A Unified Foundational Ontology and some Applications of it in Business Modeling

Giancarlo Guizzardi[1] and Gerd Wagner[2]

[1] Centre for Telematics and Information Technology, Univ. of Twente
Enschede, The Netherlands
guizzard@cs.utwente.nl
[2] Eindhoven Univ. of Technology, Faculty of Technology Management
Eindhoven, The Netherlands
G.Wagner@tm.tue.nl,
http://tmiswww.tm.tue.nl/staff/gwagner

Abstract: Foundational ontologies provide the basic concepts upon which any domain-specific ontology is built. This paper presents a new foundational ontology, UFO, and shows how it can be used as a guideline in business modeling and for evaluating business modeling methods. UFO is derived from a synthesis of two other foundational ontologies, GFO/GOL and OntoClean/DOLCE. While their main areas of application are natural sciences and linguistics/cognitive engineering, respectively, the main purpose of UFO is to provide a foundation for conceptual modeling, including business modeling.

1 Introduction

A *foundational ontology*, sometimes also called 'upper level ontology', defines a range of top-level domain-independent ontological categories, which form a general foundation for more elaborated domain-specific ontologies. A well-known example of a foundational ontology is the *Bunge-Wand-Weber (BWW)* ontology proposed by Wand and Weber in a series of articles (e.g. Wand & Weber, 1990; 1995) on the basis of the original metaphysical theory developed by Bunge (1977; 1979).

As has been shown in a large number of recent works (e.g., Green & Rosemann, 2000; Evermann & Wand, 2001; Guizzardi, Herre & Wagner, 2002a-b; Opdahl & Henderson-Sellers, 2002) foundational ontologies can be used to evaluate conceptual modeling languages and to develop guidelines for their use. Business modeling can be viewed as a main application domain of conceptual modeling languages and methods. In the *Model-Driven Architecture* approach of the OMG, a business model is called a "computation-independent model" because it must not be expressed in terms of IT concepts, but solely in terms of business language. The business domain, since it contains so many different kinds of things, poses many challenges to foundational ontologies.

A unified foundational ontology represents a synthesis of a selection of foundational ontologies. Our main goal in making such a synthesis is to obtain a foundational ontology that is tailored towards applications in conceptual modeling. For this purpose we have to capture the ontological categories underlying natural language and human cognition, which are also reflected in conceptual modelling languages such as ER diagrams or UML class diagrams. In (Gangemi et al, 2002), this approach is called 'descriptive ontology' as opposed to 'prescriptive ontology', which claims to be 'realistic' and robust against the state of the art in scientific knowledge.

For UFO 0.1, the first experimental version of our Unified Foundational Ontology (UFO), we combine the following two ontologies: (1) the General Formal Ontology (GFO), which is underlying the General Ontological Language (GOL) developed by the OntoMed research group at the University of Leipzig, Germany; see www.ontomed.de and (Degen, Heller, Herre & Smith, 2001); (2) the OntoClean ontology (Welty and Guarino, 2001) and the Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE) developed by the ISTC-CNR-LOA research group in Italy, as part of WonderWeb Project; see http://wonderweb.semanticweb.org/.

Our choice is based on personal familiarity and preferences and not on an evaluation of all alternatives. Nonetheless, in previous attempts, GFO has been proven insightful in providing a principled foundation for analyzing and extending conceptual modeling and ontology representation languages and constructs (Guizzardi, Herre & Wagner, 2002a-b).

We have obtained our synthesis by: (i) selecting categories from the union of both category sets; (ii) renaming certain terms in order to create a more 'natural' language; (iii) and adding some additional categories based on relevance for conceptual modeling according to our experience. We also make references to BWW, the Web ontology language OWL, the Unified Modeling Language (UML), the terminology standard ISO1087-1:2000 (ISO, 2000), and to the *Business Rules Team submission* to the OMG Business Semantics for Business Rules RFP (Chapin et al, 2004). For making a distinction between terms used differently in different vocabularies, we use the XML namespace prefix syntax and write, e.g., "BWW:thing" and "owl:Thing".

We present UFO 0.1 both as a MOF/UML model and as a vocabulary in structured English, similar to the *BSBR Structured English* of (Chapin et al, 2004). The vocabulary consists of three kinds of entries marked up with different font styles:

- term : a term in this font style denotes being of a type and is used to refer to things of that type
- *name* : a name of an individual or a type; when abc is a type term referring to things of that type, *abc* is a name referring to the type itself
- term1 relationship predicate term2 : an expression that denotes being of a relationship type and that is used to refer to relationships of that type

A vocabulary entry may contain, additionally,

- 'Corresponding terms' (or 'corresponding relationship type expressions'): terms (or relationship type expressions) that are roughly equivalent
- Examples
- Constraints: logical statements that have to hold in any given ontology based on UFO 0.1

When there is a primary source for a definition, we append it in brackets, like [based on GFO].

UFO is divided into three incrementally layered *compliance sets*: (1) UFO-A defines the core of UFO, excluding terms related to perdurants and terms related to the spheres of intentional and social things; (2) UFO-B defines, as an increment to UFO-A, terms related to perdurants; (3) UFO-C defines, as an increment to UFO-B, terms related to the spheres of intentional and social things, including linguistic things.

This division reflects a certain stratification of our "world". It also reflects different degrees of scientific consensus: there is more consensus about the ontology of endurants than about the ontology of perdurants, and there is more consensus about the ontology of perdurants than about the ontology of intentional and social things.

We hope that this division into different compliance sets will facilitate both the further evolution of UFO and the adoption of UFO in business modeling and ontology engineering. In section 2 we present UFO-A 0.1, while UFO-B 0.1 and UFO-C 0.1 are presented in the sections 3 and 4, respectively. Section 5 illustrates how UFO can be used to evaluate some business modeling methods. Section 6 concludes the article.

2 UFO-A 0.1 – the core of A Unified foundational Ontology

2.1 Things, Sets, Entities, Individuals and Types

We first present the upper part of UFO 0.1 as a MOF/UML model in Figure 1. Notice the fundamental distinction made between *sets* and *entities* as things that are not sets (called 'urelements' in GFO).

In structured English, the upper part of UFO 0.1 can be introduced as follows.

thing : anything perceivable or conceivable [ISO:object]. *Corresponding terms*: GFO:entity; DOLCE:entity, owl:Thing; BSBR:thing

set : thing that has other things as members (in the sense of set theory)

thing is member of set : designated relationship that is irreflexive, asymmetric and intransitive

member : role name that refers to the first argument of the *thing is member of set* relationship type

set is subset of set : designated relationship that is reflexive, asymmetric and transitive. *Constraint*: For all t:thing; s_1, s_2 : set – if t is member of s_1 and s_1 is subset of s_2, then t is member of s_2

entity : thing that is not a set; neither the set-theoretic membership relation nor the subset relation can unfold the internal structure of an entity [GFO:urelement]

type : entity that has an extension (being a set of entitys that are instances of it) and an intension, which includes an applicability criterion for determining if an entity is an instance of it; and which is captured by means of an axiomatic specification, i.e., a set of axioms that may involve a number of other types representing its essential features. A type is a space-time independent pattern of features, which can be realized in a number of different individuals. [based on GFO:universal]. *Corresponding terms*: UML:class; DOLCE:universal; owl:Class; BSBR:"generic thing"

entity is instance of type : designated relationship (called *classification*)

instance : role name that refers to the first argument of the *entity is instance of type* relationship type

set is extension of type : designated relationship. *Constraint:* For all *o*:entity, *t*:type, *s*:set – if *o* is instance of *t* and *s* is extension of *t*, then *o* is member of *s*.

extension : role name that refers to the first argument of the *set is extension of type* relationship type

type is subtype of type : designated relationship that is irreflexive, asymmetric and transitive (also called *generalization*). *Constraint:* For all t_1, t_2 : type; s_1, s_2 : set – if t_1 is subtype of t_2 and s_1 is extension of t_1 and s_2 is extension of t_2, then s_1 is subset of s_2.

subtype : role name that refers to the first argument of the *type is subtype of type* relationship type

individual : entity that is not a type. The relation between individual and type is one of classification. *Corresponding terms*: GFO:individual; DOLCE:particular

thing is part of individual : designated relationship that is reflexive, asymmetric and transitive (also called *aggregation* relationship).

part : role name that refers to the first argument of the *thing is part of individual* relationship type

type is categorization type of type : designated relationship where the first argument/role is a higher-order type whose instances form a subtype partition of the second argument (also called *categorization* relationship). *Examples*: BiologicalSpecies is categorization type of Animal; PassengerAircraftType is categorization type of PassengerAircraft. *Constraint*: For all t_1, t_2, t_3: type – if t_3 is categorization type of t_1 and t_2 is instance of t_3, then t_2 is subtype of t_1

categorization type : role name that refers to the first argument of the *type is categorization type of type* relationship type. *Corresponding names*: GFO:higher-order universal; BSBR:"categorization scheme"; UML:powertype

type is categorized by type : designated relationship that is the inverse of type is categorization type of type. *Corresponding relationship type expressions*: BSBR:"type has categorization-scheme"

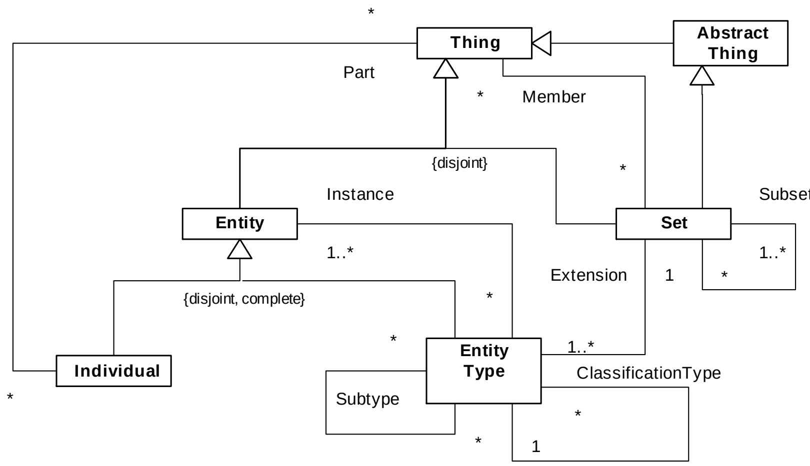

Figure 1: The upper part of UFO 0.1 as a MOF/UML model.

2.2 Different Kinds of Types

Based on (Wiggins, 2001; van Leeuwen, 1991; Gupta, 1980; Hirsch, 1982), we distinguish between several different kinds of types, as shown in Figure 2. These distinctions are elaborated in (Guizzardi et al, 2004a), in which we present a philosophically and psychologically well-founded theory of types for conceptual modeling. In (Guizzardi et al, 2004b), this theory is used to propose: (i) a profile for UML whose elements represent finer-grained distinctions between different kinds of types; (ii) a set of constraints defining the admissible relations between these elements. One should refer to (Guizzardi et al, 2004a-b) for: (a) an in depth discussion on the theory underlying these categories as well as the constraints on their relations; (b) a formal characterization of the profile; (c) the application of the profile to propose a ontological design pattern that addresses a recurrent problem in the practice of conceptual modeling.

In structured English, the different kinds of types are defined as follows.

relationship type : type whose instances are relationships

sortal type : type that carries a criterion for determining the individuation, persistence and identity of its instances. An identity criterion supports the judgment whether two instances are the same. Every instance in a conceptual model must have an identity and, hence, must be an instance of **sortal type**.

base type : sortal type that is rigid (all its instances are necessarily its instances) and that supplies an identity criterion for its instances [OntoClean:type]. *Examples*: Mountain; Person. *Corresponding terms*: BWW:"natural kind"

phase type : sortal type that is anti-rigid (its instances could also not be instances of it without loosing their identity) and that is an element of a subtype partition of a base type [OntoClean:"phased sortal"]. *Examples*: Town and Metropolis are phase subtypes of City; Baby, Teenager and Adult are phase subtypes of Person

role type : sortal type that is anti-rigid and for which there is a relationship type such that it is the subtype of a base type formed by all instances participating in the relationship type [OntoClean:role]. *Examples*: DestinationCity as role subtype of City; Student as role subtype of Person

Role and phase types cannot supply an identity criterion for its instances. For this reason, roles and phases must be subsumed by a base type from which an identity criterion is inherited.

mixin type : type that is not a sortal type and can be partitioned into disjoint subtypes which are sortal types (typically role types) with different identity criteria. Since a mixin is a non-sortal it cannot have direct instances [OntoClean:non-sortal]. *Examples*: Object; Part; Customer; Product

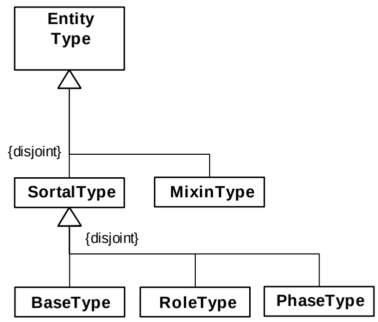

Figure 2: Different kinds of types.

The theory of types which is part of UFO-A provides a foundation for a number of modeling primitives that, albeit often used, are commonly defined in an ad hoc manner in the practice of conceptual modeling (e.g. kind, phase or state, role, mixin). In particular, this theory can be considered as an elaboration in the way types are accounted for in the BWW approach. In one of the BWW papers (Evermann & Wand, 2001), it is proposed that a UML class should be used to represent a BWW-natural kind (it should be equivalent to functional schema of a BWW-natural kind). As discussed in (Guizzardi et al, 2004a), a natural kind is in the same ontological footing as what is named here a Base type, i.e. it is a rigid type that provides an identity criterion for its instances. It is been demonstrated in several works in the literature (Welty & Guarino, 2001; Gupta, 1980; Wiggins, 2001; van Leeuwen, 1991; Guizzardi et al, 2004a-b) that this kind of type construct constitutes only one of the sorts which are necessary to represent the phenomena available in cognition and language. In other words, a conceptual modeling construct representing a base type is only one of a set of modeling constructs which should be available to the conceptual modeler.

2.3 Different Kinds of Individuals

We distinguish between a number of different kinds of individuals, as shown in Figure 3.

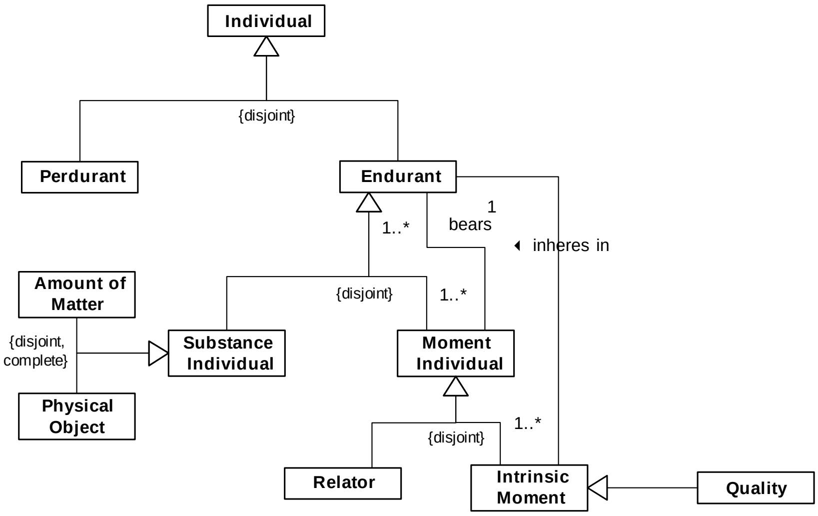

Figure 3: Different kinds of individuals.

In structured English, these different kinds of individuals are explained as follows.

endurant : individual that is wholly present whenever it is present, i.e. it does not have temporal parts. An endurant is something which persists in time while keeping its identity. Examples are a house, a person, the moon, a hole, the redness of an apple and an amount of sand. [DOLCE]. *Corresponding terms*: GFO:3D-individual

perdurant : individual that is composed of temporal parts; whenever a perdurant is present, it is not the case that all its temporal parts are present. The distinction between endurants and perdurants can be understood in terms of the intuitive distinction between "objects"(things, entities) and "processes"(events) in ordinary parlance. Examples of perdurants are a race, a conversation, the Second World War and a business process [DOLCE]

substance individual : endurant that consists of matter (i.e., is `tangible' or concrete), possesses spatio-temporal properties and can exist by itself; that is, it does not existentially depend on other endurants, except possibly on some of its parts) [based on GFO:substance]. *Corresponding terms*: BWW:thing

moment individual : endurant that cannot exist by itself; that is, it depends on other endurants, which are not among its parts [based on GFO:moment]

endurant bears moment individual : designated relationship [based on GFO:"substance bears moment"]

physical object : substance individual that satisfies a condition of unity and for which certain parts can change without affecting its identity

amount of matter : substance individual that does not satisfy a condition of unity; typically referred to by means of mass nouns. Amounts of matter are, in general, mereologically invariant, i.e., they cannot change any of their parts without changing their identity [DOLCE]. *Examples*: water; gold; wood; milk; sand

intrinsic moment : moment individual that is existentially dependent on one single individual

intrinsic moment inheres in endurant : designated relationship [GFO]

quality : intrinsic moment that inheres in exactly one endurant and can be mapped to a value (quale) in a quality dimension (Gärdenfors, 2000). *Corresponding terms*: GFO:quality; DOLCE:quality; BWW:"intrinsic property". *Examples*: the color (height, weight) of a physical object; an electric charge. *Constraint*: For all e_1, e_2 : endurant; q:quality — if q inheres in e_1 and q inheres in e_2, then e_1 is equal to e_2.

relational moment: moment individual that is existentially dependent on more than one individual. Relational moments provide a foundation for the construction of material relationships between individuals (Guizzardi, Herre & Wagner, 2002b). The category of relational moments in UFO is based on the concept of a [GFO:Relator]. The notion of relators is supported in several works in the philosophical literature (Smith & Mulligan, 1986; Smith & Mulligan, 1986) and, the position advocated here is that, it plays an important role in: (i) distinguishing material relations such as *'being married to'* and *'studies at'* from their formal counterparts (e.g. 5 *is greater than* 3, this day is *part-of* this month); (ii) answering questions of the sort: what does it mean say that John is married to Mary? Why is it true to say that Bill works for Company X but not for Company Y? *Corresponding terms*: BWW:"mutual property". *Examples*: a particular employment (Susan is employed by IBM); a particular flight connection (LH403 flies from Berlin to Munich); a kiss; a handshake.

2.4 Some Applications of UFO-A 0.1 to Business Modeling Problems

2.4.1 Modeling Customers

Most business information systems include a 'business object class' `Customer` for representing the customers of the business. In figure 4-a, the role type *Customer* is defined as a supertype of *Person* and *Corporation*. This model is deemed ontologically incorrect for two reasons: first, not all persons are customers, i.e. it is not the case that the extension of *Person* is necessarily included in the extension of *Customer*. Moreover, an instance of *Person* is not necessarily (in the modal sense) a

Customer. Both arguments are also valid for *Organization*. In a series of papers (e.g., Steimann, 2000), Steimann discusses the difficulties in specifying supertypes for Roles that can be filled by instances of disjoint types[1]. As a conclusion, he claims that the solution to this problem lies in separating the hierarchies of role type and base type (named natural type in the article) - a solution which strongly impacts the metamodel of all major conceptual modeling language. By using the theory of types underlying UFO-A we can show that this claim is not warranted and we are able to propose a *design pattern* that can be used as an ontologically correct solution to this recurrent problem (Guizzardi et al, 2004b).

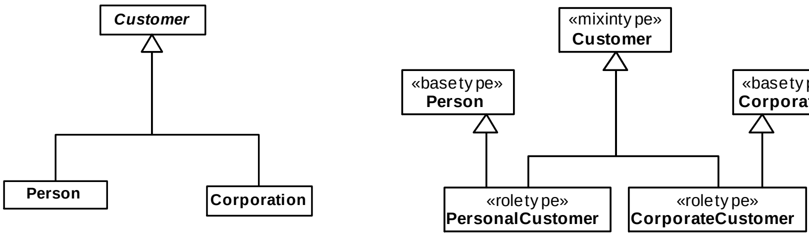

Figure 4-a (left): An ontologically incorrect model of roles; **Figure 5-b:** An ontologically correct version of (Fig.4-a) according to UFO 0.1.

In this example, Customer has in its extension individuals that that obey different identity criteria, i.e., it is not the case that there is a single identity criteria which applies both for Persons and Corporations. Customer is hence a mixin type (a non-sortal) and, by definition, cannot supply an identity criterion for its instances. Since every instance in the model must have an identity, thus, every instance of Customer must be an instance of one of its subtypes (forming a partition) that carries an identity criterion. For example, we can define the sortals PrivateCustomer and CorporateCustomer as subtypes of Customer (fig.4-b). These sortals, in turn, carry the (incompatible) identity criteria supplied by the base types Person and Corporation, respectively.

2.4.2 Product Modeling

In many business information systems, both individual products and product types have to be represented. In a prototypical case, the product individual type, whose instances are identified with the help of serial numbers, is categorized by the corresponding product model type, which is a 2nd order categorization type, whose instances are subtypes of the product individual type. Figure 6 shows this situation for

---

[1] This problem is also mentioned in (van Belle, 1999): *"how would one model the customer entity conceptually? The Customer as a supertype of Organisation and Person? The Customer as a subtype of Organisation and Person? The Customer as a relationship between or Organisation and (Organization or Person)?."*

the case of cars and car models. In a proposal for ontological foundations of the REA model (Geert & McCarthy, 2000), the authors argue about the importance of the type/categorization type distinction accounted here: "*Economic Resources like (especially) inventory have an instance/type definition problem that must be solved in the REA ontology (or in any information system)… cars in an automobile dealership would be modeled with instances (a car with a given engine#) …with classes of cars (1975 Corvette) as type-images.*"

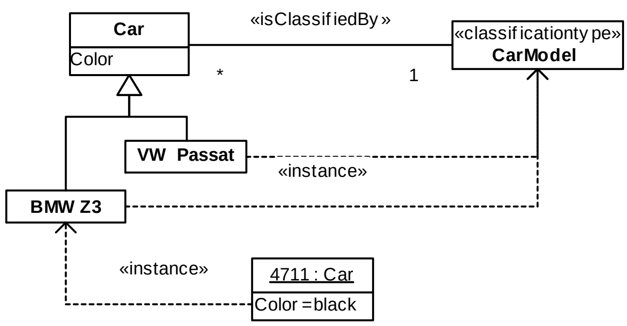

Figure 6: UML Product modeling with UFO-based stereotypes.

3 UFO-B 0.1 – Perdurants

A complete treatment of an ontology of perdurants requires a detailed discussion on an ontology of temporal entities (chornoids) (Degen, Heller, Herre & Smith, 2001). In this section, instead, we focus our attention to some basic categories of UFO-B 0.1 that will be used in section 4 in order to characterize some intentional entities and in section 5 to review some enterprise modelling approaches. In the sequel we (informally) discuss the following basic kinds of perdurants shown in Figure 6: (atomic and complex) events and states.

state : perdurant that is homeomeric (each of its temporal parts belongs to the same state type as the whole) [based on DOLCE]

event : perdurant that is related to exactly two states (its pre-state and its post-state). An event is related to the states before and after it has happened.

atomic event : event that happens instantaneously, i.e. an event without duration. [based on BWW:event and GFO:change]

complex event : event that is composed of two or more events. Examples: a football game, a conversation, a race, a birthday party, a business process.

state is pre-state of event : designated relationship

state is post-state of event : designated relationship

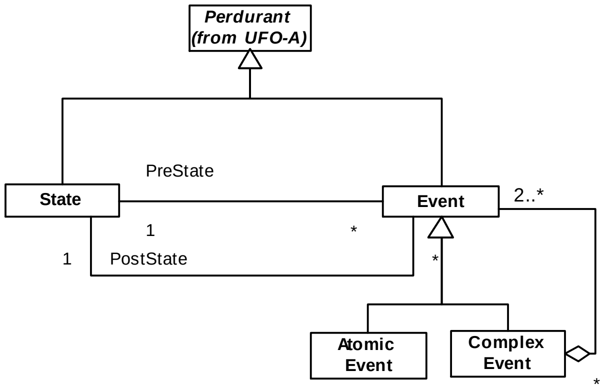

Figure 7: The perdurant categories of UFO-B 0.1

4 UFO-C 0.1 – Intentional, Social and Linguistic things

The ‘objective’ perdurant categories *(atomic and complex) event* and *state* defined in UFO-B are essential concepts for process modeling, but they are not sufficient for *business process* modeling, where intentional and social concepts such as *action*, *activity*, and *communication* are needed. The following account of intentional and social things is at an early stage of development and therefore rather incomplete. Nevertheless, we think that it gives an impression of the range of ontological categories that is needed to explain business process modeling.

physical agent : physical object that creates action events affecting other physical objects, that perceives events, possibly created by other physical agents, and to which we can ascribe a mental state. *Examples*: a dog; a human; a robot

action event : event that is created through the action of a physical agent

non-action event : event that is not created through an action of a physical agent

physical agent creates action event: designated relationship

physical agent perceives event: designated relationship

non-agentive object : physical object that is not a physical agent. *Examples*: a chair; a mountain

mental moment : intrinsic moment that is existentially dependent on a particular agent, being an inseparable part of its mental state. *Examples*: a thought; a perception; a belief; a desire; an individual goal. *Constraint*: For all *mm* : mental moment; *e*:endurant — if *mm* inheres in *e* then *e* is physical agent

communicating physical agent : physical agent that communicates with other communicating physical agents. *Examples*: a dog; a human; a communication-enabled robot

institutional agent : institutional fact (Searle, 1995) that is an aggregate consisting of communicating agents (its *internal agents*), which share a collective mental state, and that acts, perceives and communicates through them. *Examples*: a business unit; a voluntary association

agent : endurant that is either a physical agent or an institutional agent

communicating agent : agent that communicates with other communicating agents

social moment : relational individual that is existentially dependent on more than one communicating agent. *Examples*: a commitment; a joint intention

5 Using UFO to Evaluate Business Modeling Methods

In the following subsections we briefly present some preliminary results in order to exemplify how UFO can be used to evaluate some business modeling methods.

5.1 Enterprise Ontology

The *Enterprise Ontology*, which was developed in a project led by the AI Applications Institute at the University of Edinburgh (see Uschold, King, Moralee & Zorgios, 1998). Based on a simple upper-level ontology ('meta-ontology') consisting of the three modeling concepts *entity*, *relationship* and *actor*, it provides definitions for nearly 100 terms, both in natural language and in the formalism of *Ontolingua*.

For simplicity, the distinction between an *entity* (individual) and an *entity* type is avoided. An agent (called *actor)* is defined as a special entitiy that can play an actor role in certain relationships (such as in *performs Activity*, *has Capability*, etc.).

There is no independent concept of an event in the Enterprise Ontology: events are defined as 'a kind of activity'. Remarkably, the authors consider also events that take place as a result of natural necessity (such as 'water flowing down a hill') as activities of 'inanimate actors' (such as gravity).

The following points highlight some shortcomings of the Enterprise Ontology: (i) For conceptual modeling, it is essential to distinguish between individuals and types; (ii) It seems to be questionable to view natural forces that cause certain events to happen, such as gravity, as actors/agents; in UFO 0.1 agents have a mental state and are able to act (create action events), perceive and possibly to communicate; (iii) Events should not be subsumed under activities. Rather, they should be first-class citizens of the metamodel.

5.2 The Eriksson-Penker Business Extensions

In (Eriksson and Penker, 1999), it is proposed an approach to business modeling with UML based on four primary concepts: *resources*, *processes*, *goals*, and *rules*. In this proposal, there is no specific treatment of agents. They are subsumed, together with 'material','products' (substantial individuals), and 'information' (non-physical endurant) under the concept of *resources*. This unfortunate subsumption of human agents under the traditional `resource' metaphor prevents a proper treatment of many agent-related concepts such as commitments, authorization, and communication/interaction.

5.3 The REA (Resource-Event-Agent) Model

The REA framework, whose ontological foundations are defined in (Geert & McCarthy, 2000), is based on a fundamental notion of an economic exchange. An economic exchange comprises a pair of economic events: an inflow and an outflow event. Economic agents participate in economic events and resources are affected (e.g. produced, used, acquired) by these events. In UFO, an economic event is a type of complex action and resource is a type of substantial individual (resources can be both physical objects and amounts of matter). The notion of an economic agent cannot be directly related to the notion of agent defined in UFO-C. In UFO, agent is a rigid concept that denotes an entity to which we can ascribe a mental state, independently whether the entity participates in some action. In REA, conversely, an entity is an (economic) agent by virtue of its participation in an economic event. Hence, in REA, agent is an anti-rigid concept akin the notion of role individual discussed here.

Despite of considering both individual and types, the authors do not elaborate on the different sorts of types which are necessary for conceptual enterprise modeling (see section 2.4.1).

An example of lack of ontological clarity is found when the authors mix the notions of event and commitments. For instance, figure 5, commitment and economic event are collapsed in one single type-image. Additionally, the relationships *partner* and *reserves* (defined to hold between agent/commitment and resource/commitment, respectively) are considered as subtypes of *participation* and *stock-flow* (defined between agent/economic event and resource/economic event). In our framework, whilst an economic event is a type of complex action, a commitment is a type of relational moment. Examples of other types of social moments (a subtype of relational moment) defined in REA include *accountability*, *responsibility*, *assignment*, and *custody*.

Despite recognizing the importance of part-whole relations in the enterprise domain (for example to model the relation between a resource and its parts), the treatment offered is insufficient. The authors only briefly mention a relation of composition that, together with other relations such as *substitutes* (meaning that a resource can substitute another), is subsumed under the relation of *linkage* between resources. No axiomatization for composition is provided. In a companion paper (Guizzardi, Herre & Wagner, 2002b), we provide a formal characterization for parthood and discuss different types of this relation which are important for conceptual modeling.

6 Conclusions

The unified foundational ontology UFO 0.1 presented in this paper should be viewed as an attempt to assemble a foundational ontology for conceptual modeling on the basis of other, already well-established and philosophically justified foundational ontologies. We have stratified UFO into three ontological layers in order to distinguish its core, UFO-A, from the perdurant extension layer UFO-B and from the agent extension layer UFO-C. Although there is not much consensus yet in the literature regarding the ontology of agents, such an ontology is needed for building the foundation of conceptual business process modeling. UFO-C 0.1 is a first attempt to construct these foundations. We hope that we can validate and further improve it by investigating its applicability to business modeling problems.

References

1. Bunge, M. (1977). Treatise on Basic Philosophy. Vol. 3. Ontology I. The Furniture of the World. D. Reidel Publishing, New York.
2. Bunge, M. (1979). Treatise on Basic Philosophy. Vol. 4. Ontology II. A World of Systems. D. Reidel Publishing, New York.
3. Chapin et al (2004). Business Semantics of Business Rules (BSBR). Initial Submission to OMG BEI RFP br/2003-06-03, 12 January 2004. Available from http://www.omg.org/cgi-bin/doc?bei/04-01-04.
4. Degen, W., Heller, B., Herre, H. & Smith, B. (2001). GOL: Towards an axiomatized upper level ontology. In Smith, B. & Guarino, N. (eds.), Proceedings of FOIS’01, Ogunquit, Maine, USA, October 2001. ACM Press.
5. Eriksson, H.E. and Penker, M. (1999): Business Modeling with UML: Business Patterns at Work, John Wiley & Sons.
6. Evermann, J. & Wand Y. (2001). Towards ontologically based semantics for UML constructs. In Kunii, H.S., Jajodia, S. & Solvberg, A. (eds.), Proceedings of ER 2001, pages 354–367, Springer-Verlag.
7. Gangemi, A., Guarino N., Masolo C., Oltramari, A. & Schneider L. (2002). Sweetening Ontologies with DOLCE. Proceedings of EKAW 2002, Siguenza, Spain.
8. Gärdenfors, P. Conceptual Spaces (2000): the Geometry of Thought. MIT Press, USA.
9. Geert, G. & McCarthy, W.E. (2000): The Ontological Foundation of REA Enterprise Information Systems, online: http://www.msu.edu/user/mccarth4/rea-ontology/
10. Green, P.F. & Rosemann, M. (2000). Integrated Process Modelling: An Ontological Evaluation. Information Systems 25 (2), 73-87.
11. Green, P.F. & Rosemann, M. (2002). Usefulness of the BWW Ontological Models as a "Core" Theory of Information Systems. In Proceedings Information Systems Foundations: Building the Theoretical Base, Canberra, 2002, 147-164.

12. Guizzardi, G., Herre, H. & Wagner G. (2002a). On the General Ontological Foundations of Conceptual Modeling. Springer-Verlag, Berlin, Lecture Notes in Computer Science Vol. 2503.
13. Guizzardi, G., Herre, H., Wagner G. (2002b): Towards Ontological Foundations for UML Conceptual Models. Springer-Verlag, Berlin, Lecture Notes in Computer Science Vol. 2519.
14. Guizzardi, G., Wagner G., van Sinderen, M. (2004a): A Formal Theory of Conceptual Modeling Universals, In Proceedings of the International Workshop on Philosophy and Informatics (WSPI), Cologne, Germany.
15. Guizzardi, G., Wagner G., Guarino, N.; van Sinderen, M. (2004b): An Ontologically well-Founded Profile for UML Conceptual Models, Springer-Verlag, Berlin, Lecture Notes in Artificial Intelligence Vol. 3084.
16. Gupta, A. (1980). The Logic of Common Nouns: an investigation in quantified modal logic, Yale University Press, New Haven.
17. Hirsch, E. (1982). The Concept of Identity. Oxford University Press, New York, Oxford.
18. ISO (2000). ISO 1087-1 Terminology work - Vocabulary - Part 1: Theory and application. Copies of all ISO standards can be purchased from ANSI, 25 West 43rd Street, New York, NY 10036, (212) 642-4980, info@ansi.org, or http:webstore.ansi.org".
19. Leeuwen, J. van (1991). Individuals and sortal concepts: an essay in logical descriptive metaphysics, PhD Thesis, University of Amsterdam.
20. Opdahl, A.L. & Henderson-Sellers, B. (2002). Ontological evaluation of the UML using the Bunge-Wand-Weber Model. *Software and Systems Journal* 1 (1), 43-67.
21. Searle, J.R. (1995). *The Construction of Social Reality*. Free Press, New York.
22. Smith, B.; Mulligan, K. (1986): A Relational Theory of the Act, Topoi (5/2), 115-30.
23. Smith, B.; Mulligan, K. (1983): Framework for Formal Ontology, Topoi (3), 73-85.
24. Steimann, F. (2000): On the representation of roles in object-oriented and conceptual modeling, Data & Knowledge Engineering 35:1, 83–106.
25. van Belle, J.P. (1999). Moving Towards Generic Enterprise Information Models: From Pacioli to CyC. Proceedings of the 10th Australasian Conference on Information Systems, Wellington.
26. Wand, Y. & Weber, R. (1990). Mario Bunge's Ontology as a formal foundation for information systems concepts. In Weingartner, P. & Dorn, G.J.W. (eds.), *Studies on Mario Bunge's Treatise*, Rodopi, Atlanta.
27. Wand, Y. & Weber, R. (1995). On the deep structure of information systems. Information Systems Journal 5, 203-223.
28. Welty, C. & Guarino, N (2001). Supporting Ontological Analysis of Taxonomic Relationships. *Data and Knowledge Engineering*, 39(1), 51-74.
29. Wiggins, D. (2001). Sameness and Substance Renewed, Cambridge University Press.