A Common Foundational Theory for Bridging two levels in Ontology-Driven Conceptual Modeling

Giancarlo Guizzardi, Veruska Zamborlini

Ontology and Conceptual Modeling Research Group (NEMO)
Computer Science Department,
Federal University of Espírito Santo (UFES), Brazil
gguizzardi@inf.ufes.br, vczamborlini@inf.ufes.br

Abstract. In recent years, there has been a growing interest in the use of Foundational Ontologies, i.e., ontological theories in the philosophical sense to provide real-world semantics and principled modeling guidelines for conceptual domain modeling languages. In this paper, we demonstrate how a philosophically sound and cognitively-oriented ontological theory of *objects* and *moments* (property-instances) has been used to: (i) (re)design a system of modeling primitives underlying the conceptual domain modeling language OntoUML; (ii) derive supporting technology for mapping these conceptual domain models to less-expressive computationally-oriented codification languages. In particular, we address here a mapping strategy to OWL (Web Ontology Language) which addresses the issue of temporally changing information.

Keywords: Ontological Foundations, Conceptual Domain Modeling, Temporally Changing Information, UFO, OntoUML, OWL

1 Introduction

In December 2011, the Object Management Group (OMG) released a new Request for Proposal (RFP) entitled SIMF (Semantic Information Modeling Federation) [1]. The SIMF initiative is aimed at developing a *"standard that addresses the federation of information across different representations, levels of abstraction, communities, organizations, viewpoints, and authorities. Federation, in this context, means using independently conceived information sets together for purposes beyond those for which the individual information sets were originally defined"*. Moreover, the proposal should *"define, adopt and/or adapt languages to express the conceptual domain models, logical information models and model bridging relationships needed to achieve this federation"*.

Information Federation is inherently a semantic interoperability problem and underlying this RFP there is the recognition that current modeling technologies fall short in suitably supporting this task of semantic interoperability. At first, at the conceptual domain modeling level, we need a language which is truthful to the subtleties of the subject domains being represented. Moreover, this language should be expressive enough to make explicit the ontological commitment of the different worldviews underlying different models to be federated.

In a seminal paper [2], John Mylopoulos defines conceptual modeling as *"the activity of representing aspects of the physical and social world for the purpose of communication, learning and problem solving among human users"* and states that *"the adequacy of a conceptual modeling notation rests in its ability to promote understanding and problem solving regarding these domains among these human users…not machines"*. In summary, conceptual modeling is about representing in diagrammatic notations, conceptualizations of reality to be shared among human users. For this reason, as defended by a number of authors over the years [3,4], a conceptual modeling notation should have its primitives grounded in the categories of a Foundational Ontology. Moreover, following the aforementioned desiderata, this Foundational Ontology should be one that takes both Cognition and Linguistic Competence seriously into consideration [5,6].

The expressivity in a modeling language needed to make explicit the ontological commitments of a complex domain tends to make this language prohibitive from a computational point of view in tasks such as automated reasoning. Conversely, computationally tractable logical languages tend to lack the expressivity to handle this essential aspect of semantic interoperability. For this reason, as defended in [5,6], we need a two level approach for domain modeling: (i) firstly, we should develop conceptual models as rich as possible to efficiently support the tasks of meaning negotiation and semantic interoperability across "communities, organizations, viewpoints, and authorities"; (ii) once the proper relationship between different information models is establish, we can generate (perhaps several different) implementations in different logical languages addressing different sets of non-functional design requirements.

In this paper, we illustrate a number of the aforementioned aspects. Firstly, we present a fragment of a Cognitive Foundational Ontology which has been employed over the years to analyze, re-design and integrate a number of conceptual modeling languages and reference models (section 2). Secondly, we illustrate how this Foundational Ontology has been used to redesign an Ontologically and Cognitively well-founded Conceptual Domain Modeling Language (CDML) (section 3). Finally, we show that these theory's ontological categories (which define the ontological semantics of this CDML) can be directly employed for creating transformations between models in this language and computationally-oriented representations (section 4). In particular, we address here the issue of devising transformation strategies for representing the important modal (temporal) aspects of this CDML in OWL (Web Ontology Language), given the limitations of the latter language in representing this sort of information. And at last, section 5 of the article presents some final considerations.

2 Ontological Background

In this section, we discuss the Unified Foundational Ontology (UFO). UFO is a reference ontology of endurants based on a number of theories from Formal Ontology, Philosophical Logics, Philosophy of Language, Linguistics and Cognitive Psychology. In the sequel, we restrict ourselves to a fragment of this ontology, depicted in Figure 1. Moreover, due to space limitations and the focus of the paper, we present the ontological categories comprising UFO superficially. For an in depth presentation and corresponding formalization, one should refer to [7].

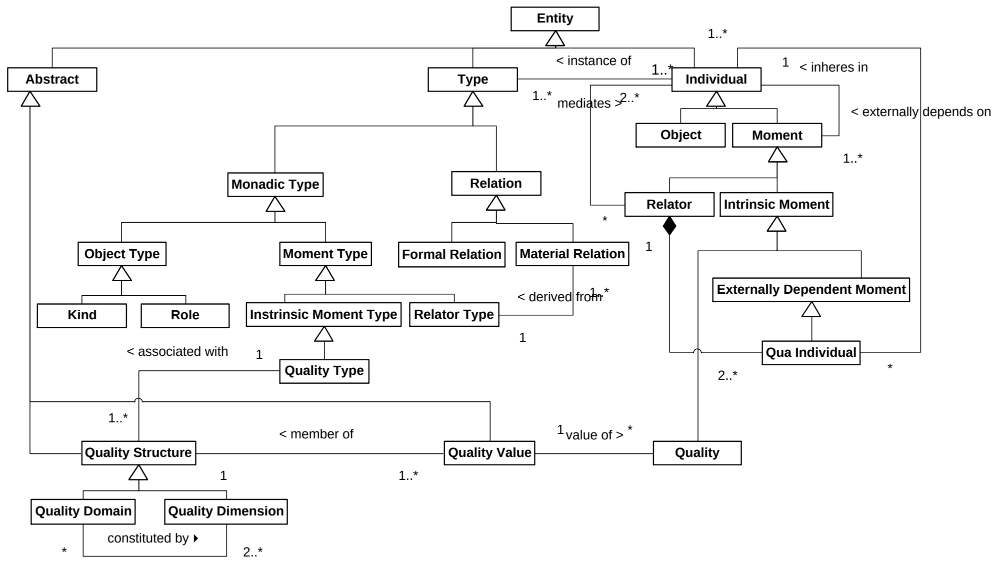

Fig.1. A Fragment of the Unified Foundational Ontology (UFO)

2.1 Objects and Moments

A fundamental distinction in this ontology is between the categories of *Individual* and *Universal*. Individuals are entities that exist in reality possessing a unique identity. Universals, conversely, are pattern of features which can be realized in a number of different individuals. The core of this ontology exemplifies the so-called *Aristotelian ontological square* or what is termed a "*Four-Category Ontology*" [8] comprising the category pairs *Object-Object Universal*, *Moment-Moment Universal*. From a metaphysical point of view, this choice allows for the construction of a parsimonious ontology, based on the primitive and formally defined notion of *existential dependence*: We have that a particular *x* is *existentially dependent* (*ed*) on another particular *y* iff, as a matter of necessity, y must exist whenever x exists. Existential dependence is a modally constant relation, i.e., if x is dependent on y, this relation holds between these two specific particulars in all possible worlds in which x exists.

The word *Moment* is derived from the german *Momente* in the writings of E. Husserl and it denotes, in general terms, what is sometimes named *trope*, *abstract particular*, *individual accident*, *mode* or *property instance*. Thus, in the scope of this work, the term bears no relation to the notion of time instant in colloquial language. Typical examples of moments are: a color, a connection, an electric charge, a social commitment. An important feature that characterizes all *moments* is that they can only exist in other particulars (in the way in which, for example, electrical charge can exist only in some conductor). To put it more technically, we say that moments are *existentially dependent* on other individuals (named their *bearers*). Existential dependence can also be used to differentiate intrinsic and relational moments: *intrinsic moments* are dependent of one single particular (e.g., color, a headache, a temperature); *relational moments* (or *relators*) depend on a plurality of individuals (e.g., an employment, a medical treatment, a marriage). A special type of existential dependence relation that holds between a moment *x* and the particular *y* of which *x* depends is the relation of *inherence* (*i*). Thus, for a particular *x* to be a moment of another particular *y*, the relation *i(x,y)* must hold between the two. For example, inherence glues your smile to your face, or the charge in a specific conductor to the conductor itself. Here, we admit that moments can inhere in other moments. Examples include the individualized time extension, or the graveness of a particular symptom. The infinite regress in the inherence chain is prevented by the fact that there are individuals that cannot inhere in other individuals, namely, *Objects*.

Examples of *objects* include ordinary entities of everyday experience such as an individual person, a dog, a house, a hammer, a car, Alan Turing and The Rolling Stones but also the so-called *Fiat Objects* such as the North-Sea and its proper-parts and a non-smoking area of a restaurant. In contrast with moments, objects do not inhere in anything and, as a consequence, they enjoy a higher degree of independence. To state this precisely we say that: an object *x* is *independent* of all other objects which are disjoint from *x*, i.e., that do not share a common part with *x*. This definition excludes the dependence between an object and its *essential* and *inseparable parts* [7], and the obvious dependence between an object and its essential moments.

To complete the Aristotelian Square, depicted in Figure 2, we consider here the categories of *object universal* and *moment universal*. We use the term universal here in a broader sense without making any *a priori* commitment to a specific theory of universals. A universal thus can be considered here simply as something (i) which can be predicated of other entities and (ii) that can potentially be represented in language by *predicative terms*. We also use the relation of *instantiation (or classification)* between individuals and universals. Object universals classify objects and moment universals classify moments. Examples of the former include Apple, Planet and Person. Examples of the latter include Color, Electric Charge and Headache. Finally, we define the relation of *characterization* between moment universals and the universals instantiated by the individuals that exemplify them: a moment universal *M* characterizes a universal *U* iff every instance of *U* bears and instance of *M*.

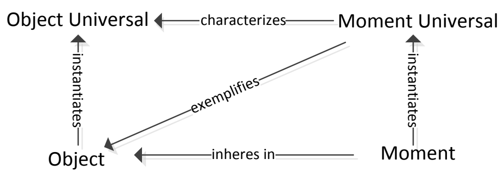

Fig.2. The ontological Square

2.2 Object Universals

Within the category of object universals, we make a fundamental distinction based on the formal notions of *rigidity* and *anti-rigidity*: A universal *U* is rigid if for every instance *x* of *U*, *x* is necessarily (in the modal sense) an instance of *U*. In other words, if *x* instantiates *U* in a given world *w*, then *x* must instantiate *U* in every possible world *w'*. In contrast, a universal *U* is anti-rigid if for every instance *x* of *U*, *x* is *possibly* (in the modal sense) not an instance of *U*. In other words, if *x* instantiates *U* in a given world *w*, then there must be a possible world *w'* in which *x* does not instantiate *U*. An object universal which is rigid is named here a ***Kind***. In contrast, an anti-rigid object universal is termed here a ***Phased-Sortal*** [7]. The prototypical example highlighting the modal distinction between these two categories is the difference between the Kind Person and the Phase-Sortals Student and Adolescent instantiated by the individual John in a given circumstance. Whilst John can cease to be a Student and Adolescent (and there were circumstances in which John was not one), he cannot cease to be a Person. In other words, while the instantiation of the phase-sortals Student and Adolescent has no impact on the identity of a particular, if an individual ceases to instantiate the universal Person, then she ceases to exist as the same individual.

In the example above, John can move in and out of the Student universal, while being the same individual, i.e. without losing his identity. This is because the principle of identity that applies to instances of Student and, in particular, that can be applied to John, is the one which is supplied by the Kind Person of which the Phase-Sortal Student is a subtype. This is always the case with Phased-Sortals, i.e., for every Phased-Sortal *PS*, there is a unique ultimate Kind *K*, such that: (i) *PS* is a specialization of *K*; (ii) *K* supplies the unique principle of identity obeyed by the instances of *PS*. If *PS* is a Phased-Sortal and *K* is the Kind specialized by *PS*, there is a *specialization condition* φ such that *x* is an instance of *PS* iff *x* is an instance of *K* that satisfies condition φ.

A particular type of Phased-Sortal emphasized in this article is what is named in the literature a ***Role***. A role *Rl* is an anti-rigid object type whose specialization condition φ is an extrinsic (relational) one. For example, one might say that if John is a Student then John is a Person who is enrolled in some educational institution, if Peter is a Customer then Peter is a Person who buys a Product *x* from a Supplier *y*, or if Mary is a Patient than she is a Person who is treated in a certain medical unit. In other words, an entity plays a role in a certain context, demarcated by its relation with other entities. This meta-property of Roles is named *Relational Dependence* and can be formally characterized as follows: A universal *T* is relationally dependent on another universal *P* via relation *R* iff for every instance *x* of *T* there is an instance *y* of *P* such that *x* and *y* are related via *R* [7].

2.3 Qualities and Quality Structures

An attempt to model the relation between intrinsic moments and their representation in human cognitive structures is presented in the theory of *conceptual spaces* introduced in [9]. The theory is based on the notion of *quality dimension*. The idea is that for several perceivable or conceivable quality universals there are associated quality dimensions in human cognition. For example, height and mass are associated with one-dimensional structures with a zero point isomorphic to the half-line of nonnegative numbers. Other properties such as color and taste are represented by multi-dimensional structures. Moreover, the author distinguishes between *integral* and *separable* quality dimensions: *"certain quality dimensions are integral in the sense that one cannot assign an object a value on one dimension without giving it a value on the other. For example, an object cannot be given a hue without giving it a brightness value (...) Dimensions that are not integral are said to be separable, as for example the size and hue dimensions.*" He then defines a *quality domain* as "a set of integral dimensions that are separable from all other dimensions" [9]. Furthermore, he defends that the notion of conceptual space should be understood literally, i.e., quality domains are endowed with certain geometrical structures (topological or ordering structures) that constrain the relations between its constituting dimensions. Finally, the perception or conception of an intrinsic aspect can be represented as a point in a quality domain. This point is named here a *quality value*.

Once more, an example of a quality domain is the set of integral dimensions related to color perception. A color quality *c* of an apple *a* takes its value in a three-dimensional color domain constituted of the dimensions hue, saturation and brightness. The geometric structure of this space (the *color spindle* [9]) constrains the relation between some of these dimensions. In particular, saturation and brightness are not totally independent, since the possible variation of saturation decreases as brightness approaches the extreme points of black and white, i.e., for almost black or almost white, there can be very little variation in saturation. A similar constraint could be postulated for the relation between saturation and hue. When saturation is very low, all hues become similarly approximate to grey.

We adopt in this work the term *quality structures* to refer to quality dimensions and quality domains, and we define the formal relation of *association* between a quality structure and an intrinsic aspect universal. Additionally, we use the terms *quality universals* for those intrinsic moment universals that are directly *associated with* a quality structure, and the term *quality* for an aspect classified under a quality universal. Furthermore, we define the relation of *valueOf* connecting a quality to its quality value in a given quality structure. Finally, we also have that quality structures are always associated with a unique quality universal, i.e., a quality structure associated with the universal Weight cannot be associated with the universal Color. This is not to say, however, that different quality structures cannot be associated with the same quality universal. For instance, with the quality universal color, we can have both the HSB (Hue-Saturation-Brightness) structure and the RGB (Red-Green-Blue) structure. In Figure 3 below, we illustrate an entity, its intrinsic color quality and the value of this quality mapped to into two different quality structures, hence, producing two different (albeit comparable) quality values.

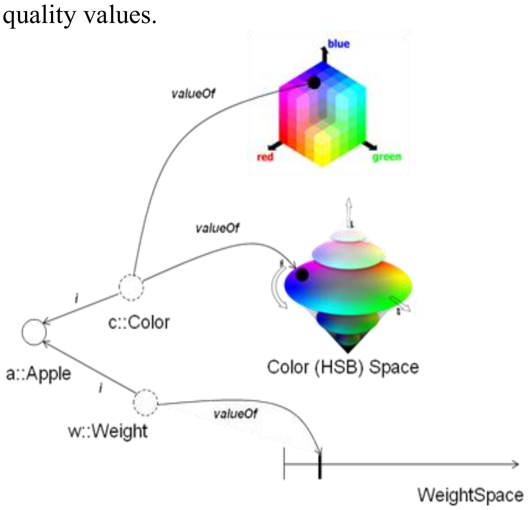

Fig.3. An object, some of its inhering qualities and the associated quality structures.

The view of qualities defended here assumes that change is substitution (as opposed to variation) of moments, i.e. "the color of *x* turned from red to brown" is represented by a red-quality of *x* that temporally precedes a brown-quality of *x*. As a consequence, we have that although a quality q can have different quality values in different quality spaces, their values in each of these structures cannot be changed. Taking this view into consideration, we elaborate further in two orthogonal partitions capturing specific characteristics of qualities which are related to aspects of temporal change of their bearers.

In the first partition, we distinguish between **necessary** (mandatory) versus **contingent** (optional) qualities; in the second, we distinguish between **immutable** versus **mutable** ones. The former distinction refers to the need for an entity to bear that property, regardless of its value. For instance, suppose that in a given conceptualization of (legal) Person, both *name* and *age* are mandatory characteristics of people (every person has a name and an age) whilst *ssn* (*social security number*) and *nickname* (*alias*) are, in contrast, optional characteristics of people. Now, notice that the relation between a person and age is a relation of *generic dependence*, i.e., the same person can bear different age qualities in different situations as long as they are all instances of the quality universal *age*. This brings us to the second of these partitions: a quality *q* is immutable to a bearer *x* of type *T* iff *x* must bear that very same quality in all possible situations in which *x* instantiates *T*. In this case, the relation between *x* and *q* is a relation of *specific dependence* (as opposed to a generic one). Again, let us suppose that, in a given conceptualization, (legal) persons cannot change their proper names. In this situation, a name would not only be a necessary but also an immutable characteristic of people. Suppose now that, in this conceptualization, that although *ssn* is an optional characteristic of people, once an *ssn* is assigned to a person, it cannot be changed. In this case, *ssn* would be an immutable and contingent quality. Finally, in this conceptualization, we assume that nicknames are both optional to people and, once assigned, can always be changed. In this case, nickname would be an example of a contingent and mutable quality.

2.5 Relators, Relations, Roles and Qua Individuals

Following the philosophical literature, we recognize here two broad categories of relations, namely, *material* and *formal* relations [7]. Formal relations hold between two or more entities directly, without any further intervening individual. Examples include the relations of *existential dependence (ed), subtype*, *instantiation*, *parthood, inherence (i)*, among many others not discussed here [7]. Domain relations such as *working at*, *being enrolled at*, and *being the husband of* are of a completely different nature. These relations, exemplifying the category of *Material relations*, have material structure of their own. Whilst a formal relation such as the one between Paul and his headache x holds directly and as soon as Paul and x exist, for a material relation of *being treated in* between Paul and the medical unit MU_1 to exist, another entity must exist which *mediates* Paul and MU_1. These entities are termed *relators*.

Relators are individuals with the power of connecting entities. For example, a medical treatment connects a patient with a medical unit; an enrollment connects a student with an educational institution; a covalent bond connects two atoms. The notion of relator is supported by several works in the philosophical literature [7] and, they play an important role in answering questions of the sort: what does it mean to say that John is married to Mary? Why is it true to say that Bill works for Company *X* but not for Company *Y*? Again, relators are special types of moments which, therefore, are existentially dependent entities. The relation of *mediation* (symbolized *m*) between a relator *r* and the entities *r* connects is a sort of (non-exclusive) inherence and, hence, a special type of existential dependence relation. It is formally required that a relator mediates at least two distinct individuals [7].

An important notion for the characterization of relators (and, hence, for the characterization of material relations) is the notion of *foundation*. Foundation can be seen as a type of *historical dependence* [7,10], in the way that, for instance, an instance of *being kissed* is founded on an individual *kiss,* or an instance of *being punched by* is founded on an individual *punch*, an instance of *being connected to* between airports is founded on a particular flight connection. Suppose that John *is married to* Mary. In this case, we can assume that there is an individual relator m_1 of type *marriage* that mediates John and Mary. The foundation of this relator can be, for instance, a wedding event or the signing of a social contract between the involved parties. In other words, for instance, a certain event e_1 in which John and Mary participate can create an individual marriage m_1 which existentially depends on John and Mary and which mediates them. The event e_1 in this case is the foundation of relator m_1.

Now, let us elaborate on the nature of the relator m_1. There are many intrinsic moments that John acquires by virtue of being married to Mary. For example, imagine all the legal responsibilities that John has in the context of this relation. These newly acquired properties are intrinsic moments of John which, therefore, are existentially dependent on him. However, these moments also depend on the existence of Mary. We name this type of moment *externally dependent moments*, i.e., externally dependent moments are intrinsic moments that inhere in a single individual but are existentially dependent on (possibly multiple) other individuals. The individual which is the aggregation of all externally dependent moments that John acquires by virtue of being married to Mary is named a *qua individual* (in this case, John-qua-husband-of-Mary). A qua individual is, thus, defined as an individual composed of all externally dependent moments that inhere in the same individual and share the same foundation. In the same manner, by virtue of being married to John, Mary bears an individual Mary-qua-wife-of-John.

The notion of qua individuals is the ontological counterpart of what has been named *role instance* in the literature [11] and represent the properties that characterize a particular mode of participation of an individual in a relation. Now, the entity which is the sum of all qua individuals that share the same foundation is a relator. In this example, the relator m_1 which is the aggregation of all properties that John and Mary acquire by virtue of being married to each other is an instance of the relational property *marriage*. The relation between the two qua individuals and the relator m_1 is an example of formal relation of parthood [7].

The relator m_1 in this case is said to be the *truthmaker* of propositions such as "John is married to Mary", "Mary is married to John", "John is the husband of Mary", and "Mary is the wife of John". In other words, material relations such as *being married to*, *being legally bound to*, *being the husband of* can be said to hold for the individuals John and Mary because and only because there is an individual relator marriage m_1 mediating the two. Thus, as demonstrated in [7,10], material relations are purely linguistic/logical constructions which are founded on and can be completely derived from the existence of relators. In fact, in [7], we have defined a formal relation of derivation (symbolized as *der*) between a relator type (e.g., Marriage) and each material relation which is derived from it.

Finally, there is an intimate connection between qua individuals and *role types*: let T be a natural type (kind) instantiated by an individual *x*, and let *R* be a role type specializing *T*. We have that there is a qua individual type *Q* such that *x* instantiates *R* iff *x* bears an instance of *Q*. Alternatively, we have that for every role type *R* there is a relator type *RR* such that *x* instantiates *R* iff *x* is mediated by an instance of *RR*. Note that this conforms to the formal property of roles as *relationally dependent* types.

The summary of the discussion promoted in this section is illustrated in Figures 4 to 6. Figure 4, illustrates the inherence relation between John and his externally dependent moments which are existentially dependent on Mary (as well as analogous relations in the converse direction). In figure 5, John instantiates the role type Husband (which is a specialization of the kind (Male) Person) iff there is a qua individual John-qua-husband-of-Mary which inheres in John. Moreover, this figure illustrates that the qua individuals John-qua-husband-of-mary and Mary-qua-wife-of-John are mutually existentially dependent. In other words, John cannot be the Husband of Mary without Mary being the wife of John. Finally, Figure 6 shows that the material relation *married to* is derived from the relator type Marriage and, thus, tuples such as <John,Mary> and <John,Mary> are instances of this relation iff there is an instance of Marriage that mediates the elements of the tuple.

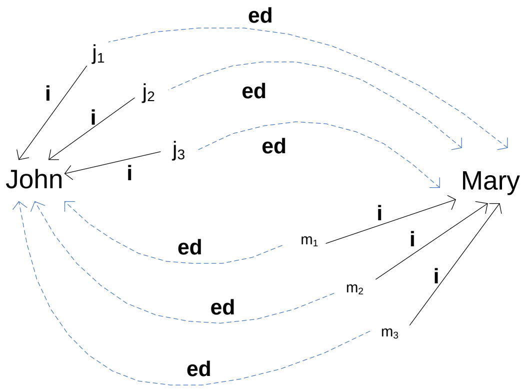

Fig.4. Objects and their inhering externally dependent moments: in this example, the object bears a number of moments (j_1,j_2,j_3), which inhere (i) in John but which are also existentially dependent (*ed*) on Mary. Mutatis Mutandis, the model depicts a number of moments of Mary (m_1,m_2,m_3), which inhere in Mary but which are also existentially dependent on John.

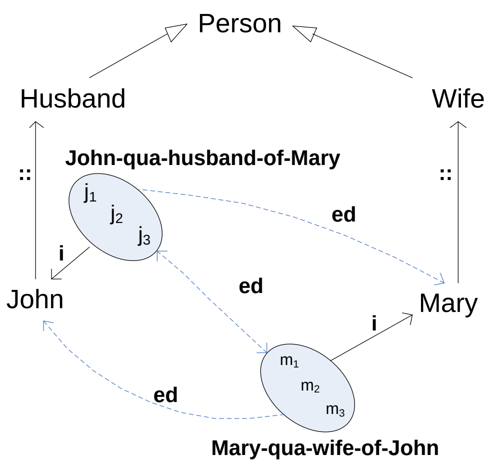

Fig.5. Objects, their instantiating roles and their inhering qua individuals: in this example, John and Mary instantiate (::) the roles Husband and Wife, respectively, in virtue of the qua individuals that inhere (*i*) in them. These roles are specializations of the type Person ($\rightarrow$). Moreover, *John-qua-husband-of-Mary* (which is an aggregations of the moments j_1, j_2 and j_3) is mutually existentially dependent (*ed*) on *Mary-qua-wife-of-John* (an aggregation of moments m_1, m_2 and m_3).

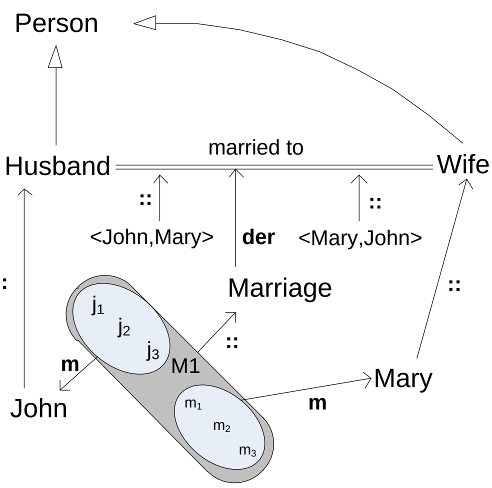

Fig.6. Material Relations are founded on relators that mediate their relata: in this example, the marriage relator M_1 between John and Mary mediates (*m*) these two entities by virtue of being existentially dependent on both of them. This relator is an aggregation of the qua individuals *John-qua-husband-of-Mary* and *Mary-qua-wife-of-John* (represented by the two ellipses). Moreover, M_1 is the foundation for the tuples <John,Mary> and <Mary,John>, which instantiate (::) the material relation *married to*, which, in turn, is derived (*der*) from the relator universal Marriage which M_1 instantiates.

3. Using a Foundational Ontology to design a Well-Founded Conceptual Modeling Language

In this section, we present a Conceptual Domain Modeling language termed OntoUML [7]. OntoUML is an example of a conceptual modeling language whose metamodel has been designed to comply with the ontological distinctions and axiomatization of the UFO foundational ontology. This language and its foundations are currently being considered as candidates to contribute to a response to the SIMF Request for Proposal [1].

The OntoUML metamodel contains: (i) elements that represent ontological distinctions prescribed by an underlying foundational ontology; (ii) constrains that govern the possible relations that can be established between these elements. These constraints, which are derived from the axiomatization of the ontological theory, restrict the ways in which the modeling primitives can be related. The goal is to have a metamodel such that all grammatically correct specifications according to this metamodel have logical models that represented *intended state of affairs* of the underlying conceptualization [5].

For instance, the language has modeling primitives to explicitly represent the notions of *kinds*, *subkind* and *roles* as well as the notions *quality* and *relator* previously discussed. Kinds and subkinds are represented by the corresponding stereotypes «kind» and «subkind». In an analogous manner, roles are represented by the stereotype «role». In the axiomatization of the UFO ontology we have that anti-rigid types cannot be a supertype of rigid one [7]. So, as an example of formal constraint in this language, we have that classes stereotyped as «kind» or «subkind» cannot appear in an OntoUML model as a subtype of class stereotyped as «role».

As discussed at length in [12], quality universals are typically not represented in a conceptual model explicitly but via *attribute functions* that map each of their instances to points (quality values) in a quality structure. Accordingly, the *datatype* associated with an attribute A of class C is the representation of the quality structure that is the co-domain of the attribute function represented by A. In other words, a quality structure is the ontological interpretation of the (Onto)UML datatype construct. Moreover, we have that a multidimensional quality structure (quality domain) is the ontological interpretation of the so-called *structured datatypes*. Quality domains are composed of multiple integral dimensions. This means that the value of one dimension cannot be represented without representing the values of others. The fields of a datatype representing a quality domain QD represent each of its integral quality dimensions. Alternatively, we can say that each field of a datatype should always be interpreted as representing one of the integral dimensions of the QD represented by the datatype. The constructor method of the datatype representing a quality domain must reinforce that its tuples always have values for all the integral dimensions. Finally, an algebra (as a set of formal constraints) can be defined for a datatype so that the relations constraining and informing the geometry of represented quality dimensions are also suitably characterized.

There are, nonetheless, two situations in which one might want to represent quality universals explicitly. The first of these is when we want to represent that a quality might be projected to different quality spaces (i.e., the underlying quality universal is associated with alternative quality structures). This idea is represented in figure 7. In this case, the color quality aggregates the different values (in different quality spaces) that can be associated with that object (the apple, in this case). Notice that, in these situations, it is as if the color quality is representing a certain 'aspectual slice' of the Apple, or the *Apple-qua-colored object*.

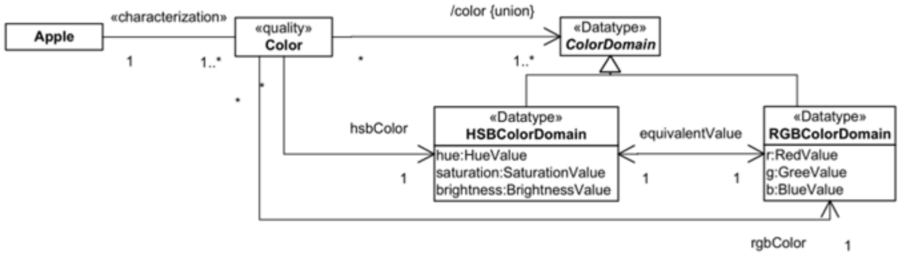

Fig.7. Representing quality types which can be associated to multiple quality structures [12]

A second situation in which one might want to represent qualities explicitly is when modeling a temporal perspective on qualities. This is illustrated in Figure 8 below. In that model, we have different color qualities (with different associated quality values) inhering in a given apple in different situations.

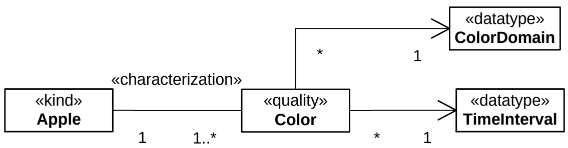

Fig.8. Temporal change in properties as quality replacement

As illustrated in Figures 7 and 8, in this language, one can employ the stereotype «quality» to explicitly represented quality universals and a stereotyped relation of «characterization» to represent its ontological counterpart. As discussed in section 2, the *characterization* relation between an intrinsic moment universal and the universal it characterizes is mapped at the instance level onto an *inherence* relation between the corresponding individual moments and their bearers. That means that every instance *m* of a class *M* stereotyped as «quality» is existentially dependent of an individual *c*, which is an instance of the class *C* related to *M* via the «characterization» relation. Inherence has the following characteristics: (a) it is a sort of existential dependence relation; (b) it is a binary formal relation; (c) it is a functional relation. These three characteristics impose the following metamodel constraints on the «characterization» construct: by (a) and (c), the association end connected to the characterized universal must have the cardinality constraints of one and exactly one; by (a), the association end connected to the characterized universal must have the meta-attribute (isReadOnly = true); «characterization» associations are always binary associations.

Regarding mutability/immutability and necessity/contingency of qualities, we use the following representation strategy. The necessity of a given quality (or, consequently, of a given quality value) is represented by a minimum cardinality ≥ 1 in the association end connected to the «quality» class (or the association end connected to the associated datatype). Alternatively, a contingent quality is represented by a minimum cardinality = 0 in the referred association end. The immutability of a quality (or the corresponding quality value) is represented by using the tagged value *readOnly* applied to the referred association end (i.e., by making its meta-attribute *isReadOnly = true*). Finally, the absence of this tagged value in a given association end indicates that a mutable quality (quality value) is being represented.

Finally, in the language, the stereotype «relator» is used to represent the ontological category of relator universals. As discussed in section 2, a relator particular is the actual instantiation of the corresponding relational property. Material relations stand merely for the facts derived from the relator particular and its mediating entities. In other words, relations are logico-linguistic constructions which supervene on relators. Therefore, as argue at length in [7,10], the representation of the relators of material relations must have primacy over the representation of the material relations themselves. In this paper, we simply omit the representation of material relations.

In the sequel, we provide a final example of formal constraints incorporated in the OntoUML metamodel which is derived from its underlying ontological foundations. Relators are existentially dependent entities. Thus, as much as a characterization relation, mediation is also a directed, binary, existential dependence relation. As consequence, we have that a relation stereotyped as «mediation» in OntoUML must obey the following constraints: (i) the association end connected to the mediated universals must have the cardinality constraints of at least one; (ii) the association end connected to the mediated universals must have the meta-attribute (isReadOnly = true); (iii) «mediation» associations are always binary associations. Moreover, since a relator is dependent (mediates) on at least two numerically distinct entities, we have the following additional constraint (iv) Let R be a class representing a relator universal and let $\{C_1 \ldots C_n\}$ be a set of classes mediated by R (related to R via a *mediation* relation). Finally, let $lower_{Ci}$ be the value of the minimum cardinality constraint of the association end connected to C_i in the mediation relation. Then, $(\sum_{i=1}^{n} lower_{Ci}) \geq 2$.

3.1 Discussion

As shown in [7], the distinction among rigid and anti-rigid object types incorporated in the OntoUML language provides for a semantically precise and ontologically well-founded semantics for some of the much discussed but still *ad hoc* distinctions among conceptual modeling constructs. Since its first proposal in this line of work [13], this distinction has had an impact in conceptual model validation [14], in the discovery of important ontological design patterns [13], as well as in the formal and ontological semantics of derived types in conceptual modeling [15]. Moreover, it has influenced the evolution of other conceptual modeling languages, such as ORM 2.0 [16]. Finally, as argued in [7,17], this distinction has a direct impact even in the choice of different design alternatives in different implementation environments.

Analogously, the explicit representation of intrinsic moments and quality structures in the language allows for providing an ontological semantics and clear modeling guidelines for attributes, datatypes, weak entities and domain formal relations [10,12]. Moreover, the model presented in Figure 7 illustrates a design pattern for *modeling properties associated to alternative quality structures*. Since its first proposal in [12], this pattern has been applied in several domains (e.g., [18]).

Finally, the strategy for the representation of material relational properties discussed here has been applied in a series of publications to address a number of important and recurrent conceptual modeling problems. For instance, in [10], it was used to address the problem of the collapse of cardinality constraints in material relations; in [19], it has used as an integral part in the development of a solution the so-called problem of transitivity of parthood relations; in [20], in an industrial case study of ontology reverse engineering, the systematic identification of the material content of relations (i.e., relators) was reported as a fruitful technique for knowledge elicitation when interacting with domain experts; in [21], the ontological theory of relations underlying this approach has been used to disambiguate the semantics of two fundamental modeling constructs in Goal-Oriented Requirements Engineering; finally, in [22], the same theory has been employed to provide ontological semantics and clear modeling guidelines for disambiguating the constructs of association specialization, association subsetting and association redefinition in UML 2.0.

Because the distinctions and constraints comprising this language are explicitly and declaratively defined in the language metamodel, they can be directly implemented using metamodeling architectures such as the OMG's MOF (Meta Object Facility). Following this strategy, [7] reports on an implementation of an OntoUML graphical editor by employing a number of basic Eclipse-based frameworks such as the ECore (for metamodeling purposes) and MDT (for the purpose of having automatic verification of OCL constraints). An interesting aspect of this strategy is that, once the ontological constraints have been incorporated in the metamodel, they give rise to syntactical constraints. These constraints in the language metamodel, thus, limit the set of grammatically correct models that can be produced using the language to those whose instances only represent consistent state of affairs according to the underlying ontology.

4 From an Ontology-Driven Conceptual Domain Model to a Computationally-Driven Specification in OWL

4.1 Temporally Changing Information in Conceptual Domain Models

The model of Figure 9 below (termed as *running example* in the remainder of this section) illustrates some important aspects related to change that should be highlighted in the discussion that follows. This model represents a situation in which a person, who can be a man or a woman, is identified by his/her name. Moreover, he/she can have a social security number (ssn) that cannot change. He/she has an age that change annually, and can also be referred by one or more nicknames that may change along his/her life. Finally, a man can get married to only one woman per time (and vice-versa), thus, becoming husband and wife, respectively.

We distinguish here three sources of changes: attributes, relations and type instantiation. Regarding intrinsic properties, we can classify them under the dimensions of **necessary** (mandatory) versus **contingent** (optional), and **mutable** versus **immutable** as discussed in section 2. Furthermore, the generic dependence between the Kind Person and Quality Universal age is also present in the relation *marriedTo* between Husband and Wife (and vice-versa). In other words, both association ends of the relation *marriedTo*, albeit mandatory for the associated types (Husband and Wife), are mutable. Finally, we have a third source of change in this model, related to the anti-rigidity of the role universals Husband and Wife. As previously discussed, a particular man instantiates the Role Husband contingently and when mediated by a particular marriage relator. *Mutatis Mutandis*, the same can be said for the role Wife.

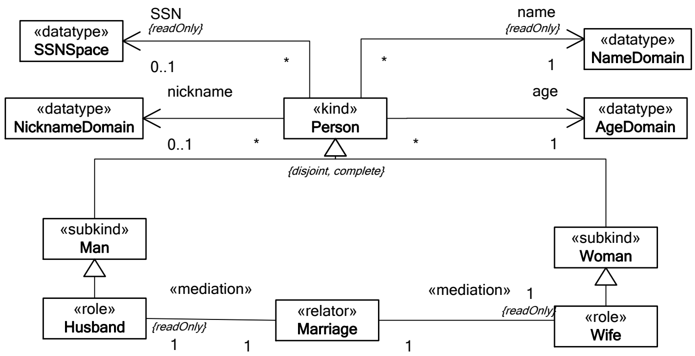

Fig.9. An OntoUML Conceptual Domain Model with sources of temporal change

4.2 OWL

The OWL (Web Ontology Language) is a well known formal language for representing ontologies on the Semantic Web. In this work, we are particularly interested in its DL based variants, which we refer simply as OWL in the remainder of this text. DL consists of a family of subsets of classical first order logics that is designed focusing on decidable reasoning. Using DL-based languages, one is able to represent static scenarios with immutable truth-values such that the information about the domain can be completed but cannot be really changed. In particular, the instantiation of a class or property cannot be retracted, except through external intervention. For example, once a model represents that *John being 28 years old* instantiates the class *Husband*, this information cannot be changed.

DL has two important characteristics to be taken into account here, namely, open world assumption (OWA) and monotonicity. The former entails that what is stated in the model is true but not necessarily the entire truth about a domain, i.e., new information can always be discovered about the represented entities. However, a monotonic logical system is such that the addition of new information/premises must not interfere in the information that has been previously derived. Consequently, what is true in one situation must remain true regardless of any addition of information to the model.

In an OWL model, we can codify the distinction between mandatory versus optional properties (represented by cardinality constraints). However, we cannot represent the distinction between immutable versus mutable, neither the one between rigid versus anti-rigid types. Due to the aforementioned monotonicity of the language, all stated relations, attribute assignments and classification assignments become immutable. In order to circumvent these limitations, a number of authors have been investigating different strategies for representing temporally changing information in OWL [24, 25].

Most of these approaches employ a strategy which consists of interpreting all enduring entities in a domain model (e.g., objects, qualities, relators) as events (processes). This view is grounded in a philosophical stance named *Perdurantism* [26,27]. In a perdurantistic view, a domain individual is seen as a 4D (four-dimensional) "space-time worm" whose temporal parts are slices (snapshots) of the worm. As argued in [6], although such a view can be accurate in representing the current state of knowledge in natural sciences, the distinction between enduring and perduring entities is a fundamental cognitive distinction, present both in human cognition and language. For this reason, as argued in [5], we advocate that such a distinction should be explicitly considered both in conceptual modeling languages as well as in their underlying foundational ontologies. Moreover, besides the philosophical controversy associated with perdurantism, there are a number of issues triggered by such 4D-driven approaches which can become prohibitive for certain design scenarios. Some of these issues are discussed in the next section and are addressed by an alternative approach considered in this article, namely, a reification-driven approach.

Property-reification is definitely not a new idea. In fact, it is a well-know solution for representing temporal information in knowledge representation going back at least to the eighties. Despite this, and despite some clear advantages of this approach for certain design problems, this solution is dismissed in [24] for the lack of an ontological interpretation (or ontological counterpart) for the reified properties. In the next section, we demonstrate that: (i) the ontological categories underlying OntoUML provides for a direct ontological interpretation for these reified entities in the proposed approach; and (ii) these categories can be directly employed for creating transformation patterns between OntoUML models and OWL specifications, in which at least part of the original modal semantics is retained.

4.3 Reifying Temporal Knowledge in OWL supported by Ontological Categories

Reification is an operation that makes the reifed (objectified) entity amenable to reference, qualification and quantification. In [28], Quine presents reification as a strategy for forging links between sentences or clauses represented in a first order logic (FOL) language. For example, the sentence 'Sebastian walked slowly and aimlessly in Bologna at *t'* can be reified as $\exists x$ (x is a walk and x is slow and x is aimless and x is in Bologna and x is at t and x is by Sebastian) where x is the objective reference that connect all clauses.

In this section, we are particularly interested in reification as a strategy for representing temporal knowledge using DL-based versions of OWL. It means that we are restricted to a subset of FOL whose predicates are at most binary. For example, the statement 'John is married to Mary at *t*' is to be reified as something like $\exists x$ (isRelatedTo(x, John) $\wedge$ isRelatedTo(x, Mary) $\wedge$ holds(x, t)). Indeed, in face of this representation some questions arise: what is this thing that is related to *John* and *Mary*? Can this thing keep existing (holding) without being related to both *John* and *Mary*? Are *John* and *Mary* related to each other in the very same way?

In the sequel, we employ the ontological notions defined in section 2 to answer such questions and to provide ontological meaning for the reified temporal knowledge. More specifically, we intend to reify/objectify the individuals' properties and to attribute to them the time interval during which they hold having a certain value. For example, the time interval during which John has the age of 27 years old, or the one during which he is married to Mary.

As mentioned, reifying the properties of an individual allows one predicating and quantifying over them. It includes attributing to them a time interval during which it is held to be true. Thereby, in Figure 10, we present two illustrative schemas of applying an ontologically-grounded reification approach to the running example in a temporal view. The object and moment individuals are represent by graphical elements in different shapes, whose projection onto the timeline corresponds to the individual's temporal extension. Moreover, the spatial inclusion of elements represents the *inherence* relation (i.e. the spatially included elements *inhere* in the container) and also reflects the temporal inclusion imposed by the existential dependence. The mandatory properties are represented as rectangles, while the optional properties are represented as rounded corner rectangles. Moreover, the mutable properties are in a lighter grey shade than those immutable ones.

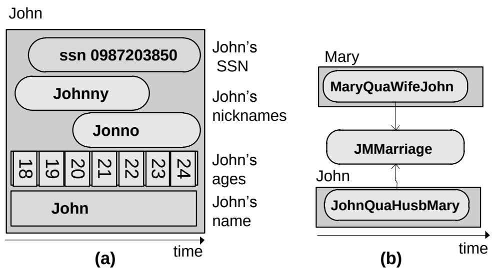

Fig.10. A schematic representation of an object with (a) its reified qualities; (b) representing reified relators and qua individuals

In Figure 10a, the larger rectangle represents the object individual *John* that is an instance of the class *Person*; the elements contained in that rectangle represent the qualities corresponding to the reification of John´s attributes. Particularly, the quality *John's name* has the same width extension than the individual *John*, representing that it has the same temporal extension of John. In contrast, the necessary and mutable attribute *age* is represented by many qualities (*John's ages*) that together must have the same width extension than the individual *John*. The Figure 10b represents the founding relator of the material relation *marriedTo* between the object individuals *John* and *Mary*, as well as the reification of the correspondent role instantiations (qua individuals). The relator that mediates the couple is represented by the rounded corner rectangle identified as *JMMarriage*, and the qua-individuals that compose it are represented by the elements connected to it by an arrow.

In Figure 11, we propose a framework that reflects the ontological notions presented in section 2 and allows for representing temporal information in OWL. Every individual *has a temporal extent;* individuals are specialized into *moments* and *objects*; a *moment* is *existentially dependent of* at least one individual, and can be either a *relator* or an *intrinsic moment*. The former *mediates* two or more individuals, whilst the latter *inheres in* exactly one individual and can be either a *quality* or a *qua-*

individual; a *quality* has one *datatype value;* a *qua individual* is *part of* one *relator* and is *existentiallyDependentOf* at least another qua-individual. The relations *inheresIn*, *mediates* and *partOf* are specializations of *existentiallyDependentOf*.

Following, the model of Figure 9 will be used as support for explaining a number of methodological guidelines discussed in the sequel which can systematically be used to specialize the framework represented in Figure 11.

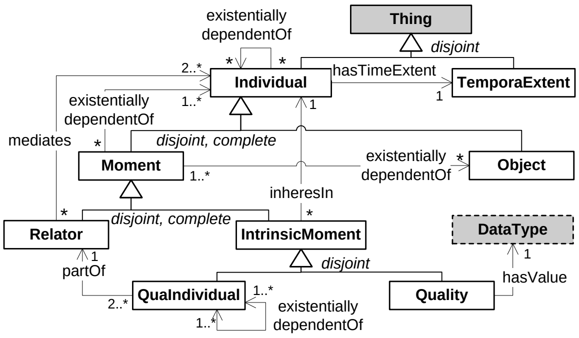

Fig.11. An Ontology-based Framework for the systematic reification of properties in OWL. The classes depicted in gray are original OWL constructs which are then specialized by elements of the proposed framework (whose classes are depicted in white).

a. The rigid/**necessary classes** (e.g. *Person*) should specialize the class ***Object***.
b. The anti-rigid/**contingent classes** (roles) should be represented as subclasses of the class ***QuaIndividual***. This qua individual type classifies all the qua-individuals resulting from the reification of the participation of individuals of a same object class in a same material relation. For example, the class *Husband* is represented as the class *QuaHusband*, which group all the qua-individuals resulting from the reification of the participation of *Man's* individuals in the material relation *marriedTo*.
c. The **material relations** of the domain should be explicitly represented as subclasses of class ***Relator***. This relator types classifies all the relator individuals resulting from the reification of the same material relation. For example, the material relation *marriedTo* is represented as the *Marriage* class;
d. **Attributes** should be represented as subclasses of the class ***Quality***. A quality type classifies all the qualities resulting from the reification of a certain attribute of individuals of the same type. For example, the attribute *name* of the concept *Person* is represented as the class *Name*, which classifies all the quality individuals resulting from the reification of the instantiation of the attribute name of individuals of the class *Person*.

Moreover, we must restrict which and how properties can be or must be applied over the classes. We use the terms minC, maxC and exacC for referring to the minimum, maximum and exact values of cardinality holding for attributes or relation association ends, respectively.

a. every instance of a **qua-individual** class must ***inheresIn*** exactly one individual of the correspondent object class and only *inheresIn* it. For example, any individual *quaHusband* must *inheresIn* exactly one instance of *Man* and cannot *inheresIn* anything else;

b. every instance of a **qua-individual** class must be ***partOf*** exactly one individual of the correspondent relator class and only be *partOf* it. For example, any individual *quaHusband* must be *partOf* exactly one instance of *Marriage* and cannot be *partOf* anything else;

c. every instance of a **qua-individual** class must be ***existentiallyDependentOf*** all other qua-individuals participating in the same relation. For example, any individual *quaHusband* must be ***existentiallyDependentOf*** all other qua-individuals that are part of the relator *Marriage* and cannot be ***existentiallyDependentOf*** anyother qua-individual;

d. every instance of a **relator** class must ***mediates*** only individuals of the correspondent object classes (e.g. an individual of the class *Marriage* must ***mediates*** only instances of the classes *Man* or *Woman*);

e. every instance of a **relator** class must have as part (***inverse partOf***) only individuals of the qua-individual classes that inhere in the individuals of object classes that the relators mediate (e.g. any individual of the class *Marriage* must have as part only instances of the classes *QuaHusband* or *QuaWife.* These qua individuals inhere in individuals of the classes *Man* and *Woman* mediated by individuals of the class *Marriage*);

f. every instance of a **relator** class must have as part (***inverse partOf***) at least ***minC***, at most ***maxC*** or exactly ***exactC*** instances of the correspondent qua-individual classes (e.g. any individual of the class *Marriage* must be part of exactly one instance of the class *Man* and exactly one instance of the class *Woman*);

g. every instance of a **relator** class must ***mediate*** at least ***minC***, at most ***maxC*** or exactly ***exactC*** instances of the correspondent object classes (e.g. any individual of the class *Marriage* must mediate exactly one instance of the class *Man* and exactly one instance of the class *Woman*);

h. for **immutable material relation**, the domain individuals must be mediated by (***inverse mediates***) at most ***maxC*** or ***exactC*** instances of the **relator** class. Otherwise, if it is **mutable**, **no cardinality restrictions** are imposed to the number of **relators** mediating the domain individuals (***inverse mediates***);

i. every instance of a **quality** class must ***inheresIn*** exactly one individual of the correspondent object class and only *inheresIn* it. For example, any individual *Name* must *inheresIn* exactly one instance of *Person* and cannot *inheresIn* anything else;

j. every instance of a **quality** class must ***hasValue*** exactly one value of the correspondent datatype and *only it*. For example, any individual *Name* must *hasValue* exactly one *String* value and cannot be related via *hasValue* to anything else;

k. for **necessary attributes**, every instance of the correspondent object class must bear (***inverse inheresIn***) at least one instance of the **quality** class. Otherwise, for **contingent attributes**, the minimum cardinality is not restricted. For example, every instance of the class *Person* must have at least one instance of the quality *Age* inhering in it, whilst such restriction does not hold for the quality *SSN*.

l. for **immutable attributes**, every instance of the correspondent object class must bear (***inverse inheresIn***) at most maxC or exactly exacC instances of the **quality** class. In contrast, for **mutable attributes**, the maximum cardinality is not restricted. It means that every time the attribute changes, a new *quality* individual is necessary for holding the new value. For example, every instance of the class *Person* must have at most one instance of the quality *SSN* inhering in it, whilst such restriction does not hold for the quality *Age*.

Figure 12 depicts an implementation of the running example following the proposed reification approach. Notice that possible instantiations of this model are the situations illustrated by Figures 10.a and 10.b.

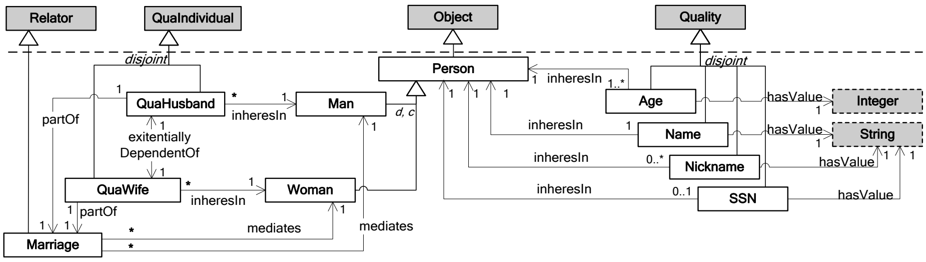

Fig.12. Mapping the model of Figure 9 to OWL using the framework of Figure 11. In this model, the domain independent classes specializing Thing in figure 11 (i.e., the classes proposed in our framework representing different ontological categories of individuals) are depicted in full-lined grey boxes; domain-specific classes extending those are represented in full-lined white boxes. Finally, specializations of the OWL construct datatype are represented in dashed grey boxes.

4.4 Discussion

In a logical theory representing a conceptual model, time-indexed properties are often represented introducing a temporal parameter in the instantiation relation, i.e. at *t*, *x* is an instance of the property *P*. There are at least three different interpretations of this temporalization: (i) 'at *t*' is a modal operator that applies to propositions, like 'at *t* (*x* is red)'; (ii) *t* is just an additional argument that transforms unary properties in binary ones (i.e. in relations) like '*x* is red-at-*t*'; (iii) 'at *t*' is a modifier of the particular, i.e., '*x*-at-*t* is red', where '*x*-at-*t*' is, in a four dimensional view, the temporal slice of *x*.

Both option (i) and a solution somewhat similar to option (ii) are widely used in conceptual modeling (see Figures 13.a and 13.b, respectively). Option (iii) can be found in some novel proposals in data modeling (see, for instance, [29]). The view defended here allows for an alternative representation, which is similar but not equivalent to (iii). As previously discussed, this alternative view assumes that a change in an endurant is given by a substitution of moments, i.e., the temporal information is coded in the temporal extension of moments. This solution could be easily represented in Figure 8 without adding complexity to the definition of intrinsic properties. Additionally, this solution has two benefits when compared to (iii). First, as opposed to (iii), one does not necessarily commit to four-dimensionalism, since moments can be conceived as persisting entities in the same way as substantial individuals (objects). In other words, in the alternative proposed in this paper, one does not have to assume the existence of temporal slices of moments. Second, in a (Onto)UML class diagram for conceptual modeling, classes are supposed to represent persisting objects such as Apple in (iii), not snapshots of objects such as AppleSnapshot in the same model. Snapshots of objects that instantiate the types depicted in a class diagram are supposed be represented via instance diagrams.

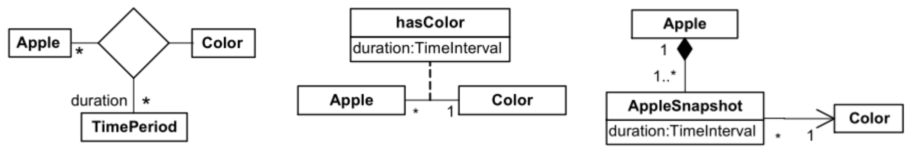

Fig.13. Different strategies for representing temporally changing information in Conceptual Modeling: (a-left) time modality; (b-center) time-indexed relations (c) entity snapshots.

Regarding the conceptual representations in Figure 13, notice that neither (a) nor (b) could be directly represented in OWL since: (i) OWL cannot represent ternary relations; (ii) OWL properties cannot have properties themselves. Regarding (c), in [25], we have proposed two alternative approaches for representing temporal information in OWL following a 4D (perdurantist) view. In both approaches, we divide the entities in two levels: individual concepts level, for the properties that do not change, and time slice level, for registering the changes on mutable properties. Although that proposal allows one to reasonably represent the intended models, those approaches have the following drawbacks:

- proliferation of time slices: any change occurred in a certain time slice leads to what we call a *proliferation of time slices*. It means that it is necessary to duplicate every time slice in the chain of connected instances that includes the instance on change;
- oddity in ontological interpretation of contingent concepts: in 4D approaches the anti-rigid classes are classes that apply only to time-slices, whilst the rigid classes apply both to 3D entities (ordinary objects, qualities and relators) and their time-slices. This makes the ontological interpretation for the anti-rigid classes (like *Husband* and *Wife*) rather odd;
- repetition of the immutable information on time slice level: the properties that are immutable but not necessary are represented at the time slice level. This leads to a tedious repetition of this information across the time slices of the same individual concept;
- not guaranteeing immutability in the time slice level: since the immutable properties represented at time slice level must be repeated across the time slices of the same individual concept, we cannot guarantee that this property value does not change across time slices.

If we compare the reification approach proposed here with these 4D-based proposals, the following can be stated regarding the aforementioned drawbacks:

- *proliferation of (time-slice) individuals*: changes no longer cause proliferation of individuals. Although we do have, in this case, the need for new (reified) individuals, the number of these individuals do not increase for each change. For this reason, under this respect, we consider the reification proposal more scalable than the 4D ones;

- *oddity in the ontological interpretation of contingent concepts*: we have homogeneous ontological interpretation for necessary and contingent concepts in the reification proposal;
- *repetition of the immutable contingent information:* except for the mutable properties, no other property is repeated in the reification proposal;
- *not guaranteeing the immutability of contingent properties*: since the immutable properties are represented just once in the reification proposal, its value cannot change.

It is important to highlight that, despite these benefits, there are also limitations and drawbacks in the reification approach. For instance, as pointed out in [24], when reifying relations, we lose the ability to (directly) associate with them meta-properties such as symmetry, reflexivity, transitivity and functionality. However, as discussed in depth in [7], the application of these meta-properties to material relation is far from a trivial issue. For instance: (i) material relations are never reflexive (since relators must mediate at least two distinct individuals); (ii) symmetry has to differentiate extensional symmetry from intentional symmetry (which can properly be represented here by the roles associated with relators); (iii) transitivity of material relations is an issue of great complexity which has been partially treated, for example, in [19], for the case of parthood relations.

In any case, this discussion highlights our argument in section 1 that there is not one single design solution that should fit all design problems. This is by itself enough a good reason for separating conceptual domain modeling from the multiple implementations which can be derived from it and which can be chosen for maximizing specific sets of non-functional requirements.

Finally, although we are aware of initiatives for addressing time representation and reasoning in OWL, we deemed this issue out of scope for this particular paper. However, having a proper axiomatization in that respect is necessary for imposing the temporal restrictions pointed out in our reification proposal, namely: (i) the existential dependence relation must imply temporal inclusion of the dependent individual in the time-extent of the individual(s) it depends on; (ii) a reified necessary and immutable property must have exactly the same time-extent of the individual it depends on; and (iii) a reified necessary and mutable property must have the temporal projection of all its individuals equal to the time-extent of the individual they depend on (i.e. the property age). These issues should be properly dealt with in a fuller approach.

5 Final Considerations

To promote semantic interoperability between information models (and applications which depend on them), we need to be able to guarantee truthfulness to the domain in reality being represented in these models (intra-model consistency). Moreover, we need to guarantee that we establish the correct relations (with the correct semantics) between elements pertaining to different models (inter-model consistency). In order to achieve these objectives, we must rely on representation mechanisms that are able to make explicit its ontological commitments and which are able to capture the subtleties of the subject domains being represented. Moreover, this should be done in a manner that are consistent with how humans as cognitive agents construct and shared their mental models of those subject domains. After all, tasks such as domain understanding, problem-solving and meaning negotiation are meant to be performed by human agents in these scenarios.

Following a tradition on what is now termed *Ontology-Driven Conceptual Modeling*, we argue in this article that these representation mechanisms should be grounded in Foundational Ontologies. In this paper, we present an ontological theory which is built on the idea of property-instances (tropes, moments, modes). This idea affords an ontology which has an illustrious pedigree in philosophy and which has corresponding support both in cognition and language. Moreover, this idea can provide an ontological interpretation and can be used to derive modeling guidelines to many conceptual modeling constructs (e.g., weak entities, reified attributes and associations, datatypes). Finally, as demonstrated in this paper, this idea provides a modeling framework for systematically representing temporally changing information in a class of description-logics/frame-based languages, represented here by the language OWL.

Acknowledgments. This research is supported by FAPES (PRONEX grant #52272362) and CNPq (productivity grant #311578/2011-0). Moreover, the authors would like to thank Claudio Masolo and Stefano Borgo for the fruitful discussions regarding “change as moment replacement” and, in particular, for important insights regarding the representation of temporal information in conceptual modeling as discussed in section 4.4.

References

1. Object Management Group, Semantic Information Model Federation (SIMF): Candidates and Gaps, online: http://www.omgwiki.org/architecture-ecosystem/.
2. Mylopoulos, J., Conceptual modeling and Telos. Conceptual Modeling, Databases, and CASE (Chapter 2, pp. 49–68). Wiley, 1992.
3. Wand, Y., Weber, R., An ontological evaluation of systems analysis and design methods, Information System Concepts: An In-Depth Analysis. Elsevier Science Publishers B.V., North-Holland, 1989.
4. Recker, J., Rosemann, M., Indulska, M., P. Green, Do Ontological Deficiencies in Process Modeling Grammars Matter?, MIS Quarterly, Vol. 35, No. 1, pp. 57–80, 2011.
5. Guizzardi, G., On Ontology, ontologies, Conceptualizations, Modeling Languages, and (Meta)Models, Frontiers in Artificial Intelligence and Applications, Databases and Information Systems IV, ISBN 978-1-58603-640-8, IOS Press, Amsterdam, 2007.
6. Masolo, C., Borgo, S., Gangemi, A., Guarino, N., Oltramari, A., Ontology Library, WonderWeb Deliverable D18, 2003.
7. Guizzardi, G., Ontological Foundations for Structural Conceptual Models, Universal Press, The Netherlands, 2005. ISBN 1381-3617.
8. Lowe, E.J., The Four Category Ontology, Oxford University Press, 2006.
9. Gärdenfors, P., Conceptual Spaces: the Geometry of Thought, USA, MIT Press 2000.
10. Guizzardi, G.; Wagner, G., What´s in a Relationship: An Ontological Analysis, Proceedings of the 25th International Conference on Conceptual Modeling (ER 2008), Barcelona, Espanha, Lecture Notes in Computer Science LNCS 5231, pp. 83–97, 2008.
11. Wieringa, R.J. de Jonge, W., Spruit, P.A., Using dynamic classes and role classes to model object migration. Theory and Practice of Object Systems, 1(1), 61-83, 1995.
12. Guizzardi, G.; Masolo, C.; Borgo, S., In the Defense of a Trope-Based Ontology for Conceptual Modeling: An Example with the Foundations of Attributes, Weak Entities and Datatypes, 25th Intl. Conf. on Conceptual Modeling (ER’2006), Arizona, USA. 2006.

13. Guizzardi, G., Wagner, G., Guarino, G., van Sinderen, M., An Ontologically Well-Founded Profile for UML Conceptual Models, 16th International Conference on Advanced Information Systems Engineering (CAISE 2004), Riga.
14. Benevides, A.B. et al., Validating modal aspects of OntoUML conceptual models using automatically generated visual world structures, Journal of Universal Computer Science, Vol. 16/20, Special Issue on Evolving Theories of Conceptual Modeling, 2010.
15. Guizzardi, G., Ontological Meta-Properties of Derived Object Types, 24th International Conference on Advanced Information Systems Engineering (CAISE'12), Gdansk, 2012.
16. Halpin, T., Morgan, T., Information Modeling and Relational Dababases, Morgan Kaufman, 2008. ISBN 1558606726
17. Bauman, B. T., Prying Apart Semantics and Implementation: Generating XML Schemata directly from ontologically sound conceptual models, Balisage Markup Conference, 2009.
18. Fiorini, S., Abel, M., Scherer, C., A Symbol Grounding Model for the Interpretation of 2D-line charts, 15th IEEE International Enterprise Computing Conference (EDOC 2010) Workshop Proceedings, Vitória, Brazil, 2010.
19. Guizzardi, G., The Problem of Transitivity of Part-Whole Relations in Conceptual Modeling Revisited, 21st International Conference on Advanced Information Systems Engineering (CAISE'09), Amsterdam, The Netherlands, 2009.
20. Guizzardi, G.; Lopes, M.; Baião, F.; Falbo, R., On the importance of truly ontological representation languages, International Journal of Information Systems Modeling and Design (IJISMD), 2010. ISSN: 1947-8186.
21. Guizzardi, R. S. S., Guizzardi, G., Ontology-Based Transformation Framework from Tropos to AORML In Social Modeling for Requirements Engineering, Cooperative Information Systems Series, MIT Press, Boston, 2010.
22. Costal, D., Goméz, C., Guizzardi, G., Formal Semantics and Ontological Analysis for Understanding Subsetting, Specialization and Redefinition of Associations in UML, 30th International Conference on Conceptual Modeling (ER 2011), Brussels, Belgium, 2011.
23. Benevides, A.B.; Guizzardi, G., A Model-Based Tool for Conceptual Modeling and Domain Ontology Engineering in OntoUML, 11th International Conference on Enterprise Information Systems (ICEIS), Milan, 2009.
24. Welty, C. and Fikes, R., A Reusable Ontology for Fluents in OWL. Proceeding of the 2006 conference on Formal Ontology in Information Systems: Proceedings of the Fourth International Conference (FOIS 2006), IOS Press (2006), 226-236.
25. Zamborlini, V.; Guizzardi, G., On the representation of temporally changing information in OWL, 15th IEEE International Enterprise Computing Conference (EDOC 2010) Workshop Proceedings, Vitória, Brazil, 2010.
26. Varzi, A. C., Naming the stages, Dialectica, v. 57, n. 4, p. 387–412, 2003.
27. Sider, T., Fourdimensionalism: An Ontology of Persistence and Time, Oxford University Press, 2003
28. Quine, W.V. Events and reification. In Actions and Events: Perspectives on the Philosophy of Davidson. Blackwell, 1985, 162-71.
29. West, M., Information Modeling: An Analysis of uses and meanings of Associations, PDT Europe, 2002.