, Free University of Bozen-Bolzano, Italy
E-mail: {keet, artale}@inf.unibz.it

Abstract. Many types of part-whole relations have been proposed in the literature to aid the conceptual modeller to choose the most appropriate type, but many of those relations lack a formal specification to give clear and unambiguous semantics to them. To remedy this, a formal taxonomy of types of mereological and meronymic part-whole relations is presented that distinguishes between transitive and intransitive relations and the kind of entity types that are related. The demand to use it effectively brings afore new requirements for automated reasoning over a hierarchy of relations. To ensure logically and ontologically correct inferencing over both the class and role hierarchy, the new reasoning service *RBox compatibility* for Description Logics reasoners is introduced. The proposed combination of formal semantics and the new reasoning service will improve the representation of the application domain when using part-whole relations in conceptual models and ontologies.

Keywords: part-whole relation, mereology, conceptual modelling, Description Logics reasoning

1. Introduction

Many ontological and cognitive aspects of the part-whole relation have been discussed (e.g. Artale et al. (1996); Bittner and Donnelly (2005); Gerstl and Pribbenow (1995); Odell (1998); Shanks et al. (2004); Varzi (2004, 2006a); Vieu and Aurnague (2005); Winston et al. (1987)) and proposals to include this relation to conceptual modelling and knowledge representation languages have been suggested, such as (Barbier et al. (2003); Guizzardi (2005); Motschnig-Pitrik and Kaasbøll (1999)) for UML, (Bittner and Donnelly (2005); Lambrix and Padgham (2000); Sattler (1995); Schulz and Hahn (2000)) for Description Logics, Shanks et al. (2004) for ER, and Keet (2006a) for ORM. These communities introduced different types of part-whole relations, which sometimes are different types of mereological parthood relations, whereas others appear to be motivated by cognitive and linguistic use of ‘part’ (meronymy). To represent the Universe of Discourse as accurately as possible, it is an imperative to be able to *identify* and *use* the appropriate part-whole relation that is closest to the real world it is supposed to represent. Being more precise during the conceptual analysis stage by taking into account ontological notions of part-whole relations will improve the quality of the conceptual model, thereby reducing errors in software development and saving resources during the testing phase, which also result in better application software. Other advantages obtained by representing part-whole relations more precisely range from the implementation stage by supporting product quality assurance and fault identification of e.g. a specific part-component of a device – as one can represent the parts properly – to reasoning over conceptual models (and domain ontologies) by having distinguished transitive and intransitive part-whole relations, such as classifying hierarchies, checking concept satisfiability, and discovering new sub/super relations.

Our contribution is twofold. First, we aim at clarifying the semantics of part-whole relations by considering the recent results of philosophical analyses on both mereological theories and foundational ontologies. Second, we propose a new consistency check for automated reasoning services of description logics, which uses the introduced semantics for part-whole relations. We thereby aim at contributing to theoretical foundations of conceptual modelling as well as facilitating usage of part-whole relations in software systems.

*Corresponding author: C. Maria Keet, KRDB Research Centre, Faculty of Computer Science, Free University of Bozen-Bolzano, Italy. Email: keet@inf.unibz.it. Phone: +39 04710 16128, fax: +39 04710 16009

To clarify the semantics of part-whole relations, this paper proposes a basic formal taxonomy of part-whole relations that includes both mereological and meronymic part-whole relations. The aim is to unambiguously specify the hierarchical relations between the various part-whole relations introduced in the literature. The novelty of the approach taken is that the taxonomy is not merely a description of part-whole relations tailored to one specific usage only, but it uses well-known foundational ontology aspects and an implementation-independent, formal characterisation, to enable portability across different conceptual modelling languages and application scenarios. The obtained taxonomy and distinctions made will be justified by formally defining them with First Order Logic (FOL) formulæ. To effectively use the part-whole relations (and, in general, any hierarchy of relations), we propose a new reasoning service for automated reasoners, which is called *RBox compatibility*. This new reasoning service ensures that a hierarchy of relations is not only logically correct (as is currently the case with the extant reasoner software tools) but that is also *ontologically correct* by checking a correspondence between the role hierarchy, the domain and range restrictions for roles, and the class hierarchy. This enables earlier error detection and meaningful user feedback during conceptual model and domain ontology development.

In Section 2 we discuss related work on part-whole relations and disambiguate the types of part-whole relations with a basic formal taxonomy in Section 3. Languages and tools are briefly considered in Section 4, after which we proceed to automated reasoning over part-whole relations (and reasoning over hierarchies of relations in general) in Section 5, where the new reasoning service for automated reasoners is introduced. We close with conclusions and further research in Section 6.

2. Distinct Part-whole Relations: Related Works

Mereology is a sub-discipline in philosophy that concerns the formal ontological investigation of the part-whole relation. Meronymy studies part-whole relations from a linguistics and cognitive science perspective. There are meronymic relations that are not mereological. That is, there are usages of 'part of' in natural language and conceptual modelling that do not share the same properties as the $part\_of$ relation in mereology. The main semantic difference revolves around the (in)transitivity of the part-whole relation. To appreciate the distinction, we first introduce some basic aspects of mereology (see Guizzardi (2005); Keet (2006b); Varzi (2004) for a more comprehensive introduction and overview) and subsequently summarise contributions from meronymy and research into conceptual modelling from the perspective of engineering usefulness. Based on this analysis, we propose a taxonomy of the various of part-whole relations in §3.

2.1. Ground Mereology

The lowest common denominator concerning theories of parthood is called *Ground Mereology*, which states that *part_of* is a relation capturing a partial order that is reflexive (1), antisymmetric (2), and transitive (3); all other versions share at least these constraints. This, however, does not mean (1-3) are uncontested; in particular transitivity of *part_of* receives attention, to which we return later.

$$\forall x(part\_of(x,x)) \tag{1}$$

$$\forall x,y((part\_of(x,y) \land part\_of(y,x)) \rightarrow x=y) \tag{2}$$

$$\forall x,y,z((part\_of(x,y) \land part\_of(y,z)) \rightarrow part\_of(x,z)) \tag{3}$$

Starting with these three basic formulæ that take *part_of* as a primitive relation, several other mereological predicates can be built. A common one is the definition of *proper* part (4), which is asymmetric, irreflexive and transitive. Note that in some mereological theories, the proper parthood is taken as primitive relation.

$$\forall x,y(proper\_part\_of(x,y) \triangleq part\_of(x,y) \land \neg part\_of(y,x)) \tag{4}$$

The relatively simple Ground Mereology is often extended with weak or strong supplementation, some version of fusion, and atomicity, which we do not need for the current scope. Here, we focus on the assertion that all mereological parthood relations are transitive, which is regularly discussed (e.g. Johansson (2004); Odell (1998); Varzi (2006b)). Contrary to the straightforward mereological theories, extensions and modifications have been proposed to a) accommodate different types of part-whole relations for conceptual modelling and b) permit intransitivity in some cases of part-whole relations. But these two motivations for usage of part-whole relations go against the mereological theories that claim there is just one, transitive parthood relation, which means something else is going on. We look into this issue in more detail in the next Section.

2.2. *Part-whole relations in meronymy and conceptual modelling*

As was mentioned briefly in the previous Section, an important distinction exists between the mereological *part_of* relation and *meronymic* part-whole relations: the latter is not necessarily transitive and may not fit within any mereological theory. That is, in our communication we use the *term* "part of", but ontologically one is not part of the other. For instance, Odell (1998)'s "member-bunch" part-whole relation (subsequently referred to as *member_of*) is a meronymic part-whole relation. For instance, Del Piero, an instance of Football player, is member of the Juventus football team, and he is also member of the Juventus Torino club, which in turn is member of the Italian football clubs federation Federcalcio, but Del Piero is not a member of Federcalcio. A different case of intransitivity is due to the mixed use of different part-whole relations. Consider the canonical example of hand-musician-orchestra, where some instance of Musician is part of some instance of Orchestra and a musician's Hand is part of the instance of Musician, but it is false that that musician's hand (or any other musician's body part) is part of that orchestra. This intransitivity of 'part-of' with hand-musician-orchestra generalises also to the class-level, because no structural part (Hand) of a whole (Musician) propagates to be also a part of *any* collective whole (Orchestra); that is, here we relate different kinds of things, which is the underlying cause of intransitivity of 'part of'.

We now mention the efforts in the literature that have a list or a taxonomy of part-whole relations to sort out different usages of both the mereological parthood relation and meronymic part-whole relations. The first proposal that introduced different types of part-whole relations was motivated by the linguistic use of 'part', i.e. meronymy, and was made by Winston, Chaffin and Herrmann (WCH) (Winston et al. (1987)). Several successive articles analysed the WCH taxonomy and their modelling considerations (e.g. Artale et al. (1996); Gerstl and Pribbenow (1995); Guizzardi (2005); Odell (1998)). For instance, Gerstl and Pribbenow (1995) prefer a "common-sense theory of part-whole relations" instead, which is motivated by differences in the compositional structure of the whole and they propose three different types: i) a homogenous mass with quantities, ii) a collection of uniform elements, and iii) a complex of heterogeneous components. They add a notion of "different views on the entities" in the sense that from one viewpoint a complex may be just a collection, which can be a source of problems for semantic interoperability. Although they provide linguistic motivations by showing various examples supporting the existence of these three types of meronymic relations, they do not provide ontologically rigorous definitions for them. Sattler (1995) constrained her list of part-whole relations to types of "direct parthood", which was motivated by the constraint that Description Logic (DL) roles could not be transitive due to limitations on the language ($\mathcal{ALC}$) and role hierarchies were not yet supported. The same problem motivated Schulz and Hahn (2000) to develop so-called SEP triples as engineering solution to circumvent the intransitive DL roles by representing the partonomy as a taxonomy[1]. Observe that both role hierarchies and role transitivity are supported in the more recent expressive DLs $\mathcal{SROIQ}$ and $\mathcal{DLR}_\mu$, as well as (ir)reflexivity and (a)symmetry, while reflexive antisymmetry is still an open issue (Horrocks et al. (2006)), thereby making them the two formal knowledge representation languages known to be within the decidable fragment of FOL to support most of the aforementioned properties of parthood and proper parthood.

Odell (1998) proposes a list of six types of part-whole relations, providing descriptions and examples. This is summarised in Table 1 and criticised by, among others, Guizzardi (2005); Keet (2006b);

[1]The three core items of a SEP triples are the **S**tructure-concept node that subsumes one (anatomical) entity, called **E**ntity node, and the parts of that entity (the **P**-node). The *is_a* hierarchy is then built up by relating the P-node of a whole concept D to the S-node of the part C, where in turn the P-node of C is linked to the S-node of C's part.

Table 1

Odell's types of part-whole relations, in the first column, and our explanation in the second column.

Type of part-of	Explanation
component – integral object	Discrete type of part-whole relation, with atoms
material – object	Constitution of objects
portion – object	a) some amount of matter is part of the whole, and b) scale-based partonomic relations
place – area	Where part-place cannot be separated from the whole-area
member – bunch	Whole bunch is generally denoted with a collective noun and its members can change over time
member – partnership	Like member-bunch, but changing a member does destroy the whole

Motschnig-Pitrik and Kaasbøll (1999). Of these types, only component-integral object and place-area meet mereological parthood criteria, whereas the underspecified portion-object and member-bunch are meronymic part-whole relations. Later, Motschnig-Pitrik and Kaasbøll (1999) reintroduced a small informal taxonomy that has all relations subsumed by a generic "meronymic relationship" that subsumes "(core) part-of", which groups together mereological parthood with portion/mass, process/subprocess and place/area as well, then there is the *member_of* relation, a *made_of* for constitution, and a "noun-feature/activity" relation, which is better known in ontology under the name *participates_in* (see e.g. Masolo et al. (2003); Smith et al. (2005)). Opdahl et al. (2001); Barbier et al. (2003) distinguish between three kinds of properties of "whole-part relationships" – "primary" (asymmetry, antisymmetry, and emergent property), "secondary" (transitivity or intransitivity, shareability, mutability, and separability), and "consequent" (ownership, propagation of operations, and abstraction)[2] – and provide a characterisation in UML's Object Constraint Language (OCL). The differentiation of kinds of properties exhibits many curious features; we discuss two. First, we mention the prioritization of, for conceptual data modelling, more and less important requirements to represent part-whole relations, which blurs the distinction between requirements for one object-oriented language and properties of the part-whole relation. The authors intermingle object-oriented application-driven modelling considerations – such as software object creation & destruction, encapsulation, and ownership – and the (non-)representation of attributes with the Bunge Wand Weber (BWW) "ontology used as a model" the authors had taken originally. BWW does make a distinction between existence of properties in reality and their representations (see also Guizzardi (2005)). Put differently, for Opdahl et al. (2001) (p393), a part-whole relation should not be represented in a UML class diagram if some of the classes' "resultant and emergent" attributes are not represented, even though the part-whole relation exists in reality between the affected classes. Second, primary properties of part-whole relations do not correspond to combinations of properties of the $part\_of$ and $proper\_part\_of$ relations in mereology, thereby setting aside a great deal of ontological investigation and usable results (see Section 3). Moreover, eventually they have removed transitivity from the list of primary properties of part-whole relations —for conceptual data modelling—, based on a single confused example about intransitivity between body parts, person and research group. The weakness of their argument can also be traced back to the problem of fancying to represent some class attributes in one's UML class diagram, or not, versus properties of universals and their relations, and of mixing different types of part-whole relations and relata. Representing part-whole relations for actual domains indeed is not an easy task, but instead of avoiding representational issues and "thereby confirm" (Opdahl et al. (2001)) demotion of transitivity to secondary property, one should try to resolve the problematic aspects.

In contrast to these earlier attempts, Guizzardi (2005) provides criteria for two types of part-whole relations and distinguishes between the mereological *part_of* relation and three other part-whole relations, being *sub_quantity_of* to relate amounts of matter, *sub_collection_of* that actually represents a set-subset relation (e.g., a group of seminar attendees where there is a subset of the attendees with nationality Dutch),

[2] "a *primary* characteristic is a *necessary* condition for a relationship in an OO-model to be a WP relationship." Secondary properties do not hold for all OO-model whole-part relations, and a consequential characteristic is "a logical (or natural) consequence of one or more of the primary ones" (Opdahl et al. (2001)).

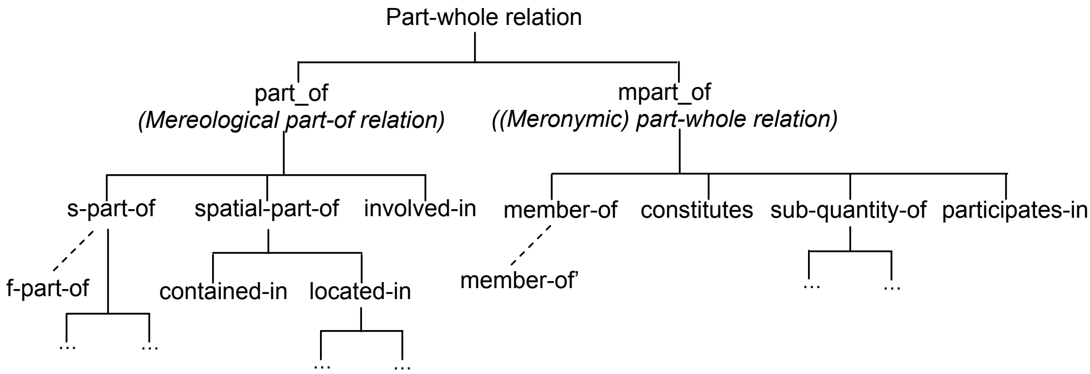

Fig. 1. Taxonomy of basic mereological and meronymic part-of relations. s-part-of = structural part-of; f-part-of = functional part-of. Dashed lines indicate that the subtype has additional constraints on the participation of the entity types; ellipses indicate several possible finer-grained extensions to the basic part-whole relations.

and *member_of*. Although Guizzardi (2005) describes clearly for each type of relation the graphical UML notation and their formal properties, he underspecifies the stereotypes (or categories) of the entity types involved in a part-whole relation and omits both parthood between processes and mereotopological relations (parts and connection/space).

How to figure out differences between parthood and parthood-like relations based on a FOL formalization, and how to structure the proposed part-whole relations in a clear manner, is the topic of the next Section. In particular, it will become clear that in case of different types of part-whole relations, different *categories* of entity types (universals) will be related. Provided one makes these required distinctions, transitivity still holds for parthood and can be easily distinguished from intransitive part-whole relations.

3. A Formal Taxonomy of Part-Whole Relations

To provide an unambiguous way of dealing with part-whole relations in conceptual data models (and domain ontologies), we propose a formal taxonomy of part-whole relations. The taxonomy structures the various part-whole relations, as introduced and/or discussed by e.g. Artale et al. (1996); Barbier et al. (2003); Bittner and Donnelly (2005); Gerstl and Pribbenow (1995); Guizzardi (2005); Johansson (2004); Keet (2006a); Motschnig-Pitrik and Kaasbøll (1999); Odell (1998); Sattler (1995); Schulz et al. (2006); Rector et al. (2006); Smith et al. (2005); Tan et al. (2003); Vieu and Aurnague (2005); Winston et al. (1987), by taking into account ontological distinctions[3]. The proposed taxonomy is depicted in Fig.1 and is explained in the remainder of this Section, including formal definitions for the leaf types of part-whole relations. The taxonomy is a balance between typing ontologically-motivated relations useful for conceptual modelling up to the minimum level of distinctions to gain benefit from specifying part-whole relations more precisely, yet avoiding ontological exuberance that would deter conceptual modellers from using it in practice during the conceptual analysis stage; where applicable, current cut-off points will be justified in the explanation below.

3.1. Overview and preliminaries

The first principal distinction in the taxonomy is made between transitive and intransitive part-whole relations. The prime reason why this ontological distinction exists has to do with transitivity of the mereological parthood relation versus other part-whole relations. Successive distinctions between the relations

[3]The taxonomy does not deal with other facets of parthood relations, such as intra-part relations, the inverse relation *has_part*, and if the parts together are *all* parts that make up the whole; these aspects are beyond the scope of this article. For a discussion and various options to address such issues, see e.g. Barbier et al. (2003); Guizzardi (2005); Lambrix and Padgham (2000); Motschnig-Pitrik and Kaasbøll (1999); Opdahl et al. (2001).

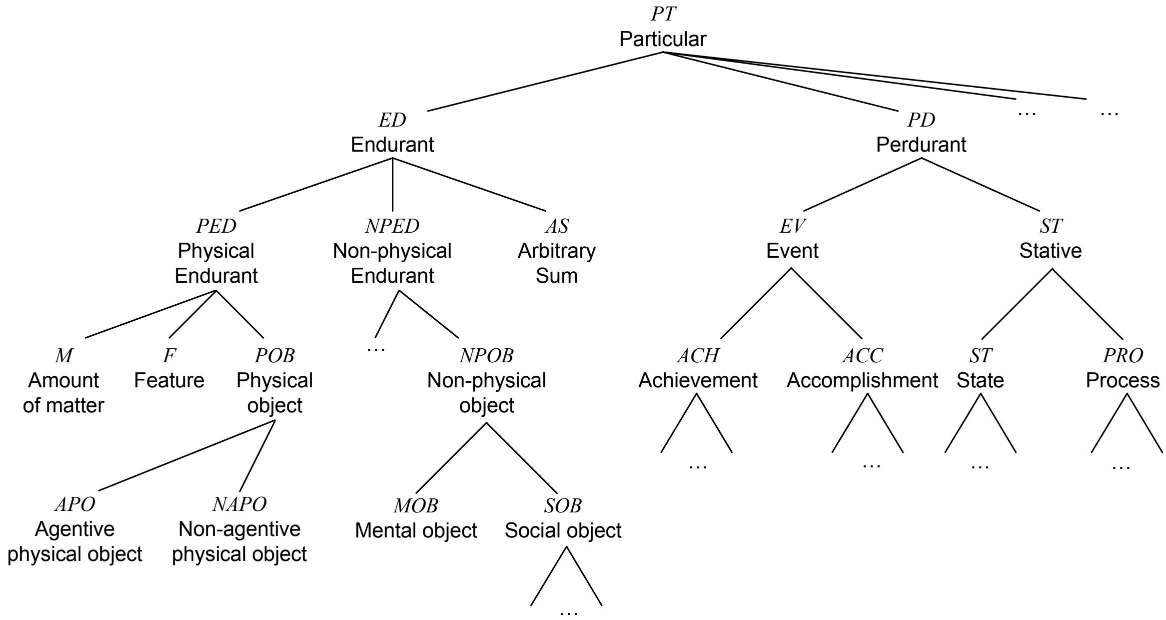

Fig. 2. Graphical rendering of a section of the foundational ontology DOLCE. In colloquial natural language communication, endurant roughly maps to entity types and perdurant to processes or (objectified) relations/associations in conceptual models.

are made based on the categories of the entity types participating in the relation—also called relata, domain and range restriction, stereotypes of object types. The types of part-whole relations are disjoint. Further distinctions can be made on finer-grained categories of participating entity types and on other properties of the relation, such as existential dependence of the part on the whole or vice versa.

To be able to talk about the categories of the entity types involved in part-whole relations, we have to consider foundational ontologies from which we can borrow several top-level categories. Of the several extant foundational ontologies, such as DOLCE (Masolo et al. (2003)), BFO[4], OCHRE (Schneider (2003)), SUMO, and GFO (Herre and Heller (2006)), we chose DOLCE, the Descriptive Ontology for Linguistic and Cognitive Engineering, because it is the most comprehensively formalised one, has a mapping to OWL[5], and is used across several subject domains for development of ontology-driven information systems[6]. DOLCE includes the afore-mentioned Ground Mereology (as it uses Atomic General Extensional Mereology for atemporal parthood), and the definitions introduced in this paper are fully compatible with DOLCE's formal characterisation. A portion of the DOLCE foundational ontology that is relevant for the purposes of this paper is depicted in Fig.2 (directly subsumed universals are disjoint (Masolo et al., 2003)). In the remainder of this section, we will use the abbreviated names of Fig.2, such as ED for *end*urant and POB for *p*hysical *ob*ject, after their first introduction in the text.

To avoid overloading terms, the non-mereological part-whole relation is labelled with $mpart\_of$. This part-whole relation is included in the taxonomy mainly for structuring purposes and is not intended for general use. Observe also that $mpart\_of$ is '*non*-transitive', that is, neither transitive nor intransitive[7], because it is not the case that intransitivity holds for *all* its subsuming part-whole relations with *all* their related instances. Thus, its semantics differs from the mereological $part\_of$, which is assumed transitive. Because of this distinction, the top relation in the taxonomy, $part\text{-}whole\text{-}relation$, is necessarily also non-transitive. Both $part\_of$ and $mpart\_of$ inherit from $part\text{-}whole\text{-}relation$ the typing of the relata (domain & range restriction), assumed to be the DOLCE's top-category Particular (PT). For explanatory

[4]http://www.ifomis.uni-saarland.de/bfo/home.php

[5]Web Ontology Language, which is based on description logics. http://www.w3.org/TR/owl-features/.

[6]See for an overview: http://www.loa-cnr.it/DOLCE.html.

[7]non-transitive is not a new property but a short-hand notation for absence of declaring transitivity or intransitivity, which we use when it is known that the relation is neither transitive for all cases nor intransitive

purposes only, subsequent definitions in the meronymic branch contain the subscript "$_{it}$" to denote an *in*transitive relation ($\forall x, y, z(R(x, y) \wedge R(y, z) \rightarrow \neg R(x, z))$) or "$_n$" for *n*on-transitive.

3.2. *Leaf types of part-whole relations*

Here we describe the meronymic leaf types first and the mereological parthood relations afterward.

The $member\_of$ relation (in the literature also called member-bunch, collection, collective or aggregation) belongs to the meronymic branch. The whole-side of the relation is generally denoted with a *collective noun*, i.e. a social entity with aggregations like Herd or Orchestra. The part-side are physical entities or the role they play, such as Sheep and Musician, respectively. Given that roles are dependent on the bearer, a physical object in its turn, we formalise $member\_of$ with the following definition.

$$\forall x, y(member\_of_n(x, y) \triangleq mpart\_of(x, y) \wedge (POB(x) \vee SOB(x)) \wedge SOB(y)) \quad (5)$$

This $member\_of$ constrains the 'part' side of the relata to either physical objects (POB) or their roles as non-physical social objects (SOB) (a simplification of the work by Masolo et al. (2004)), while the whole the parts are aggregated into are constrained to be social objects SOB. With (5), it is easy to represent Odell (1998)'s "member-partnership" example relation for Husband and Wife in a partnership Marriage. The only addition is that the whole is existentially dependent on the part and vice versa: let ε be existential dependence (see also Guizzardi (2005)), then we can add a subtype *member_of$'$* which has "$\varepsilon(x, y) \wedge \varepsilon(y, x)$" added to definition (5).

The *material-object* relation, categorised as part-whole relation by Winston et al. (1987); Sattler (1995); Gerstl and Pribbenow (1995); Odell (1998), corresponds ontologically to *constitution* where a POB is related to an amount of matter (M) it is made of (see section 3.3.3 in Masolo et al., 2003, for detail and justification)—e.g. Statue and the Marble it is constituted of. Amounts of matter are generally denoted with *mass nouns* that are not countable. In natural language, the inverse $constituted\_of$ is used more often, where the whole is constituted of its material-parts.

$$\forall x, y(constitutes_{it}(x, y) \equiv\ constituted\_of_{it}(y, x) \triangleq mpart\_of(x, y) \wedge POB(y) \wedge M(x)) \quad (6)$$

The *sub-quantity-of* relation, also called quantity-mass or portion-object (e.g. Sattler (1995); Odell (1998); Guizzardi (2005)), relates a smaller part-amount of matter (M) to a whole-matter (M), where the amounts of matter are either the same type of stuff, e.g., a glass of Wine—in this case we require an additional measure for its quantity like glass & bottle or millilitre & litre—or, similar to Guizzardi (2005), the part-M is a different type of matter than the whole-M—e.g. sub_quantity_of(Salt, SeaWater). These examples are ontologically distinct, giving rise, in the first case, to a transitive $sub\_quantity\_of'$ relation and to an intransitive $sub\_quantity\_of''$ relation in the second case. In addition, in the second case, a part-M may have undergone a chemical reaction in the whole-M and, strictly speaking, not exist anymore compared to the part-M in isolation or existing in a different state (e.g., the molecules have released a hydrogen atom upon dissolving in the whole-M). Further, specific quantities may not matter for representing basic knowledge in ontologies, such as representing sub_quantity_of(Alcohol, Wine), but this is needed for conceptual data models where recording data on percentages of alcohol in beverages is needed. Therefore, (7) has the lowest common denominator, but it can benefit from further disambiguation.

$$\forall x, y(sub\_quantity\_of_n(x, y) \triangleq mpart\_of(x, y) \wedge M(x) \wedge M(y)) \quad (7)$$

The last meronymic relation is $participates\_in$, a noun-feature/activity in linguistics (Motschnig-Pitrik and Kaasbøll (1999)), which relates an entity of the category endurant (ED) to the process (perdurant PD) it participates in (Masolo et al. (2003))—e.g. an Enzyme that participates in a CatalyticReaction.

$$\forall x, y(participates\_in_{it}(x, y) \triangleq mpart\_of(x, y) \wedge ED(x) \wedge PD(y)) \quad (8)$$

From this characterisation, it is clear *why* meronymic part-whole relations are at least non-transitive, but generally intransitive: the category of the part is usually different from the category of the whole[8], there-

[8]The only exception being $sub\_quantity\_of$, which is under-specified.

fore one can neither make a chain with the same relation nor concatenate them with the mereological relations and assume transitivity holds.

We now consider the mereology branch of the taxonomy, where all types of relation are transitive and they can be formalised (specialised further) with proper parthood (4) as well.

We first encounter the $involved\_in$ relation, which relates two perdurants (9); for instance, Chewing is involved in Eating. This definition concurs with the parthood argument restriction "(Ad2)" in DOLCE, $P(x,y) \rightarrow (PD(x) \leftrightarrow PD(y))$, and hereby is rendered more accessible for conceptual modelling.

$$\forall x, y(involved\_in(x,y) \triangleq part\_of(x,y) \land PD(x) \land PD(y)) \quad (9)$$

We distinguish between two mereotopological relations for endurants (mereological relations that take into account space or location), one according to a 3-dimensional containment (10) and another for 2-dimensional location (11). The definitions refer to both the endurant itself and the region (R) it occupies. DOLCE has an elaborate formalisation to refer to the region of an endurant (using quality, quale, and regions), which is abbreviated here with the properties *has_3D* and *has_2D*, respectively, because this attribute-approach generally provides sufficient details in conceptual models for applications[9]. One can do away with the 2D/3D distinction and just refer to any kind of spatial region, but the distinction is included because it is a recurring relation in the informal discussions about part-whole relations for conceptual modelling (e.g. Gerstl and Pribbenow (1995); Keet (2006a); Motschnig-Pitrik and Kaasbøll (1999); Odell (1998); Winston et al. (1987)) and can be accommodated for easily.

$$\begin{aligned} &\forall x, y(contained\_in(x,y) \triangleq part\_of(x,y) \land R(x) \land R(y) \land \\ &\quad \exists z, w(has\_3D(z,x) \land has\_3D(w,y) \land ED(z) \land ED(w))) \end{aligned} \quad (10)$$

$$\begin{aligned} &\forall x, y(located\_in(x,y) \triangleq part\_of(x,y) \land R(x) \land R(y) \land \\ &\quad \exists z, w(has\_2D(z,x) \land has\_2D(w,y) \land ED(z) \land ED(w))) \end{aligned} \quad (11)$$

For instance, contained_in(John's address book, John's bag). Containment is not always conceptualised as a particular type of parthood (Bittner and Donnelly (2005); Odell (1998); Schulz et al. (2006)), but, first, $contained\_in$ always meets the above-mentioned mereological parthood and proper parthood properties. Second, those considerations on conceptualizing parts & space exhibit a hopping back and forth between considering entity only, entity & region it occupies, and region only. With examples such as contained_in(actin filament, cell) and many others in biology, both parthood and containment hold, which means that x & z and y & w coincide exactly. To address these subtle, but important, distinctions, we explicitly include *structural* parthood (see below) in the taxonomy as different from containment. The same argument holds for location. Examples for location are located_in(Amsterdam, North Holland) or located_in(Mont Blanc, Alps). Observe that the examples for location permit a finer-grained specification regarding the relata: the former relates city-province, which are entities by social convention, whereas the latter relates two physical entities (mountain and mountain range). Such ontological exuberance is not included here, because, it is generally less relevant for conceptual modelling and software system development in practice and does not result in fallacious transitivity.

Last, we constrain the *structural* parthood relation (12).

$$\forall x, y(s\_part\_of(x,y) \triangleq part\_of(x,y) \land ED(x) \land ED(y)) \quad (12)$$

We can further constrain (12) by defining two subtypes of s_part_of for POBs or to relate two $NPOB$s to ensure POBs and $NPOB$s are not interleaved (13, 14).

$$\forall x, y(s\_part\_of'(x,y) \triangleq part\_of(x,y) \land POB(x) \land POB(y)) \quad (13)$$

$$\forall x, y(s\_part\_of''(x,y) \triangleq part\_of(x,y) \land NPOB(x) \land NPOB(y)) \quad (14)$$

[9]Note that the relations do not reflect the full range of mereotopological and mereogeometrical complexities either (Borgo and Masolo (2007); Varzi (2006a)), some of which could be useful for geographic and biological information systems, but they are less relevant for the more common enterprise domain modelling. It is a point of further research how these first order logic theories can be transformed and rendered usable with formal conceptual modelling languages.

In addition, and analogous to the $member\_of'$, we can add a constraint to the part-side of s_part_of to make it a *functional* part (Vieu and Aurnague (2005); Guizzardi (2005)) and label it f_part_of; informally, individual functional dependence captures that for x to function as X then y must function as Y. Functional parthood is, however, motivated by—as opposed to 'due to'—linguistics and has not (yet) been identified explicitly as an important part-whole relation for conceptual modelling, although modellers surely have dealt with representing functional parthood. For instance, that f_part_of(Car Engine, Car) holds, with the meaning that the entity type Car denotes the set of canonical cars and each car cannot function normally without its canonical, working, engine (that instantiates the entity type Car Engine). It has been included to demonstrate ease of extension of the taxonomy if the conceptual modelling community desires further specialisation with other properties than the core mereological properties of parthood. This concludes the characterisation of the eight leaf types.

Last, observe that by adhering to Ground Mereology in the taxonomy of part-whole relations, we get six other mereological relations 'for free'. These relations (15-20) are generally useful for conceptual modelling, but in particular for conceptual modelling for Geographical Information Systems and applications for biology and biomedicine; they are as follows (after Varzi (2004)):

$$overlap(x, y) \triangleq \exists z(part\_of(z, x) \land part\_of(z, y)) \quad (15)$$

$$underlap(x, y) \triangleq \exists z(part\_of(x, z) \land part\_of(y, z)) \quad (16)$$

$$overcross(x, y) \triangleq overlap(x, y) \land \neg part\_of(x, y) \quad (17)$$

$$undercross(x, y) \triangleq underlap(x, y) \land \neg part\_of(y, x) \quad (18)$$

$$proper\_overlap(x, y) \triangleq overcross(x, y) \land overcross(y, x) \quad (19)$$

$$proper\_underlap(x, y) \triangleq undercross(x, y) \land undercross(y, x) \quad (20)$$

4. Using Part-Whole Relations: Languages and Tools

In this Section, we assess the options provided by several modelling languages for representing part-whole relations and integrating a part-whole taxonomy, and to use them for the conceptual modelling or domain ontology development stage. Conceptual data models and ontologies are not synonymous, primarily differing in *what* part of reality or knowledge for some subject domain is represented, but the requirements for dealing with representing a taxonomy of relations is the same for their languages and tools. That is, requirements for *how* to represent part-whole relations and the taxonomy in the language and the basic functionality it requires from the CASE and ontology development tools are identical for the two. Based on the semantics of part-whole relations as presented in the previous Section, we can list the following four requirements that every language intending to represent part-whole relations should pass:

1. Represent at least Ground Mereology,
2. Express ontological categories and their taxonomic relations,
3. Having the option to represent transitive and intransitive relations, and
4. Specify the domain and range restrictions (/relata/entity types) for the classes participating in a relation.

Several engineering solutions are possible to implement the basic taxonomy of part-whole relations, but the available options depend on the conceptual modelling or ontology language. In the following, we focus on UML, EER, and Object-Role Modeling (ORM) as conceptual data modelling languages, and on Description Logics (DLs) as dual-purpose conceptual modelling and ontology languages.

EER, ORM, and DLs do not have special constructors to represent the part-whole relation, although some are in favour of giving it a first-class citizen status (Artale et al. (1996); Bittner and Donnelly (2005); Keet (2006a); Lambrix and Padgham (2000); Sattler (1995); Shanks et al. (2004)). UML, on the other hand, implements two versions of the *part_of* relation: composite and shared aggregations (Object Man-

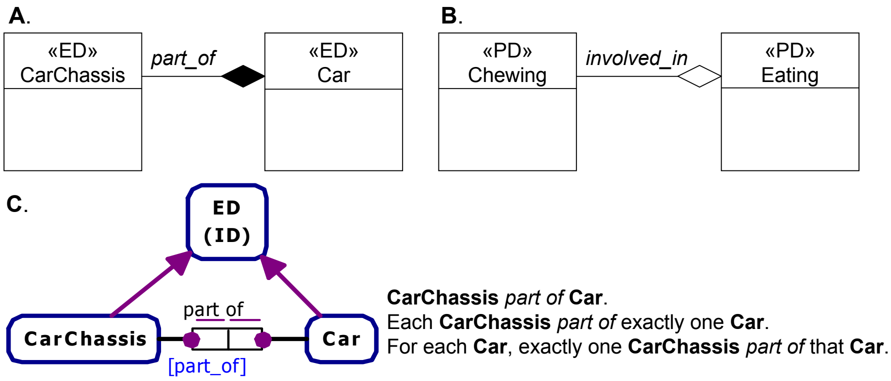

Fig. 3. Two examples of stereotyping with DOLCE categories in a UML class diagram using (A) composite and (B) shared aggregation. Without stereotyping as in ORM2 (C), the object types are subsumed by the DOLCE category ED (note that "[part_of]" is the role name and "part of" the reading label for the verbalization).

agement Group (2005)). The difference between the two UML aggregation relations is that with composite aggregations, the part is existentially dependent on exactly one whole, whereas for shared aggregation there is no constraint on the multiplicity. However, their "precise semantics [...] varies by application area and modeler" (Object Management Group (2005)), and presumably thus could be used for any of the types of part-whole relations described in Fig.1. Possibilities to include different types of part-whole relations in UML are: modifying the meta-model, adding new icons, defining stereotypes for the categories of the domain and range restrictions, and OCL constraints to describe the properties of the part-whole relation (Barbier et al. (2003); Guizzardi (2005); Motschnig-Pitrik and Kaasbøll (1999); Tan et al. (2003)); a simple example is shown in Fig.3.A–B. The relevant DOLCE categories are mapped straightforwardly onto a stereotype for each category and ensured that they form a taxonomy. Adding different icons for the 8 leaf types of part-whole relations is not advisable, because it would result in cluttered diagrams. Instead, we propose that, upon adding an association, one can select the appropriate part-whole relation from a predefined list. An additional possibility for ORM tools is to use the names of the relations in the fact editor where the user writes pseudo-natural language sentences to verbalize the fact type. This can be both to support the user in selecting a particular part-whole relation and, given an entered string for the relation, to check the types of object types surrounding the label of the part-whole relation. For subrelations, the ORM constraint of subset over binary relations is a restricted option, or it can be added through subtyping objectified (nested) relations. Given that ORM does not use stereotypes, one can add the relations to the ORM metamodel or, as depicted in Fig.3.C, have the object types subsumed by the DOLCE category-as-object type.

DLs, $\mathcal{SHIQ}$ and $\mathcal{SROIQ}$[10] (Horrocks et al. (2006)) and $\mathcal{DLR}$s (Calvanese and De Giacomo (2003)) in particular, have the ability to structure DL-roles into hierarchies (called the *Role Box*, or RBox), express both domain & range restriction and cardinality constraints on the participation of entities, represent a concept hierarchy (class taxonomy) with *Terminological Axioms* (TBox), and can support most of the mereological parthood properties (see Table 2 for details). Finally, DLs have tools, such as iCOM, for conceptual modelling development (Franconi and Ng (2000)) and "CASE"-like tools for ontology development, such as Protégé, both of which are linked to a DL reasoner such as Racer or Pellet, which enables the modeller to easily specify both a DL-concept hierarchy and a DL-role hierarchy ("object properties" in Protégé) and run the reasoner over it with one mouse-click. The next Section discusses how to integrate the two hierachies and how to reason over them.

[10] $\mathcal{SROIQ}$ extended with data types is a basis for OWL 1.1, with syntax at: http://www-db.research.bell-labs.com/user/pfps/owl/overview.html (Editor's draft 14-6-2006).

Table 2

Properties of parthood and proper parthood compared to their support in $\mathcal{DLR}_\mu$ (with fixpoint extension (Calvanese et al. (1999))), $\mathcal{SHOIN}$ and $\mathcal{SROIQ}$. *: properties of the parthood relation (in Ground Mereology); ‡: properties of the proper parthood relation (in Ground Mereology).

Feature ⇒ **Language** ⇓	Reflexivity *	Antisymmetry *	Transitivity $^{*\ \ddagger}$	Asymmetry ‡	Irreflexivity ‡
$\mathcal{DLR}_\mu$	+	-	+	+	+
$\mathcal{SHOIN}$	-	-	+	+	-
$\mathcal{SROIQ}$	+	-	+	+	+

5. Reasoning over a Part-Whole Taxonomy

Although the features in the current conceptual modelling languages and CASE tools are somewhat limited in fully supporting the representation of the part-whole taxonomy together with the relations' properties, it is possible to at least specify the domain and range for the relations. The natural next step is to ensure that this typing is done and used correctly, therefore we now focus on automated reasoning and we show that managing a taxonomy of relations requires a new reasoning service to make *ontologically* correct inferences over both conceptual models and domain ontologies. The aim of such a reasoning service is to help a modeller to be more precise in representing part-whole relations and to achieve an effective *use* of the part-whole taxonomy. A decision procedure to guide the modeller was proposed by Keet (2006a), for which several implementation options were suggested: a cheat-sheet, drop-down box in a CASE tool to type the relation, or software-support for the decision procedure with questions and examples corresponding to each decision point. However, none of the suggested options can *compute* correctness, or at least logical consistency, of the modelling decisions taken by the modeller about the relations. Given that representing part-whole relations is not an easy task, it is highly desirable to offer automated reasoning support to aid the modeller in checking consistency and satisfiability of the conceptual model and to derive new information entailed in the model.

We choose Description Logics to formulate the desired reasoning service. DLs have been shown useful for reasoning both over conceptual models like EER, ORM, and UML (Artale et al. (2003); Baader et al. (2002); Berardi et al. (2005); Calvanese et al. (1998, 1999); Keet (2007)) and ontology languages such as OWL-DL, OWL-Lite[11], OWL 1.1, and DL-Lite (Calvanese et al. (2005)). At present, most properties of the mereological theories can be represented in expressive DLs (Calvanese and De Giacomo (2003); Horrocks et al. (2006)), except for antisymmetry; see Table 2 for details.

For the current purpose, we are mainly interested in reasoning over roles given a role hierarchy (represented by the *Role Box*, RBox) and a DL-concept hierarchy (represented by the *Terminological Box*, TBox). For our purposes, it is enough to introduce briefly the simple $\mathcal{ALCI}$ DL language, which has already sufficient expressivity to support the reasoning service we are interested in. $\mathcal{ALCI}$ is a sub-language of both the proposed OWL 1.1, which is $\mathcal{SROIQ}$ with datatypes, and the $\mathcal{DLR}$ family of DL languages (Calvanese and De Giacomo (2003)), which were specifically developed to provide a formal underpinning and unifying paradigm for conceptual modelling languages and permit automated reasoning over conceptual models. With respect to the formal apparatus, we will strictly follow the concept language formalism presented in Baader et al. (2002). Basic types of $\mathcal{ALCI}$ are *concepts* and *roles*. A concept—*sensu* DL—is a description gathering the common properties among a collection of individuals; from a logical point of view, it is a unary predicate ranging over the domain of individuals. Inter-relationships between these individuals are represented by means of roles, which are interpreted as binary relations over the domain of individuals. According to the syntax rules of Fig. 4, $\mathcal{ALCI}$ *concepts* (denoted by the letters C and D) are built out of *atomic concepts* (denoted by the letter A) and *atomic roles* (denoted by the letter P). In the following we use $\exists R$ as a shortcut for $\exists R.\top$. As usual, an $\mathcal{ALCI}$ interpretation is a pair, $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$, where $\Delta^\mathcal{I}$ is a non-empty set of objects (the *domain* of $\mathcal{I}$) and $\cdot^\mathcal{I}$ an *interpretation function* such that, for

[11] http://www.w3.org/TR/2004/REC-owl-semantics-20040210/syntax.html.

$C, D \rightarrow$	$A \mid$	(atomic concept)	$A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
	$\top \mid$	(top)	$\top^{\mathcal{I}} = \Delta^{\mathcal{I}}$
	$\bot \mid$	(bottom)	$\bot^{\mathcal{I}} = \emptyset$
	$\neg C \mid$	(complement)	$(\neg C)^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
	$C \sqcap D \mid$	(conjunction)	$(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$
	$C \sqcup D \mid$	(disjunction)	$(C \sqcup D)^{\mathcal{I}} = C^{\mathcal{I}} \cup D^{\mathcal{I}}$
	$\forall R.C \mid$	(univ. quantifier)	$(\forall R.C)^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} \mid \forall b.R^{\mathcal{I}}(a,b) \Rightarrow C^{\mathcal{I}}(b)\}$
	$\exists R.C \mid$	(exist. quantifier)	$(\exists R.C)^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} \mid \exists b.R^{\mathcal{I}}(a,b) \wedge C^{\mathcal{I}}(b)\}$
$R, S \rightarrow$	$P \mid$	(atomic role)	$P^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$
	R^{-}	(inverse role)	$R^{-\mathcal{I}} = \{(a,b) \in \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \mid (b,a) \in R^{\mathcal{I}}\}$

Fig. 4. Syntax and Semantics for the $\mathcal{ALCI}$ Description Logic

every concept C, and every role R, we have $C^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ and $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. Generic concepts and roles are interpreted by $\mathcal{I}$ according to the semantic equations of Fig.4.

A *knowledge base* in this context is a pair $\Sigma = (\mathcal{T}, \mathcal{R})$ where $\mathcal{T}$ is a set of *terminological axioms* (TBox) of the form $C \sqsubseteq D$ (general concept inclusion axiom), and $\mathcal{R}$ is a set of *role axioms* (RBox) of the form $R \sqsubseteq S$ (subrole axiom) and $R \sqsubseteq C_1 \times C_2$ (Domain & Range axiom)[12]. An interpretation $\mathcal{I}$ satisfies $C \sqsubseteq D$ iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$, $R \sqsubseteq S$ iff $R^{\mathcal{I}} \subseteq S^{\mathcal{I}}$, and $R \sqsubseteq C_1 \times C_2$ iff $R^{\mathcal{I}} \subseteq C_1^{\mathcal{I}} \times C_2^{\mathcal{I}}$. A knowledge base Σ is *satisfiable* if there is an interpretation $\mathcal{I}$ which satisfies every axiom in Σ; in this case $\mathcal{I}$ is called a *model* of Σ. Σ *logically implies* an axiom α (written $\Sigma \models \alpha$) if α is satisfied by every model of Σ. A concept C (role R) is satisfiable, given a knowledge base Σ, if there exists a model $\mathcal{I}$ of Σ such that $C^{\mathcal{I}} \neq \emptyset$ ($R^{\mathcal{I}} \neq \emptyset \times \emptyset$), i.e. $\Sigma \not\models C \sqsubseteq \bot$ ($\Sigma \not\models \exists R \sqsubseteq \bot$), thus, role satisfiability can be reduced to concept satisfiability). We illustrate the DL syntax in the following example.

Example. The ORM example in Fig.3 can be represented in $\mathcal{ALCI}$ with the following knowledge base, where we use [part_of] as an atomic DL role, omit the uniqueness and object type reference modes as they are not relevant for the current purpose, and ED is endurant, PD perdurant, and PT particular (see Fig.2):

```
CarChassis ⊑ ED
Car ⊑ ED
CarChassis ⊑ ∃PARTOF.Car
Car ⊑ ∃PARTOF⁻.CarChassis
```

Further, we can describe, e.g., the domain and range restriction of `PARTOF`, one of its subtypes `INVOLVEDIN`, and its domain and range restriction:

```
PARTOF ⊑ PT × PT
INVOLVEDIN ⊑ PARTOF
INVOLVEDIN ⊑ PD × PD
```

The DOLCE categories `ED` and `PD` are both subconcepts of `PT` and are disjoint:

```
ED ⊑ PT
PD ⊑ PT
ED ⊑ ¬PD
```

Given this knowledge base, it is clear that it is logically correct that the car chassis is `PARTOF` of the car and that it can never be `INVOLVEDIN` the car in the current state of the knowledge base. Indeed, both car and car chassis are subconcepts of `ED`, which in turn is a subconcept of `PT`, hence, `PARTOF` can be used. Furthermore, `INVOLVEDIN` is typed with `PD`, i.e., it can be used to relate perdurants only, but we have that `ED` and `PD` are disjoint, and car chassis and car are both types of `ED`, hence, declaring `INVOLVEDIN` for the car and car chassis would lead to a logical inconsistency. $\diamondsuit$

[12]Note that Domain & Range axioms are a shortcut for the following two axioms: $\exists R \sqsubseteq C_1$, $\exists R^{-} \sqsubseteq C_2$, with C_1, C_2 generic concepts.

Reasoning over a role taxonomy means checking for role satisfiability and checking the compatibility of domain & range axioms with respect to subrole relations holding in the RBox. Role consistency is an issue when dealing with both RBoxes and TBoxes, since an unsatisfiable concept can be associated to either the domain or the range of a role leading to an inconsistent role. Extant reasoners, such as Pellet, FaCT, and Racer, can check role satisfiability by reducing it to concept satisfiability—i.e. by checking whether $\Sigma \not\models \exists R \sqsubseteq \bot$. In this paper we introduce a new reasoning service called *RBox compatibility* where the domain and range restrictions for roles are checked against the RBox and the TBox. We show that this new service can help in avoiding unwanted logical consequences of an RBox over a set of concept hierarchies expressed in the TBox. We demonstrate with an example both the relevance of reasoning over an RBox and the problem with extant reasoning services that do not deal adequately—in the sense of deriving *ontologically* correct results—with a role hierarchy. We thus provide a formal definition for the RBox compatibility test and then show how this new reasoning service can help in avoiding ontologically undesired consequences. For the example, the ontology development tool Protégé v3.2 beta is used (together with the Racer reasoner) because the screenshots are more illustrative than those of the formal conceptual modelling tools and the issue is the same for both domain ontologies and formal conceptual models.

Example (cont'd). Take the three top-most categories of DOLCE (as in Fig.2) and add them as classes in Protégé. Then add a role hierarchy with pwrelation at the top that subsumes part-of that in turn subsumes involved-in with their proper domain and range restrictions, as defined in Section 3. Then, relate Chewing to Eating with the involved-in relation, and Chassis to Car through the part-of relation. This is depicted in Fig.5.`A1-B` as screenshots from Protégé v3.2 beta. For illustrative purpose, we have created another class hierarchy too (Fig.5.`A2`) where Chassis and Car are not subconcepts of endurant ED anymore, but of particular PT instead. Further, we have another scenario in Fig.5.`C` with an "incompatible" role hierarchy, i.e. a role hierarchy where the domain and range restrictions are *inverted* compared to the correct scenario, since now part-of's subrole involved-in can be used to relate anything to anything whereas part-of itself can only relate perdurants.

Using the reasoning options of Protégé with Racer, we obtain the results as summarised in Fig.6. Choosing the TBox (`A1`) with the "correct" RBox (`B`) shows, as expected, that the ontology is fine. Testing the TBox (`A1`) with the "incompatible" RBox (`C`), it says Chassis is inconsistent, whereas using the TBox (`A2`) with the "incompatible" RBox (`C`) reclassifies Chassis as a type of perdurant. Although with relation to the scenarios using the RBox (`C`), these deductions are *logically* correct, we claim here that the reasoner ought to have found a compatibility issue in the RBox (`C`), because the domain and range restrictions of a subrole cannot be more general (higher up in the TBox) than those of its parent role. $\diamondsuit$

We are now able to define formally the new proposed reasoning service that we call *RBox compatibility*. We start first with some notation. In the following Definition we assume that for each role there is provided exactly one Domain & Range axiom. Without loss of generality, we can assume that the axiom $R \sqsubseteq \top \times \top$ holds when an explicit Domain & Range axiom is lacking.

Definition 1 (User-defined Domain and Range Concepts). *Let R be a role and $R \sqsubseteq C_1 \times C_2$ its associated Domain & Range axiom. Then, with the symbol D_R we indicate the* User-defined Domain *of R—i.e., $D_R = C_1$—while with the symbol R_R we indicate the* User-defined Range *of R—i.e., $R_R = C_2$.*

We now define the new reasoning service, *RBox compatibility*, that checks the compatibility of Domain & Range axioms with respect to both the role hierarchy holding in the RBox and the concept hierarchy holding in the TBox. The tests of the RBox compatibility service are not only necessary but also sufficient for finding domain-range problems, because it covers each permutation of domain and range of the parent and child relation in the role hierarchy.

Definition 2 (RBox Compatibility). *For each pair of roles, R, S, such that $\langle \mathcal{T}, \mathcal{R} \rangle \models R \sqsubseteq S$, check whether:*

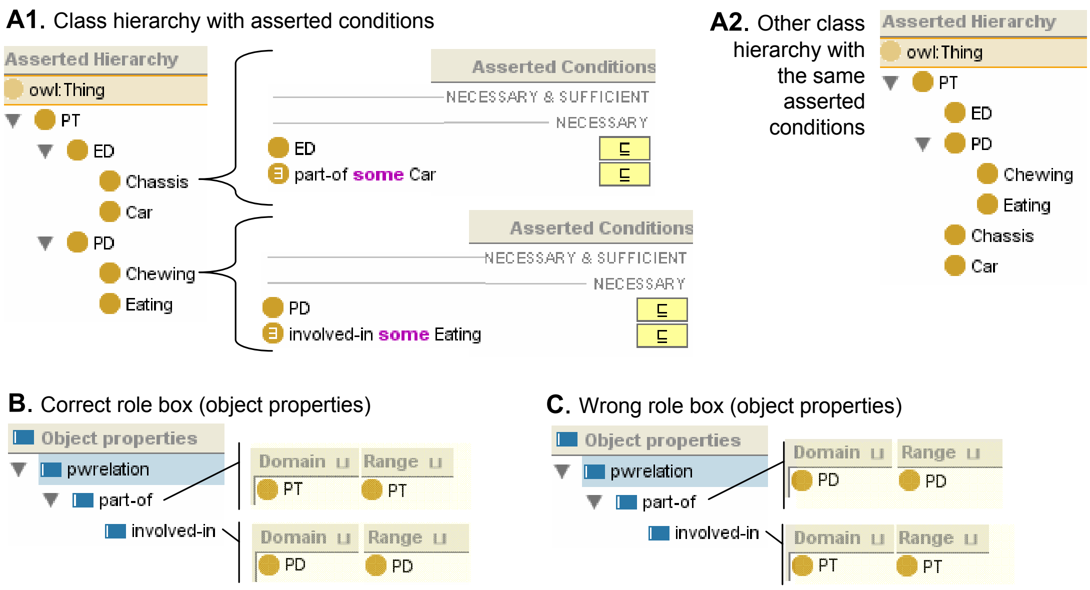

Fig. 5. Two class hierarchies and two role (object property) hierarchies in Protégé.

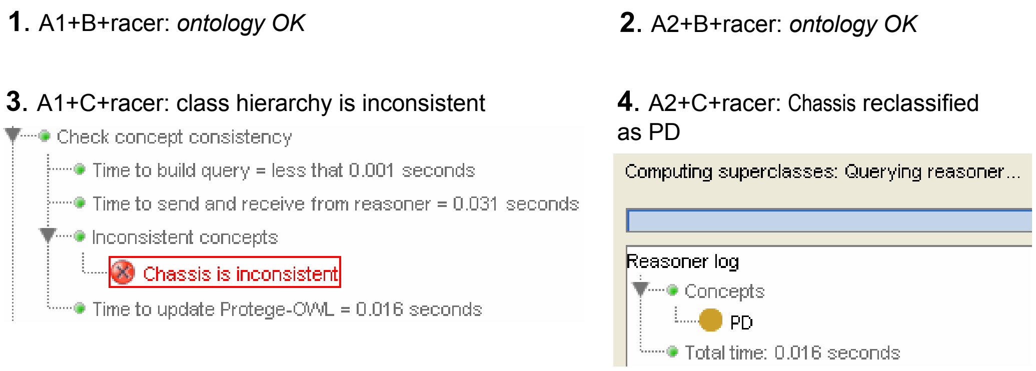

Fig. 6. The examined four combinations of automated reasoning over the in Fig.5 shown class hierarchies and object property (Role Box) hierarchies with the current results of the reasoner (Racer) when checking consistency and computing the inferred hierarchy (classifying the taxonomy).

Test 1. $\langle \mathcal{T}, \mathcal{R} \rangle \models D_R \sqsubseteq D_S$ *and* $\langle \mathcal{T}, \mathcal{R} \rangle \models R_R \sqsubseteq R_S$*;*
Test 2. $\langle \mathcal{T}, \mathcal{R} \rangle \not\models D_S \sqsubseteq D_R$*;*
Test 3. $\langle \mathcal{T}, \mathcal{R} \rangle \not\models R_S \sqsubseteq R_R$.

An RBox is said to be compatible iff `Test 1` *and* (2 *or* 3) *hold for all pairs of role-subrole in the RBox.*

A formal conceptual model or domain ontology that does not respect the RBox compatibility criterion can be considered as *ontologically flawed*. Checking for RBox compatibility and, thus, for ontological RBox correctness, can be done by using the classical (for DL reasoners) subsumption reasoning service. In particular, we propose the following actions whenever the above defined tests fail. If `Test 1` does not hold, we raise a warning that domain & range restrictions of either R or S are in conflict with the role hierarchy proposing either

(i) To change the role hierarchy or
(ii) To change domain & range restrictions or
(iii) If the test on the domains fails, then propose a new axiom $R \sqsubseteq D'_R \times R_R$, where $D'_R \equiv D_R \sqcap D_S$[13],

[13]The axiom $C_1 \equiv C_2$ is a shortcut for the axioms: $C_1 \sqsubseteq C_2$ and $C_2 \sqsubseteq C_1$.

which subsequently has to go through the RBox compatibility service (and similarly when `Test 1` fails on range restrictions).

If `Test 2` and `Test 3` fail, we raise a warning that R cannot be a proper subrole of S but that the two roles can be equivalent. We then propose two actions: either

(a) Accept the possible equivalence between the two roles or

(b) Change domain & range restrictions.

Please note that the above actions should allow the user to leave unchanged both the TBox and the RBox, since both tests do not imply any real logical inconsistency—i.e., the proposed actions are not mandatory. Thus, our proposal refines the notion of correctness by distinguishing between the notion of *logical correctness* and that of *ontological correctness*. We demonstrate the application of the RBox compatibility service with the running example.

Example (cont'd). Observe that both Protégé deductions—i.e., Chassis inconsistent in scenario `A1+C`, and Chassis as a perdurant in scenario `A2+C`—are just logical consequences of disregarding the ontological incorrect (but logically consistent) assumptions contained in the RBox (`C`). Put differently: with the RBox compatibility service in place, one obtains these deductions by ignoring all warnings raised and suggestions proposed by the service.

Now consider RBox (`C`) with the RBox compatibility service, then `Test 1` fails because both the domain and range of involved-in, particulars PT, are parent concepts of its superrole (part-of) domain and range restrictions that are set to perdurants PD. Between the possible options (i)-(iii) that an user can choose, the second one is the most appropriate. In particular, the user can assign the correct (w.r.t. the part-whole taxonomy) domain & range restrictions to the roles thus obtaining ontologically correct deductions[14].

For illustration, let us assume the user chooses option (i) instead of changing the domain and range restrictions. The result in the RBox is that the roles are inverted such that now part-of $\sqsubseteq$ involved-in, which contradicts the part-whole taxonomy, but is logically correct. `Test 2` and `3` then pass. Subsequent checking of the TBox (`A1`) will still yield an inconsistent Chassis, because the RBox is ontologically flawed. More precisely, Chassis is part-of Car and both are still subconcepts of ED, whereas part-of is typed with PD that is disjoint from ED; hence, Chassis as part of Car can never be instantiated.

Going along with option (iii) can, in this example, still yield an inconsistent theory. Choosing (iii), the following steps occur. A new axiom for involved-in is proposed for the domain restriction: with $D'_R \equiv D_R \sqcap D_S$ we have in the example $D'_R \equiv \mathsf{PT} \sqcap \mathsf{PD}$, which, with $\mathsf{PD} \sqsubseteq \mathsf{PT}$, results in $D'_R \equiv \mathsf{PD}$, and upon accepting this proposal we get involved-in $\sqsubseteq \mathsf{PD} \times \mathsf{PT}$. The same sequence is repeated for the range restrictions, such that we have involved-in $\sqsubseteq \mathsf{PD} \times \mathsf{PD}$. Then `Test 2` and `3` can be executed. Recollecting, we have: part-of $\sqsubseteq \mathsf{PD} \times \mathsf{PD}$ and involved-in $\sqsubseteq$ part-of. Thus, the compatibility reasoning service will suggest options (a) and (b) to the user. By choosing option (b) the user has the possibility to change the domain and range restrictions to those in RBox (`B`). If, on the other hand, the users accepts the option (a) both concepts Chassis and Car will be unsatisfiable.

Looking briefly at the combination (`A2+C`), then upon choosing option (i), both Chassis and Car are re-classified from PT to PD to comply with the domain & range restriction of part-of. Choosing option (iii) together with option (a) will still result into a re-classification of both Chassis and Car as subconcepts of PD. Although the conceptual model (or domain ontology) with the re-classification is satisfiable, it is ontologically flawed both regarding the role hierarchy and the concept hierarchy, and again, selecting option (ii) to correct the domain & range restrictions to that of RBox (`B`) is the appropriate choice. $\diamondsuit$

Thus, with the currently available reasoners, one may get error messages about inconsistent concepts or undesired equivalences and/or subsumptions between concepts, whereas the error is in the role hierarchy. The RBox compatibility service can be used equally after updating an ontology, and additional user-

[14] Although the precise definition of 'ontologically correct' is a topic of active research efforts, it is thus far generally used to contrast it with obvious modeling errors (is_a vs part_of, mixing criteria for subsumption) and corrections made thanks to, e.g., the OntoClean methodology (Guarino and Welty, 2004) or disambiguation of relations through usage of the Relation Ontology (Smith et al., 2005).

friendly messages might be devised for the user interface of the CASE (or ontology development) tools to explain the five choices of the three tests.

Relations/properties/roles are essential components in both conceptual models and ontologies, which receives a more prominent place when role hierarchies and domain & range restrictions are properly declared, as with the part-whole relations taxonomy. To reason over the conceptual models and domain ontologies requires inclusion of reasoning over both concepts and roles to check if they are consistent. We showed how ontologically unwanted logical implications concerning the concept hierarchy can be avoided with the additional compatibility check on the role hierarchy. These aspects, in turn, ease conceptual modelling and domain ontology development and results in a representation that is closer to the real world semantics it intends to represent.

6. Conclusions and Further Research

We have introduced a formal taxonomy of part-whole relations that are commonly used in conceptual modelling. The main rationale for distinguishing types of part-whole relations are (in)transitivity of the relation and the categories of the entity types they relate. This enables conceptual modellers (and developers of domain ontologies) to create representations that are closer to the real-world semantics, hence, improve quality of the software. In addition to addressing the problem of *identifying* different types of part-whole relations, automated reasoning for *using* the taxonomy was investigated. This resulted in a new requirement and corresponding reasoning service, called *RBox compatibility*, for automated reasoners. The RBox compatibility service checks the logical and ontological consistency of role domain & range restrictions with respect to both the role hierarchy and the class hierarchy. The notion of RBox compatibility gives also rise to formally introduce a notion of *ontological correctness* that can be distinct from the classical notion of logical consistency.

There are, however, multiple other facets when representing part-whole relations in conceptual models, which we did not address here. These aspects include, among others, properties such as degree of shareability (Motschnig-Pitrik and Kaasbøll (1999)), separability (Guizzardi (2005)), full support of more comprehensive mereological theories, distributivity of properties (Artale et al. (1996)), and reasoning scenarios with part-whole relations (e.g. Lambrix and Padgham, 2000). Methodologically, these facets need attention in the modelling process *after* identifying the most appropriate type of part-whole relation and after the basic consistency checks over the relations and the classes have been run. We are currently investigating trade-offs between comprehensiveness of representing more aspects of part-whole relations versus expressivity of several description logic languages.

References

Artale, A., Franconi, E., Guarino, N., and Pazzi, L. (1996). Part-Whole Relations in Object-Centered Systems: an Overview. *Data & Knowledge Engineering*, **20**(3), 347-383.

Artale, A., Franconi, E., Mandreoli, F. (2003). Description logics for modelling dynamic information. In Chomicki, J., van der Meyden, R., Saake, G. (Eds), *Logics for Emerging Applications of Databases*. Lecture Notes in Computer Science, Springer-Verlag, Berlin.

Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P. F. editors. (2002). *Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press, Cambridge.

Barbier, F., Henderson-Sellers, B., Le Parc-Lacayrelle, A., Bruel, J.-M. (2003). Formalization of the whole-part relationship in the Unified Modelling Language. *IEEE Transactions on Software Engineering*, **29**(5), 459-470.

Berardi, D., Calvanese, D., De Giacomo, G. (2005). Reasoning on UML class diagrams. *Artificial Intelligence*, **168**(1-2), 70-118.

Bittner, T., Donnelly, M. (2005). Computational ontologies of parthood, componenthood, and containment, In: *Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence 2005 (IJCAI05)*. Kaelbling, L. (ed.). pp382-387.

Borgo, S., Masolo, C. (2007). Full mereogeometries. *Journal of Philosophical Logic*. (*to appear*).

Calvanese, D., Lenzerini, M., Nardi, D. (1998). Description logics for conceptual data modeling. In J. Chomicki and G. Saake, editors, *Logics for Databases and Information Systems*. Kluwer, Amsterdam.

Calvanese, D., De Giacomo, G., Lenzerini, M. (1999). Reasoning in expressive description logics with fixpoints based on automata on infinite trees. In: *Proc. of the 16th Int. Joint Conf. on Artificial Intelligence (IJCAI'99)*, pp84-89.

Calvanese, D., De Giacomo, G. (2003). Expressive description logics. In: *The Description Logic Handbook: Theory, Implementation and Applications*, Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P. (Eds). Cambridge University Press, Cambridge. pp178-218.

Calvanese, D., De Giacomo, G., Lembo, D., Lenzerini, M., Rosati, R. (2005). DL-Lite: Tractable description logics for ontologies. In: *Proc. of the 20th Nat. Conf. on Artificial Intelligence (AAAI 2005)*, pp602-607.

Franconi, E., Ng, G. (2000). The iCom tool for intelligent conceptual modelling. *7th Intl Workshop on Knowledge Representation meets Databases*. Berlin, Germany.

Gerstl, P., Pribbenow, S. (1995). Midwinters, end games, and body parts: a classification of part-whole relations. *International Journal of Human-Computer Studies*, **43**, 865-889.

Guarino, N., Welty, C.A. (2004). An overview of OntoClean. In: *Handbook on ontologies*. Stab, S., Studer, R. (eds.). Springer Verlag, 2004. pp151-159.

Guizzardi, G. (2005). *Ontological foundations for structural conceptual models*. PhD Thesis, Telematica Institute, Twente University, Enschede, the Netherlands. ISBN 90-75176-81-3.

Herre, H., Heller, B. (2006). Semantic foundations of medical information systems based on top-level ontologies. *Knowledge-Based Systems*, **19**, 107-115.

Horrocks, I., Kutz, O., Sattler, U. (2006). The Even More Irresistible $\mathcal{SROIQ}$. In: *Proceedings of the 10th International Conference of Knowledge Representation and Reasoning (KR-2006)*, Lake District, UK.

Johansson, I. (2004). On the transitivity of the parthood relation. In: *Relations and predicates*, Hochberg, H., Mulligan, K. (eds.). Ontos Verlag, Frankfurt. pp161-181.

Keet, C.M. (2006a). Part-whole relations in Object-Role Models. *2nd International Workshop on Object-Role Modelling (ORM 2006)*, Montpellier, France, Nov 2-3, 2006. In: OTM Workshops 2006. Meersman, R., Tari, Z., Herrero., P. et al. (Eds.), Lecture Notes in Computer Science 4278. Berlin: Springer-Verlag, Berlin. pp1116-1127.

Keet, C.M. (2006b). *Introduction to part-whole relations: mereology, conceptual modelling and mathematical aspects*. KRDB Research Centre Technical Report KRDB06-3 (Tutorial reader), Faculty of Computer Science, Free University of Bozen-Bolzano, Italy. 2 October 2006.

Keet, C.M. (2007). *Mapping the Object-Role Modeling language ORM2 into Description Logic language* $\mathcal{DLR}_{ifd}$. KRDB Research Centre Technical Report KRDB07-2, Faculty of Computer Science, Free University of Bozen-Bolzano, Italy. 15 February 2007. arXiv:cs.LO/0702089v1.

Lambrix, P., Padgham, L. (2000). Conceptual modeling in a document management environment using part-of reasoning in description logics. *Data & Knowledge Engineering*, **32**(1), 51-86.

Masolo, C., Borgo, S., Gangemi, A., Guarino, N. and Oltramari, A. (2003). *Ontology Library*. WonderWeb Deliverable D18 (ver. 1.0, 31-12-2003). http://wonderweb.semanticweb.org.

Masolo, C., Vieu, L., Bottazzi, E., Catenacci, C., Ferrario, R., Gangemi, A., Guarino, N. (2004). Social Roles and their Descriptions. *9th International Conference on Principles of Knowledge Representation and Reasoning (KR2004)*. Banff, Canada.

Motschnig-Pitrik, R., Kaasbøll, J. (1999). Part-Whole Relationship Categories and Their Application in Object-Oriented Analysis. *IEEE Transactions on Knowledge and Data Engineering*, **11**(5), 779-797.

Object Management Group. (2005). *Unified Modeling Language: Superstructure*. v2.0. formal/05-07-04. http://www.omg.org/cgi-bin/doc?formal/05-07-04.

Odell, J.J. (1998). Six different kinds of composition. *Advanced Object-Oriented Analysis & Design using UML*. Cambridge University Press, Cambridge.

Opdahl, A., Henderson-Sellers, B., Barbier F. (2001). Ontological Analysis of whole-part relationships in OO-models. *Information and Software Technology*, **43**, 387-399.

Rector, A., Rogers, J., Bittner, T. (2006). Granularity, scale and collectivity: When size does and does not matter. *Journal of Biomedical Informatics*, **39**(3), 333-349.

Sattler, U. (1995). A concept language for an engineering application with part-whole relations. In: *Proceedings of the international workshop on description logics*, Borgida, A., Lenzerini, M., Nardi, D., Nebel, B. (Eds.), pp119-123.

Schneider, L. (2003). Designing Foundational Ontologies. The Object-Centered High-level Reference Ontology OCHRE as a Case Study. In: Song, I.-Y.; Liddle, S.W.; Ling, T.W.; Scheuermann, P. (eds.), *Conceptual Modeling - ER 2003, 22nd International Conference on Conceptual Modeling*. LNCS 2813. Heidelberg: Springer, 91-104.

Schulz, S., Hahn, U., Romacker, M. (2000). Modeling Anatomical Spatial Relations with Description Logics. *Proceedings of the AMIA Symposium 2000*. Overhage, J.M. (ed.), pp779-83.

Schulz, S., Kumar, A., Bittner, T. (2006). Biomedical ontologies: What part-of is and isn't. *Journal of Biomedical Informatics*, **39**(3), 350-361.

Shanks, G., Tansley, E., Weber, R. (2004). Representing composites in conceptual modeling. *Communications of the ACM*, **47**(7), 77-80.

Simons, P. (1987). *Parts: a study in Ontology*. Clarendon Press, Oxford.

Smith, B., Ceusters, W., Klagges, B., Köhler, J., Kumar, A., Lomax, J., Mungall, C.J., Neuhaus, F., Rector, A., Rosse, C. (2005). Relations in Biomedical Ontologies. *Genome Biology*, **6**, R46.

Tan, H.B.K, Hao, L., Yang, Y. (2003). On formalizations of the whole-part relationship in the Unified Modeling Language. *IEEE Transactions on Software Engineering*, **29**(11), 1054-1055.

Varzi, A.C. (2004). Mereology. *The Stanford Encyclopedia of Philosophy*. (Fall 2004 Edition), Zalta, E.N. (ed.). http://plato.stanford.edu/archives/fall2004/entries/mereology/.

Varzi, A.C. (2006a). Spatial reasoning and ontology: parts, wholes, and locations. In: *The Logic of Space*, Aiello, M., Pratt-Hartmann I., van Benthem J. (eds.). Dordrecht: Kluwer Academic Publishers. *(to appear)* http://www.columbia.edu/~av72/papers/Space_2006.pdf, date accessed: 18-9-2006.

Varzi, A.C. (2006b). A Note on the Transitivity of Parthood. *Applied Ontology* (to appear).

Vieu, L., Aurnague, M. (2005). Part-of Relations, Functionality and Dependence. In: M. Aurnague, M. Hickmann, L. Vieu (eds.), *Categorization of Spatial Entities in Language and Cognition*. John Benjamins, Amsterdam.

Winston, M.E., Chaffin, R., Herrmann, D. (1987). A taxonomy of part-whole relations. *Cognitive Science*, **11**(4), 417-444.