Taking it to the Next Level: Nicola Guarino, Formal Ontology and Conceptual Modeling

Giancarlo GUIZZARDI [a,b,1], John MYLOPOULOS [c]
[a] *Conceptual and Cognitive Modeling Research Group (CORE),*
Faculty of Computer Science, Free University of Bozen-Bolzano, Italy
[b] *Ontology and Conceptual Modeling Research Group (NEMO),*
Computer Science Department, Federal University of Espírito Santo (UFES), Brazil
[c] *University of Trento, Italy*

1. Introduction

This paper is dedicated to Nicola Guarino, on the occasion of his 65th birthday. Nicola has made seminal contributions to Conceptual Modeling that include some of the greatest advances in this field of research over the past thirty years.

Nicola's contributions include OntoClean [26, 27], proposed jointly with Chris Welty, the first proposal of formal ontological analysis. This work has been widely cited, but is also used in academic and industrial settings around the world, thus having had tremendous impact. One of the OntoClean papers has had more than 1,000 citations (Google Scholar, October 2018) and won an "Thomson-ISI recognition of an "Emerging Research Front" award in 2004. Another seminal contribution of Guarino's research is his work on the DOLCE foundational ontology, which has also had broad and deep impact in the field [59, 60, 42, 51, 43]. But by far his most significant contribution lies in his critique of conceptual modelling and knowledge representation languages for being ontologically neutral. Instead, he has argued convincingly that such languages should make commitments for the primitive concepts they offer on their ontological properties concerning existence, dependence, identity and rigidity. Such commitments reduce the space of possible interpretations for conceptual models and align them more closely to modeller intentions. This view that Conceptual Modeling Languages should break with ontological neutrality by committing to a suitable ontological theory strongly influenced the design of a next-generation of conceptual modeling approaches such as, for example, OntoUML [29, 38, 31].

The main objective of this paper is to present some of Nicola's contributions and highlight their importance to Conceptual Modeling. To do so, we begin with an account of what is Conceptual Modelling (section 2), followed by the essential elements of the ontological framework proposed by Nicola (section 3). In section 4, we present the no-

---

[1] Corresponding Author: gguizzardi@unibz.it

tion of Ontology-Driven Conceptual Modeling and, in particular, how it has been implemented in the OntoUML program, with important direct contributions from Nicola. Section 5 presents some final considerations.

2. Conceptual Modeling

It is a foundational tenet of Cognitive Science and Philosophy of Mind that cognitive processes create, use and transform mental representations of the world. Such representations are “intentional" in the sense that they refer to, or are about something. Mental representations may be conceptual in the sense that they consist of concepts, such as thoughts, or non-conceptual, such as sensations. We call these conceptual mental representations *conceptualizations*.

The field of research we call Conceptual Modeling aims to develop concepts, tools and techniques for building computational models of conceptualizations, to be used for purposes of understanding and communication. The same can be said for the related field of Knowledge Representation in Artificial Intelligence (AI). The difference between the two fields is that for Conceptual Modelling, these models are used to support the design of databases, software, business processes, enterprises etc., whereas in Knowledge Representation, these models (aka knowledge bases) are used to endow an intelligent computational agent with suitable knowledge for the performance of an intelligent task, such as planning, diagnosis, design, etc.

As far back as Aristotle there have been theories that conceptualizations consist of concepts and associations that relate similar concepts. According to empiricists (Hume et al), associations come about when concepts co-occur in the experiences of a cognitive agent. For example, the concepts of ‘Student’ and ‘Person’ co-occur every time you encounter a student, so a Student/Person association is meaningful, and likely. Most proposals for conceptual models adopt such an associationist stance, including Semantic Networks, Object-Oriented models and Description Logics.

The origins of conceptual modeling can be traced back to the 60s. Ross Quillian [52] proposed in his PhD thesis the notion of Semantic Networks, a form of directed, labelled graph, as models of human (semantic) memory. Nodes of his semantic network proposal represented concepts (more precisely, word senses.) For words with multiple meanings, such as “plant", there would be several nodes, one for each sense, e.g., “plant" as in “industrial plant", “plant" as in “evergreen plant”, “plant" as in “plant my garden every year", etc. Nodes were related through links representing semantic relationships, such as isA (“A bird is a(n) animal", “a shark is a fish”), has (“A bird has feathers"), and eat (“Sharks eat humans"). Moreover, each concept could have associated attributes, representing properties, such as “Penguins can’t fly". There are several noteworthy ideas in Quillian’s proposal. Firstly, his conceptual models consisted of concepts and associations. Moreover, generic concepts were organized into an isA (or, generalization) hierarchy, supported by attribute inheritance. In addition, his proposal came with a radical computational model where finding meanings for a noun phrase, such as ‘horse food’, was accomplished by finding paths that connect the two nodes ‘horse’ and ‘food’, for example,

$$\text{horse} \xrightarrow{eats} hay \xrightarrow{IsA} \text{food}$$

$$\text{horse} \xrightarrow{madeOf} meat \xrightarrow{IsA} \text{food}$$

There was much research on semantic network-based conceptual modeling languages in the early ’70. In AI, there were many proposals, some came with an interpreter that could draw inferences from the labels associated with relationships. Others had attached assertions or procedures with every node, capturing the semantics of the concept being represented ([5], [45]). In Databases, there were proposals for semantic data models, such as the Entity-Relationship Model[2][6] and Taxis [48]. Among them, the KL-ONE knowledge representation language, proposed in the thesis of Ron Brachman, stands out for its treatment of concepts, and the reasoning support provided by the language. In KL-ONE, concepts are represented by descriptions consisting of concepts and roles. Moreover, KL-ONE supported subsumption reasoning with descriptions, where $description_1$ was subsumed by $description_2$ if all its instances were also instances of $description_2$. For example, “a man on (a hill with a telescope)” is subsumed by “a man on a hill”. KL-ONE led to a family of languages known as Description Logics that constitute the state-of-the-art of automated reasoning support in Conceptual Modeling. There is also a WWW standard for Description Logics, known as the Web Ontology Language (OWL), which was designed as a key technology in making the Semantic Web a reality.

Roughly at the same time as Quillian, Ole-Johan Dahl proposed SIMULA 67, a programming language for simulation programs [49]. This language was defined as an extension of Algol 60. The main extension consisted of the notion of a class that had instances, each with associated code so such instances were active, instead of passive data structures. The idea behind SIMULA 67 was that when you want to simulate some part of the world, for example barber shops, you define classes for the kinds of objects you are simulating, such as barber shops, barbers, customers and haircuts. SIMULA was followed by Smalltalk, developed at Xerox PARC, which formed a foundation for object-oriented programming and object-oriented modeling starting in the early 80s. The Unified Modeling Language (aka UML) constitutes a major achievement of this line of research, as it combined several proposals into one, rather loosely defined, language for modeling software designs. UML has been extended into SysML to support modeling systems, as opposed to just software.

Douglas Ross proposed in the mid-’70s the Structured Analysis and Design Technique (SADT) as a “language for communicating ideas" [54]. The technique was used by Softech, a Boston-based software company, to specify requirements for software systems. According to SADT, the world consists of activities and data. Each activity consumes some data, represented through input arrows from left to right, produces some data, represented through output arrows from left to right, and also has some data that control the execution of the activity but are neither consume nor produce. For instance, the Buy Supplies activity of Figure 1 has input arrow ’Farm Supplies’, output arrows ‘Fertilizer’ and ‘Seeds’ and control arrows ‘Seed and Vegetable Prices’ and ‘Plan Budget’. Each activity may be defined through a diagram such as that shown in Figure 1 in terms of sub-activities. Thus ’Growing Vegetables’ is defined in terms of the sub-activities ’Buy Supplies’, ‘Cultivate’, ‘Pick Produce’ and ’Extract Seeds’. Ross’ contributions include a modeling language that can capture both static and dynamic aspects of an application. Ross is credited with convincing software engineers, researchers and

---

[2] Despite its name, the Entity Relationship Model is a modeling language

practitioners alike, that it pays to have diagrammatic descriptions of how a software system is to fit its intended operational environment. This contribution helped launch Requirements Engineering as an accepted and important early phase in software development.

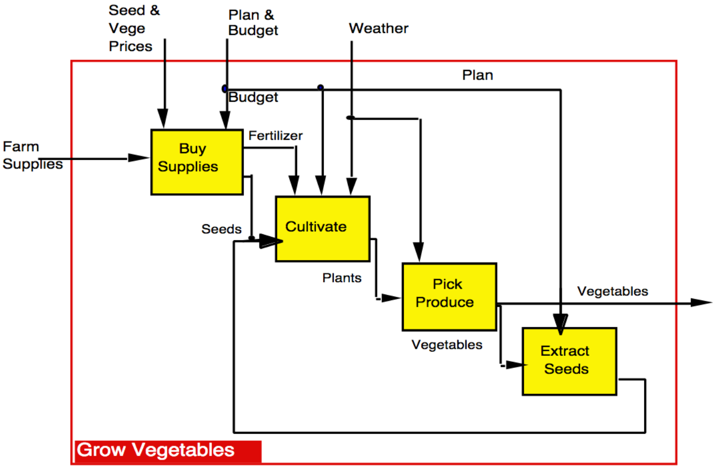

Figure 1. SADT diagram for the ‘Grow Vegetables’ activity (from [47]).

Given that the task-at-hand for Conceptual Modeling is to build models of conceptualizations, the main constituent of any proposal has been a conceptual modelling language. Such languages include the ER Model, SADT, and Simula 67. Since these early days, there have been hundreds of proposals for such languages in the literature, including popular standards used in industrial practice, such as UML, SysML and BPMN.

Any conceptual modelling language consists of (a) a set of primitive classes that represent concepts and associations; (b) a set of abstractions mechanisms, such as generalization, and aggregation, through which models expressed in the language are structured; (c) a logical language for making statements in the language; (d) a set of questions that can be answered through reasoning with respect to a model. For example, the ER Model has primitive classes Entity, Relationship and Attribute; the ER Model does not support any abstraction mechanisms, but the Extended ER Model does support generalization. Finally the language of the ER Model is a visual language that supports the definition of a collection of entity and relationship classes and their associated attributes and does not support any reasoning, so it is a rather rudimentary Logic.

In all cases, the modeling languages proposed until Nicola’s work paid scant attention to the primitive concepts they offered for modeling, treating them as mere sorts.

3. Conceptual Modeling and the Ontological Level

As previously mentioned, two of the most basic modeling primitives in domain modeling are Entity Types as well as Relationship Types. Nicola’s work made fundamental contributions in providing real-world semantics for both types of constructs and methodological support for modeling them. The key aspect here was to break with the ontological agnosticism (or neutrality) of traditional conceptual modeling and knowledge representation languages.

The importance of philosophical ontology for domain modeling (including Conceptual Modeling and Knowledge Representation) permeated Nicola's work for nearly the past 30 years. As early as in [12], he explicitly defends the importance of breaking with ontological neutrality: *"formal semantics of current knowledge representation languages usually account for a set of models which is much larger than the models we are interested in, i.e., real world models. As a consequence, the possibility to state something which is reasonable for the system but not reasonable in the real world is very high. What we need, instead, is a semantics which is not neutral with respect to some basic ontological assumptions"*.

Despite recognizing the fundamental role played by formal ontology in this context, Nicola never defended the view that the role of Ontology should be providing a single reference model that every modeler should commit to (i.e., universally accepted ontologies[3]). Instead, in his view, the role of Ontology should be to provide modelers with tools such that they could make explicit the content of their (possibly shared) assumptions, their own worldviews, or as he would formally define in [19], their *Ontological Commitment*.

The formalization of this notion of Ontological Commitment is one of the key contributions of Nicola to the conceptual and terminological clarification of the notions of ontology (as the term is employed in Computer Science), conceptualization, knowledge base, logical theory, as well as their relations [16, 21]. It also strongly contributed to the definition of ontology-based quality criteria for conceptual modeling languages [30]. Additionally, as discussed in section 4, it strongly influenced a new approach for conceptual model validation via visual simulation.

In a nutshell, according to his definition, a *conceptualization* C can be defined as a set of *intended world structures*, where each world structure is a set of individuals of the domain and a projection of existing concepts in a given world. Given a logical language L with a vocabulary V, an Ontological Commitment K is then defined as an *intensional interpretation* (in contrast with the traditional classical interpretation in model-theoretic semantics) mapping elements of V to a conceptualization C. In this case, L is said to *commit* to K, while C is the conceptualization underlying K. Now, given the commitment of L to K, the set of *intended models* includes exactly those *logical models* of L that correspond to intended structures in C. Finally, an *ontology* is a logical theory that through a suitable set of axioms over V is capable to approximate as much as possible the set of logical models of L to the set of intended ones according to K and its underlying conceptualization C. Informally speaking, when creating a specification, the modeler has a conceptualization in mind (a certain worldview). The ontological commitment is the commitment of the modeler to interpret that specification according to this worldview. However, without the support of a suitable set of formal constraints (of a suitable ontology) what will inevitably happen in practice is that the specification will allow for interpretations that are (logically) valid but non-intended. Figure 2 below (from [16]) illustrates the relations between an ontology, a language, an Ontological Commitment and the intended models of that language.

---

[3] For a discussion on the difference and relations between Ontology and ontologies, one could refer to [30].

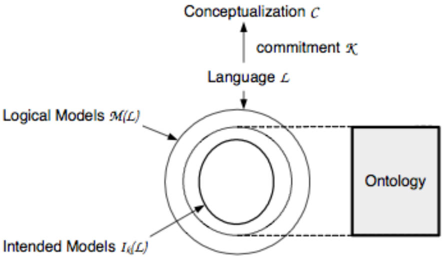

Figure 2. Relations between language (vocabulary), conceptualization, ontological commitment and ontology (from [30] after [16]).

In other words, in Nicola's view, the primary role of formal ontology for domain modeling is not to force consensus among different communities of modelers, but rather to provide these communities with theoretical and engineering tools for achieving what in [31] is termed (i) "intra-worldview consistency" and (ii) "inter world-view interoperability". A modeling approach striving for (i) should support its users in justifying their modeling choices and providing sound design rationale for choosing how the elements in the universe of discourse should be modeled in terms of language elements. Regarding (ii), it should support conceptual modelers and domain experts to be explicit regarding their ontological commitments, which in turn enables them to expose subtle distinctions between models to be integrated. Underlying this view there is the assumption that we don't have to always agree on our worldviews, the problem arises when we falsely believe we agree! This so-called *False Agreement Problem* is introduced and discussed in [16].

Following this view, in order to support goals (i) and (ii), a conceptual modeling language should offer modeling primitives which are able to capture the nuances and subtleties involving the very *essence* of the elements constituting a domain. Such a language, cannot be neutral w.r.t. to ontological choices, or to put in Nicola's terms, it should belong to *The Ontological Level* [15]. In our view, this notion of ontological level amounts to one of his most important contributions. In this seminal paper, he discusses that *Logical-Level languages* (e.g., FOL) are “flat” in the sense that they put all predicative terms (e.g., Apple and Red) in the same footing; in constrast, *Epistemological-Level* languages (e.g., KL-ONE and, hence, Semantic Network descendents, but also UML, ER, OWL) provide ways for elaborating structures which differentiate these terms. For instance, in UML, we have two alternative structuring choices: (SC1) we can define a Class of Apples with an attribute color = red; or (SC2) we can define a Class of Red with an attribute type = apple. What an Epistemological-Level language does not give us is a precise criterion for explaining why structure (a) is better than (b). As Nicola points out in that paper, structuring decisions, such as this one, should not result from heuristic considerations but instead should be motivated and explained in the terms of ontological distinctions. For instance, in this case, the choice of Apple as the sort (a) can be justified by the meta-properties that are ascribed to it. The ontological difference between the two predicates is that Apple corresponds to a Natural Kind whereas Red corresponds to an Attribution or a Mixin [29]. Whilst the former applies necessarily to its instances (an apple cannot cease to be an apple without ceasing to exist), the latter only applies contingently. Moreover, whilst the former supplies a principle of identity for its instances, i.e., a principle through which we judge if two apples are the same, the latter cannot supply one. However, it is not the case that an entity could exist without obeying a principle of identity [29], an idea which is defended both in philosophical ontology (e.g., Quine's dicto "no entity without identity" [53]), and in conceptual modeling (e.g.,Chen's design rational for ER [6]). Consequently, the structuring choice expressed in (SC2) cannot be justified. In summary, the ontological level is a level where alternative structuring (epistemological) choices over the same logical expression can be assessed and precisely justified on ontological grounds.

The idea of using formal ontology and, in particular, ontological meta-properties for motivating distinctions among types of Entity types (or, as he typically prefers to say, types of unary properties or unary relations) dates back to the beginning of the 90's. For example, as early as [13] and [14], he proposes the use of ontological meta-properties such as the Husserlian notion of foundation and ontological dependence as well as (ontological and temporal) rigidity to give an ontological semantics distinguishing modeling notions and primitives such as natural Concepts, Roles and Qualities. In addition to providing an ontological semantics for these modeling primitives, these meta-properties provide for methodological guidelines that one can use for precisely making and justifying her ontological choices: *"we argue that formal ontology may help to distinguish among the relevant kinds of relations we can define in a domain, in order to guide the correct choice of available knowledge representation primitives"* [13]. Again, the role of these methodological primitives is to make clear people's assumptions about a domain. So, he claims: *"for instance, a relation like Red may be considered as temporally rigid or not. What is important is that, as soon as the user declares a particular property for a relation, its ontological behavior becomes clear and governed by specific axioms"*. This strategy was later refined in [20, 17] and finally converged into an ontology of unary properties (universals) and a methodology based on that, which was called OntoClean [26, 67, 27]. OntoClean was perhaps the first methodology in Ontology Engineering to use a precisely defined set of ontological meta-properties and this ontology of properties to systematically analyze, rectify (i.e., "clean") and (re)design taxonomic structures.

Complementary to the ontology of universals underlying OntoClean, Nicola leads the group at the Laboratory of Applied Ontology (LOA), in Trento, Italy that proposed an ontology of particulars termed the *Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE)*. The linguistic and cognitive bias reflected in the name of this ontology represents another fundamental tenet of Nicola's view of the role of Ontology in domain modeling, namely, that an ontology suitable for supporting "intra-worldview consistency" and "inter world-view interoperability" must be one that takes language and human cognition seriously. In particular, he defends that a system of ontological categories aimed at supporting domain modeling should result from a Descriptive Metaphysics effort. As discussed in [22]: *"Descriptive metaphysics aims to lay bare the most general features of the conceptual scheme that are in fact employed in human activities, which is roughly that of common sense. The goal is to make explicit the ontological distinctions underlying natural language and human cognition. As a consequence, the categories refer to cognitive artifacts more or less depending on human perception, cultural imprints and social conventions"*. This descriptive bias in DOLCE is reflected in one of the most original constructs in the ontology, which is the notion of *Quality* adopted therein.

As a so-called "Four-Category Ontology" [36], DOLCE includes a category for property instances termed *qualities* (or abstract particulars, property instances, tropes, aspects). Qualities are "specific aspects of things we use to compare them". They *inhere* in their bearers, where inherence is a special kind of asymmetric and anti-transitive existential dependence relation. Qualities are directly comparable, while objects and events can be compared only in respect to a certain quality kind (e.g., to compare physical objects, one resorts to the comparison of their shapes, sizes, weights, and so on). Qualities are distinct from their values (a.k.a. qualia), which are abstract entities representing what exactly resembling qualities have in common, and are organized in spaces called quality spaces; each quality kind has its own quality space. For instance, weight is a quality kind, whose qualia form a linear quality space. Quality spaces may have a complex structure with multiple dimensions, each corresponding to a simple quality that inheres in a complex quality. Typical examples of complex qualities are colors, sound and taste.

The notion of qualities in DOLCE was inspired by the classical notion of tropes. However, in a classical trope-based theory, tropes are super-determinate entities that cannot change, except for being replaced by another trope (a phenomenon called trope-replacement [40]). In DOLCE, instead, qualities can maintain their identity while qualitatively changing. This move allows DOLCE to account for linguistic phenomena such as: (a) the color of the apple is changing from red to brown; (b) the temperature of the patient is rising. In (a), we have one single aspect that changes qualitatively while maintaining its identity, namely, the color. In other words, it is not "red" (which happens to be the color of the apple at t1) that is changing but a dependent aspect of the apple, namely, its color. Red and Brown are simply regions in a quality space. Analogously, in (b), it is not 38 °C that is rising (being a number, an abstract entity, 38 °C cannot rise!) but an aspect of the patient, namely, its temperature.

The original DOLCE treatment of qualities leave a number of fundamental points open. Firstly, the way a quality changes is by "pointing to" a different region in the same quality space. However, what accounts for such a change is left undefined. In other words, what should be the truthmakers of "$apple_1$ is Red at t1" and "$apple_1$ is brown at t2"? Secondly, in the original treatment, all qualities are essential qualitites, i.e., there is no room there for qualities that an entity loses or acquire in its lifetime. Thirdly, DOLCE leaves it completely undefined whether qualities are endurants (entity-like) or perdurants (event-like). Fourthly, all qualities in DOLCE are instrinsic qualities and, hence, there is no treatment there of *relational qualities*, which, besides being existentially dependent on their bearer, are also existentially dependent on something else. An example may be John's love for Mary, which inheres in John but is existentially dependent on Mary[4].

These issues have been addressed by new work on aspects (including qualities) and their connection to events. As discussed in depth in [24, 23, 33], aspects such as qualities, modes, relationships are full-fledged endurants and, as such:

1. They can qualitatively change while maintaining their numerical identity. Moreover, change happens via a mechanism that is akin to the notion of *variable and rigid embodiments* proposed by [9]. In other words, the same color quality (e.g., the color of apple-1) can be constituted by different color tropes (in the classical sense) in different points in time; the same marriage between John and Mary can be constituted by different sums of commitments and claims in different points

---

[4] Technically speaking, relational qualities inhere in one entity while being specifically dependent on another entity that is mereologically disjoint from their bearer. This is termed *external dependence* in [29].

in time; the same Dengue Fever inhering in Paul can be composed of different qualities (e.g., its severity) in different points in time;

2. They are the natural bearers of modal properties. So, while the marriage between John and Mary is necessarily a marriage, it is only contingently a marriage with full separation of assets; while Paul's Dengue Fever is necessarily a disease it is only contingently a hemorrhagic fever; while the color of apple-1 is necessarily a color, it is only contingently a red color;
3. As the natural bearers of modal properties, endurants of all these types could have been different from what they are, i.e., there are cross-world identities defined for entities of these categories. For example, we could (counter-factually) pose a different world in which Paul's Dengue fever remained asymptomatic or a a different world in which John and Mary's marriage lasted much longer and was never subject to the change from full separation of assets to partial separation of assets;
4. As subjects of change, there are changes that endurants of all these categories can undergo while remaining the same (i.e., while maintaining their identity). This criterion, as for all endurants, is given by the unique ultimate sortal type that a particular individual instantiates. For example, a DOG can change a number of features (its size, weight, age, fur color, etc.) without any impact on its identity provided that it does not change any of the essential properties (e.g., psychological continuity, maintenance of the body's autopoiesis) prescribed by the kind DOG. In an analogous manner, the marriage between John and Mary can change in a number of manners (e.g., change its marital regime, it can become recognized in a different jurisdiction) but there are ways in which it cannot change without ceasing to exist (as the same marriage) and these ways are defined a priori by the kind Marriage.

This new theory allows for providing a uniform treatment of endurants dealing with qualitative change in a way that endurants in general (hence, also qualities and relationships) can acquire and lose their own qualities, and which includes both intrinsic and relational qualities (relationships). It also accounts for the ontological status of qualities (as endurants). Moreover, it precisely defines the connection between aspects and events. On one hand, events are manifestations of aspects (qualities, modes, dispositions). For example, the movement of a needle towards a magnet is the manifestation of a number of aspects, such as the disposition of the magnet to attract metallic material, the weight of the needle, their distance, the friction of the surface, etc. Also, in the sense, the marriage between John and Mary qua-process is the sum of manifestations of (qualities of) their marriage qua-endurant, i.e., their mutual commitments, claims, feelings, etc. On the other hand, these endurants give a *focus* to an event, enriching it with criteria for individuation and unity. For example, how do we establish which events are part of the John and Mary's marriage qua-process? The answer is: it is those events that are manifestations of the (qualities of the) marriage qua-endurant. Finally, this investigation on the natural of relationships enabled the proposal of a fuller theory of relations that: (a) advanced a typology of relations revising and making finer-grained distinctions within the former broad categories of formal and material relations [37]; (b) clarified the connection between these different types of relations and their different types of *truthmakers* [24, 25][5].

4. Ontology-Driven Conceptual Modeling

In conceptual modeling, the idea that *"data are fragments of a theory of the real-world and data processing is about manipulating models of such a theory"* was there since Mealy's seminal paper entitled 'Another Look on Data' [46]. In fact, Mealy's paper includes the first mention of the term 'ontology' in the Computer and Information Science literature. A number of fundamental conceptual modeling issues of an ontological nature were also discussed in Bill Kent's classic book 'Data and Reality' [44]. This book brought attention to issues like identity, unity and classification, and started exposing the subtleties of fundamental conceptual modeling constructs such as relationships.

However, neither Mealy's paper nor Kent's book tried to actually develop comprehensive ontological foundations for conceptual modeling. Perhaps the first corpus of work to attempt that goal was reported in the series of publications initiated by Yair Wand, Ron Weber and colleagues [64, 63, 65] in the late 80's. Instead of developing a new ontology themselves, Wand and Weber proposed an adaptation of the ontological theory put forth by the Argentinean physicist and philosopher of science Mario Bunge. The result of this effort came to be known as the BWW (Bunge-Wand-Weber) ontology. The authors then employed this theory to evaluate a number of conceptual modeling languages including NIAM [66], ER [62], UML [7] and OWL [4].

Despite the pioneering nature of these efforts, and the guidelines proposed for building ontologically sound conceptual models, the conceptual modeling languages used were still *ontologically neutral languages*. In the beginning of 2000, there was no conceptual modeling language fully-designed on the basis of a formal ontological theory. By that we mean a language that: (i) contains as modeling primitives the ontological distinctions put forth by a foundational ontology; (ii) restricts possible interpretations to intended ones. In summary, paraphrasing Nicola's dictum [12], a language whose abstract syntax and semantics were not neutral w.r.t. ontological assumptions. Moreover, in conformance with Nicola's second aforementioned claim, the system of ontological categories a conceptual modeling language should commit to should be one that takes human language and cognition seriously, i.e., a descriptive ontology (as opposed to a revisionary ontology such as Bunge's and, hence, BWW - see discussion in [22]).

In the early 2000's, Guizzardi and Wagner initiated a research program aimed at exactly such an objective, i.e., proposing an ontological-level conceptual modeling language satisfying (i) and (ii) above [34, 35]. However, in order to that, they needed a reference ontology that could provide proper foundations for conceptual modeling's most fundamental constructs. As an extension and evolution of a combination of DOLCE, GFO and the ontology of property types underlying OntoClean, the authors then proposed the Unified Foundational Ontology (UFO) [38][6]. In [39], also together with Nicola, they introduced a part of this future language (an extension of UML) based on one of the first

---

[5] More technically, between different types of relational propositions and different types of ontological entities that can make true these relational propositions

[6] According to the survey presented at [10], UFO became one of the most influential foundational ontologies in Conceptual Modeling.

of these micro-theories comprising UFO, namely, a theory dealing with Entity Types and the taxonomic structures involving them. That specific proposal can be seen as an evolution of Nicola's early proposals of an ontology of unary property types and ultimately as an evolution OntoClean. Inspired in Nicola's early ideas, the proposal systematically employed a number of formal meta-properties to create a theory and typology of entity types. These entity types were then used to propose finer-grained modeling primitives extending the basic notion of class in UML (see fig.3). Moreover, by employing the axioms of this theory, the proposed UML extension (a UML profile) includes constraints restricting the valid relations that could be established between these entity types.

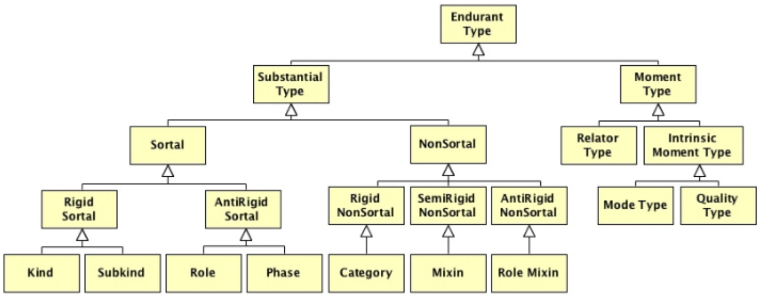

Figure 3. A Typology of Entity Types.

This ontologically well-founded version of UML came to be later known as OntoUML [38]. In figure 4, we can see an example of an OntoUML model employing these ontological distinctions. As one can observe, the language makes explicit distinctions such as being (necessarily) of a Kind (e.g., Person), being (contingently) in a certain Phase (e.g., Living Person) and playing (contingely) a certain role (e.g., Private Customer) in a relational context (e.g., a Service Agreement). Moreover, it explicitly models types that instantiated by instances of multiple kinds (e.g., the non-sortal type Customer that can have as instances both people and organizations).

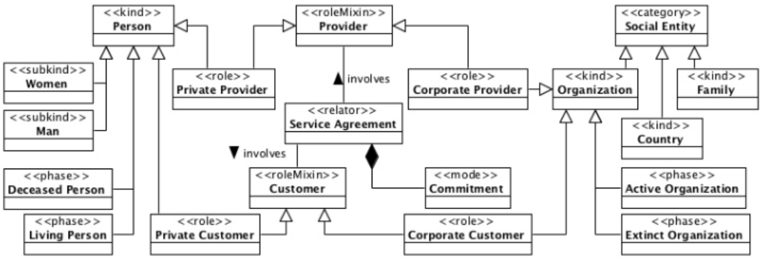

Figure 4. An Example of an OntoUML Model (from [32])

Firstly, this theory allowed for bringing some conceptual clarification to the notions involving Entity Types in conceptual modeling. Notions such as Type, Kind, Class, Role, Phase and Mixin were frequently discussed for many years in the literature of Conceptual and Object-Oriented Modeling. However, up to that point, there was little consensus about their definition.

Secondly, these ontological meta-properties allowed for more than just defining a formal ontological semantics for these notions. They also provided precise methodological guidelines for choosing how to model things in the universe of discourse. For example, suppose that in a given conceptualization of the domain, the type Student was conceive as: (a) a substantial type, i.e., its instances were existentially independent entities like you and me; (b) a sortal, i.e., all its instances are of the same Kind (e.g., Persons); (c) anti-rigid, i.e., no instance of student is necessarily an instance of student; (d) relationally dependent, i.e., instances of Person become (or cease to be) due to a change in a relational property (their enrollment)[7]. So, as a consequence, we would have that, in that particular conceptualization, the type Student should be modeled as a Role.

Thirdly, as demonstrated in [55, 68], the axiomatization of each of these types in the theory actually defines a particular micro-theory (e.g., a micro-theory defining what Roles are and how they behave w.r.t. to other ontological categories) and these micro-theories, in turn, constrain the way the primitives representing these types can appear in conceptual models. In fact, in a modeling language following this theory, constructs representing roles, phase, mixins can only appear in certain configurations forming patterns of a true ontological nature. As a result, the resulting modeling language can actually serve as an Ontology Pattern Language, whose modeling primitives are not low-level primitives (such as class, relation, attribute) but high-level building blocks (Ontology Design Patterns). The advantages of having this include knowledge reuse, agility in the construction of models, a smoother learning curve for novices, and greater uniformity for models.

Finally, as demonstrated in [8], these ontological distinctions can also offer unique support for complexity management of large conceptual models by providing natural criteria for model modularization.

The new theory of relations and relationships (including unary relationships, i.e., qualities) proposed in [24, 23] also allows for some methodological support for recognizing to what ontological kind a domain relation belongs, by systematically searching and exposing the nature of the truthmakers of these relations. Once more, by following a set of ontological meta-properties (e.g., descriptive/non-descriptive, internal/external, essential/non-essential), this theory allows for the clarification and organization of the space of relation types in ontology and allowed for the development of a methodology for the modeling of relations that is evolving towards what Nicola called (half-joking) *"OntoClean for Relations"*. Furthermore, as demonstrated in [25], once more, by using the categories of relations proposed by this theory, a number of modeling patterns for the modeling of relations was proposed.

This new theory of relationships is, in fact, part of a new theory of dependent endurants (aspects). As previously mentioned, the theory proposes that a hallmark of all endurants is their ability to qualitatively change while maintaining their identity, i.e., endurants are the natural bearers of modal properties, they have essence and accidents and, hence, they could have been different from what they are. Moreover, identity for all endurants are determined by the unique ultimate sortal type (i.e., the kind) they instantiate. When connecting this theory with the previously mentioned theory of entity types, it becomes clear that all the previously discussed distinctions among entity types are actually

---

[7] One should not mix up the notions of existential dependence and relational dependence. The former applies to individuals and the latter to types. For example, although the type Student is relationally dependent, the instances of Students (e.g., John and Mary) exist independently of other existents [29].

distinctions among endurant types. In other words, distinctions such as Kinds, Phases, Roles, Mixins (and versions of all the design patterns connected to them) can now be applied to types whose instances are endurants of all types and not only independent endurants (substantials)! For example, the particular marriage between John and Mary is of the Marriage KIND but it can be in a "Full-Separation of Assets" PHASE and it can play the ROLE of a "Marriage that is Legally Recognized in Brazil".

A new theory connecting these two former theories (the theory of entity types and the theory of aspects) gave rise in [32] to a new version of the Ontology-Driven Conceptual Modeling language OntoUML. As one can observe in figure 5, in this new combined theory, we have orthogonal classifications, namely: regarding the previously mentioned meta-properties distinguishing entity types (e.g., kind, phase, role, mixin); regarding the ontological nature of the instances of the endurant type in question (e.g., object type, quality type, relationship type). In figure 6, we have an example of a model represented in this new version of OntoUML (dubbed OntoUML 2.0). As one can observe, in this version of the language, we can employ the previously discussed ontological type distinctions also to characterize aspect types. For instance, while a relationship is (necessarily) of particular kind (e.g., Civil Partnership), it can be (contingely) in a certain phase (e.g., Longer-Term Relationship) and it can contingently play a role (e.g., Stable Civil Partnership) in the scope of a relational context (e.g., a legal recognition relation connecting it to a legal jurisdiction - not shown in the figure).

As previously discussed, this new theory of aspects has a strong connection to an ontological account of events proposed in collaboration with Nicola. In [33], the authors use this theory to address a fundamental problem in the modeling of events in structural conceptual models (roughly data models, information models). Historically, events have rarely been considered as first-class citizens in structural models. Recently, a number of authors have made a strong case advocating the explicit reification of events in these models [50, 1]. However, as demonstrated in [33], if we want: (a) to represent on-going events in these models; (b) while maintaining a classic formal semantics for object identifiers (OIDs); and (c) conforming to the classic view of events in philosophy, then we have as a consequence that our conceptual models can only represent past events, which are necessarily immutable in all sense, i.e., which cannot possibly be different in any respect! In order to address the issue of future events and the illusion of change in event properties, the authors propose a Design Pattern based on this ontological theory. This pattern is an ontologically well-founded engineering tool for dealing with a problem that was hitherto neglected or only naively addressed in the literature of conceptual modeling. Finally, in [18], Nicola builds on this notion of event (and its connection to aspects) to propose a non-classical view of events and a novel way of capturing mutable and future events in conceptual modeling.

OntoUML has its syntax (defined both as a metamodel [29] and as a Graph Grammar [68]) and (formal and real-world) semantics formally defined in terms of this ontological theory. This allowed for the development of a number of model-based computational supporting tools for pattern-based model construction [55], formal verification, verbalization and code generation [28]. In particular, it allowed for the development of a novel strategy of conceptual model validation via visual simulation [3]. The approach directly implements Nicola's notion of Ontological Commitment, which takes an ontology to be a theory that constrains the set of elements in a syntactical specification to approximate, as much as possible, the set of (logical) models of that specification to the set of

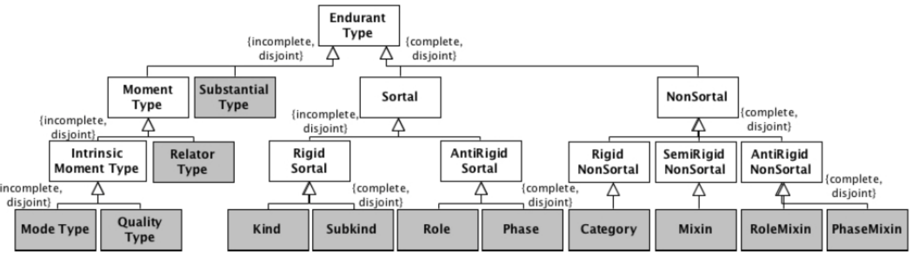

Figure 5. A Typology of Entity Types for OntoUML 2.0 (from [32]).

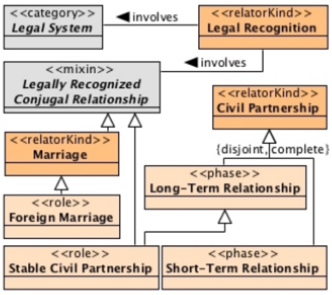

Figure 6. An Example of an OntoUML 2.0 Model [32]).

intended ones [16]. Following this idea, in the computational support for this approach, we have the automatic generation of visual instances (exemplars) of a given conceptual model (specification) such that the modeler can be confronted with what her model is actually saying on her behalf. In other words, the strategy is to systematically contrast the set of formally-valid instances of a given conceptual model (i.e., its logical models, which are automatically generated by the visual simulator) with the set of intended instances of that model (i.e., instances that represent state of affairs admissible by the underlying conceptualization), which exists only in the modeler's mind. Once the modeler detects a deviation between valid and intended instances (either due to *overconstraining* or *underconstraining* of the model), she rectifies the model, for instance, by the inclusion of formal domain-specific constraints. This approach, in turn, allowed for a new area of research in conceptual modeling, namely, the study of *Ontological Anti-Patterns in Conceptual Models*. In three different empirical studies, [56, 58, 57] managed to show that this approach is also able to detect recurrent structures that tend to cause the deviations between the sets of valid and intended logical models. Once these anti-patterns are catalogued, they were able to devise systematic computational solutions that are able to eliminate these anti-patterns.

5. Final Considerations

In this paper, we highlighted three fundamental tenets of Nicola's thought regarding the role of Formal Ontology for Conceptual Modeling and Knowledge Representation. These are: (1) Languages that are supposed to represent conceptualizations of reality cannot be ontologically neutral. Instead, they should make an explicitly commitment to a formal ontological theory; (2) the role of Ontology in Conceptual Modeling is not providing reference domain models capturing single-views of the world that all stakeholders commit to. In other words, it is not about prescribing a single view of reality to all agents and their applications. In contrast, it is about giving stakeholders philosophically and cognitively well-founded theoretical tools such that they make consistent choices in their worldviews and such that these choices can be made transparent to other stakeholders. To put it simply: ontology is not here to make us all agree but to help us understand if, when and precisely in which points we agree and disagree; (3) Conceptual Modeling is about representing aspects of the real-world for the purposes of supporting human users in tasks such as domain understanding and learning, meaning negotiation and problem solving. For this reason, a conceptual modeling language should commit to a *descriptive* ontological theory (as opposed to a *revisionary* one) that takes human language and cognition seriously.

We also showed here how Nicola's philosophy of Conceptual Modeling and, in particular, these three tenets, have strongly influenced the design of OntoUML research program for Ontology-Driven Conceptual Modeling. Over the years, OntoUML has been successfully employed in academic, industrial and governmental settings to create conceptual models in a number of different domains, including Geology, Biodiversity Management, Organ Donation, Petroleum Reservoir Modeling, Disaster Management, Context Modeling, Datawarehousing, Enterprise Architecture, Data Provenance, Measurement, Logistics, Complex Media Management, Telecommunications, Heart Electrophysiology, among many others [38, 2]. Moreover, it has influenced the design of some aspects of influential conceptual modeling languages (e.g., ORM [41]) and has been considered as a possible candidate for contributing to the OMG SIMF (Semantic Information Model Federation) standardization request for proposal [11]. Finally, empirical evidence shows that OntoUML significantly contributes to improving the quality of conceptual models without requiring an additional effort to produce them [61]. We believe that these benefits brought to Conceptual Modeling theory and practice by OntoUML are strongly a product of the direct influence of Nicola's philosophy in its design.

Nicola's work has profoundly influenced conceptual modeling languages by offering an ontological level of analysis for their primitive concepts. We anticipate that in the future, no conceptual modeling language will be considered complete until it has revealed its ontological commitments. And for this advance, it is Nicola and his collaborators who deserve full credit.

Acknowledgements

We would like to thank Nicola for the many years of pleasant, inspiring and fruitful collaboration. We have a great respect for his ideas as well as his uncompromising attitude towards science. The first author is also grateful to the members of the NEMO and CORE research groups for their collaboration in many of the results reported herein.

References

[1] Gove N. Allen and Salvatore T. March. The effects of state-based and event-based data representation on user performance in query formulation tasks. *MIS Quarterly*, 30(2):269–290, 2006.

[2] Bruce T. Bauman. Prying apart semantics and implementation: Generating xml schemata directly from ontologically sound conceptual models. In *Balisage Markup Conference*, 2009.

[3] Alessander Botti Benevides, Giancarlo Guizzardi, Bernardo Ferreira Bastos Braga, and João Paulo A. Almeida. Validating modal aspects of ontouml conceptual models using automatically generated visual world structures. *J. UCS*, 16(20):2904–2933, 2010.

[4] Palash Bera and Yair Wand. Analyzing owl using a philosophy-based ontology. In *Formal Ontology in Information Systems*, 2004.

[5] Ronald J. Brachman. What's in a concept: Structural foundations for semantic networks. *International Journal of Man-Machine Studies*, 9(2):127–152, 1977.

[6] Peter P. Chen. The entity-relationship model: Towards a unified view of data. *ACM Transactions on Database Systems*, 1(1), 1976.

[7] Joerg Evermann and Yair Wand. Towards ontologically based semantics for UML constructs. In *20th International Conference on Conceptual Modeling (ER'2001), Yokohama, Japan, November 27-30, 2001, Proceedings*, pages 354–367, 2001.

[8] Guylerme Figueiredo, Amelie Duchardt, Maria M. Hedblom, and Giancarlo Guizzardi. Breaking into pieces: An ontological approach to conceptual model complexity management. In *12th Intl. Conf. on Research Challenges in Information Science, RCIS 2018, Nantes, France, May 29-31, 2018*, pages 1–10. IEEE, 2018.

[9] Kit Fine. Things and their parts. *Midwest Studies in Philosophy*, 23(1):61–74, 1999.

[10] Frederik Gailly, Ben Roelens, and Giancarlo Guizzardi. The design of a core value ontology using ontology patterns. In Sebastian Link and Juan Trujillo, editors, *Advances in Conceptual Modeling, Gifu, Japan, November 14-17, 2016, Proceedings*, volume 9975 of *LNCS*, pages 183–193, 2016.

[11] Object Management Group. Semantic information model federation (simf): Candidates and gaps, available online: http://www.omgwiki.org/architecture-ecosystem/.

[12] Nicola Guarino. A concise presentation if ITL. *SIGART Bulletin*, 2(3):61–69, 1991.

[13] Nicola Guarino. Concepts, attributes and arbitrary relations. *Data Knowl. Eng.*, 8:249–261, 1992.

[14] Nicola Guarino. Kinds of relations: Some methodological principles for using description logics. In Robert M. MacGregor, Deborah L. McGuinness, Eric Mays, and Thomas A. Russ, editors, *Issues in Description Logics: Users Meet Developers, Boston, MA, USA, October 1992*, AAAI Fall Symposium Series, pages 39–44. AAAI Press, 1992.

[15] Nicola Guarino. The ontological level. In *Philosophy and the Cognitive Sciences*, pages 443–456. Holder-Pichler-Tempsky, 1994.

[16] Nicola Guarino. Formal ontology in information systems. In *Proceedings of FOIS'98*, pages 3–15, 1998.

[17] Nicola Guarino. The role of identity conditions in ontology design. In Christian Freksa and David M. Mark, editors, *Spatial Information Theory: Cognitive and Computational Foundations of Geographic Information Science, International Conference COSIT '99, Stade, Germany, August 25-29, 1999, Proceedings*, volume 1661 of *LNCS*, pages 221–234. Springer, 1999.

[18] Nicola Guarino. On the semantics of ongoing and future occurrence identifiers. In Heinrich C. Mayr, Giancarlo Guizzardi, Hui Ma, and Oscar Pastor, editors, *36th International Conference on Conceptual Modeling, ER 2017, Valencia, Spain, November 6-9, 2017*, volume 10650 of *LNCS*, pages 477–490. Springer, 2017.

[19] Nicola Guarino, Massimiliano Carrara, and Pierdaniele Giaretta. Formalizing ontological commitment. In Barbara Hayes-Roth and Richard E. Korf, editors, *Proc. of the 12th National Conference on Artificial Intelligence, Seattle, WA, USA, July 31 - August 4, 1994, Volume 1.*, pages 560–567. AAAI Press / The MIT Press, 1994.

[20] Nicola Guarino, Massimiliano Carrara, and Pierdaniele Giaretta. An ontology of meta-level categories. In Jon Doyle, Erik Sandewall, and Pietro Torasso, editors, *Proc. of the 4th Intl. Conf. on Principles of Knowledge Representation and Reasoning (KR'94). Bonn, 1994.*, pages 270–280. Morgan Kaufmann, 1994.

[21] Nicola Guarino and Pierdaniele Giaretta. Ontologies and knowledge bases: Towards a terminological clarification. In *Towards very Large Knowledge bases: Knowledge Building and Knowledge sharing*, pages 25–32. IOS Press, 1995.

[22] Nicola Guarino and Giancarlo Guizzardi. In the defense of ontological foundations for conceptual modelling. *Scandinavian J. Inf. Systems*, 18(1):1, 2006.

[23] Nicola Guarino and Giancarlo Guizzardi. "we need to discuss the relationship": Revisiting relationships as modeling constructs. In Jelena Zdravkovic, Marite Kirikova, and Paul Johannesson, editors, *27th Intl. Conf. on Advanced Information Systems Engineering (CAiSE 2015), Stockholm, Sweden, June 8-12, 2015, Proceedings*, volume 9097 of *LNCS*, pages 279–294. Springer, 2015.

[24] Nicola Guarino and Giancarlo Guizzardi. Relationships and events: Towards a general theory of reification and truthmaking. In Giovanni Adorni, Stefano Cagnoni, Marco Gori, and Marco Maratea, editors, *AI*IA 2016: Advances in Artificial Intelligence, Genova, Italy, November 29 - December 1, 2016*, volume 10037 of *LNCS*, pages 237–249. Springer, 2016.

[25] Nicola Guarino, Tiago Prince Sales, and Giancarlo Guizzardi. Reification and truthmaking patterns. In *Conceptual Modeling - 37th International Conference, ER 2018, Xi'an, China, October 22-25, 2018, Proceedings*, pages 151–165, 2018.

[26] Nicola Guarino and Christopher A. Welty. Ontological analysis of taxonomic relationships. In Alberto H. F. Laender, Stephen W. Liddle, and Veda C. Storey, editors, *19th International Conference on Conceptual Modeling - ER 2000, Salt Lake City, Utah, USA*, volume 1920 of *LNCS*, pages 210–224. Springer, 2000.

[27] Nicola Guarino and Christopher A. Welty. Evaluating ontological decisions with ontoclean. *Commun. ACM*, 45(2):61–65, 2002.

[28] John Guerson, Tiago Prince Sales, Giancarlo Guizzardi, and João Paulo A. Almeida. Ontouml lightweight editor: A model-based environment to build, evaluate and implement reference ontologies. In Jens Kolb, Barbara Weber, Sylvain Hallé, Wolfgang Mayer, Aditya K. Ghose, and Georg Grossmann, editors, *19th IEEE International Enterprise Distributed Object Computing Workshop, EDOC Workshops 2015, Adelaide, Australia, September 21-25, 2015*, pages 144–147. IEEE Computer Society, 2015.

[29] Giancarlo Guizzardi. *Ontological foundations for structural conceptual models*. PhD thesis, University of Twente, Enschede, 2005.

[30] Giancarlo Guizzardi. On ontology, ontologies, conceptualizations, modeling languages, and (meta)models. In Olegas Vasilecas, Johann Eder, and Albertas Caplinskas, editors, *Databases and Information Systems IV - Selected Papers from the Seventh International Baltic Conference, DB&IS 2006, July 3-6, 2006, Vilnius, Lithuania*, volume 155 of *Frontiers in Artificial Intelligence and Applications*, pages 18–39. IOS Press, 2006.

[31] Giancarlo Guizzardi. Ontological patterns, anti-patterns and pattern languages for next-generation conceptual modeling. In Eric S. K. Yu, Gillian Dobbie, Matthias Jarke, and Sandeep Purao, editors, *Conceptual Modeling - 33rd International Conference, ER 2014, Atlanta, GA, USA, October 27-29, 2014. Proceedings*, volume 8824 of *Lecture Notes in Computer Science*, pages 13–27. Springer, 2014.

[32] Giancarlo Guizzardi, Claudenir M. Fonseca, Alessander Botti Benevides, João Paulo A. Almeida, Daniele Porello, and Tiago Prince Sales. Endurant types in ontology-driven conceptual modeling: Towards ontouml 2.0. In *Conceptual Modeling - 37th International Conference, ER 2018, Xi'an, China, October 22-25, 2018, Proceedings*, pages 136–150, 2018.

[33] Giancarlo Guizzardi, Nicola Guarino, and João Paulo A. Almeida. Ontological considerations about the representation of events and endurants in business models. In Marcello La Rosa, Peter Loos, and Oscar Pastor, editors, *Business Process Management - 14th International Conference, BPM 2016, Rio de Janeiro, Brazil, September 18-22, 2016. Proceedings*, volume 9850 of *LNCS*, pages 20–36. Springer, 2016.

[34] Giancarlo Guizzardi, Heinrich Herre, and Gerd Wagner. On the general ontological foundations of conceptual modeling. In Stefano Spaccapietra, Salvatore T. March, and Yahiko Kambayashi, editors, *Conceptual Modeling - ER 2002, 21st International Conference on Conceptual Modeling, Tampere, Finland, October 7-11, 2002, Proceedings*, volume 2503 of *Lecture Notes in Computer Science*, pages 65–78. Springer, 2002.

[35] Giancarlo Guizzardi, Heinrich Herre, and Gerd Wagner. Towards ontological foundations for UML conceptual models. In Robert Meersman and Zahir Tari, editors, *On the Move to Meaningful Internet Systems, 2002 - DOA/CoopIS/ODBASE 2002 Confederated International Conferences DOA, CoopIS and ODBASE 2002 Irvine, California, USA, October 30 - November 1, 2002, Proceedings*, volume 2519 of *Lecture Notes in Computer Science*, pages 1100–1117. Springer, 2002.

[36] Giancarlo Guizzardi, Claudio Masolo, and Stefano Borgo. In defense of a trope-based ontology for conceptual modeling: An example with the foundations of attributes, weak entities and datatypes. In David W. Embley, Antoni Olivé, and Sudha Ram, editors, *Conceptual Modeling - ER 2006, 25th International Conference on Conceptual Modeling, Tucson, AZ, USA, November 6-9, 2006, Proceedings*, volume 4215 of *Lecture Notes in Computer Science*, pages 112–125. Springer, 2006.

[37] Giancarlo Guizzardi and Gerd Wagner. What's in a relationship: An ontological analysis. In Qing Li, Stefano Spaccapietra, Eric S. K. Yu, and Antoni Olivé, editors, *Conceptual Modeling - ER 2008, 27th International Conference on Conceptual Modeling, Barcelona, Spain, October 20-24, 2008. Proceedings*, volume 5231 of *Lecture Notes in Computer Science*, pages 83–97. Springer, 2008.

[38] Giancarlo Guizzardi, Gerd Wagner, João Paulo Andrade Almeida, and Renata S. S. Guizzardi. Towards ontological foundations for conceptual modeling: The unified foundational ontology (UFO) story. *Applied Ontology*, 10(3-4):259–271, 2015.

[39] Giancarlo Guizzardi, Gerd Wagner, Nicola Guarino, and Marten van Sinderen. An ontologically well-founded profile for UML conceptual models. In Anne Persson and Janis Stirna, editors, *Advanced Information Systems Engineering, 16th International Conference, CAiSE 2004, Riga, Latvia, June 7-11, 2004, Proceedings*, volume 3084 of *LNCS*, pages 112–126. Springer, 2004.

[40] Giancarlo Guizzardi and Veruska Zamborlini. Using a trope-based foundational ontology for bridging different areas of concern in ontology-driven conceptual modeling. *Sci. Comput. Program.*, 96:417–443, 2014.

[41] Terry Halpin. *Information Modeling and Relational Dababases*. Morgan Kauffman, 2008.

[42] Rinke Hoekstra, Joost Breuker, Marcello Di Bello, and Alexander Boer. Lkif core: Principled ontology development for the legal domain. In *Proceedings of the 2009 Conference on Law, Ontologies and the Semantic Web*, pages 21–52, Amsterdam, The Netherlands, The Netherlands, 2009. IOS Press.

[43] Ivan Jureta, John Mylopoulos, and Stéphane Faulkner. A core ontology for requirements. *Applied Ontology*, 4(3-4):169–244, 2009.

[44] Bill Kent. *Data and Reality*. Elsevier Science, 1978.

[45] Hector Levesque and John Mylopoulos. A procedural semantics for semantic networks. *Associative Networks: Representation and Use of Knowledge by Computers*, pages 93–120, 1979.

[46] George H. Mealy. Another look at data. In *AFIPS Conference Proceedings, Thompson Books, London*, 1968.

[47] John Mylopoulos. Information modeling in the time of the revolution. *Inf. Syst.*, 23(3-4):127–155, May 1998.

[48] John Mylopoulos, Philip A. Bernstein, and Harry K. T. Wong. A language facility for designing database-intensive applications. *ACM Trans. Database Syst.*, 5(2):185–207, 1980.

[49] K. Nyggard O-J. Dahl, B. Myhrhaug. Some features of the simula 67 language. In *2nd Conference on Applications of Simulations*, pages 29–31, 1968.

[50] Antoni Olivé and Ruth Raventós. Modeling events as entities in object-oriented conceptual modeling languages. *Data Knowl. Eng.*, 58(3):243–262, 2006.

[51] Florian Probst. Observations, measurements and semantic reference spaces. *Applied Ontology*, 3(1-2):63–89, 2008.

[52] M. Ross Quillian. *Semantic memory, Semantic Information Processing*. MIT Press, 1968.

[53] W. V. O. Quine. Ontological relativity and other essays. Columbia University Press, 1969.

[54] Douglas T. Ross. Structured analysis (SA): A language for communicating ideas. *IEEE Trans. Software Eng.*, 3(1):16–34, 1977.

[55] Fabiano Borges Ruy, Giancarlo Guizzardi, Ricardo de Almeida Falbo, Cássio Chaves Reginato, and Victor A. Santos. From reference ontologies to ontology patterns and back. *Data Knowl. Eng.*, 109:41–69, 2017.

[56] Tiago Prince Sales and Giancarlo Guizzardi. Ontological anti-patterns: empirically uncovered error-prone structures in ontology-driven conceptual models. *Data Knowl. Eng.*, 99:72–104, 2015.

[57] Tiago Prince Sales and Giancarlo Guizzardi. Anti-patterns in ontology-driven conceptual modeling: The case of role modeling in ontouml. In Pascal Hitzler, Aldo Gangemi, Krzysztof Janowicz, Adila Krisnadhi, and Valentina Presutti, editors, *Ontology Engineering with Ontology Design Patterns - Foundations and Applications*, volume 25 of *Studies on the Semantic Web*, pages 161–187. IOS Press, 2016.

[58] Tiago Prince Sales and Giancarlo Guizzardi. "is it a fleet or a collection of ships?": Ontological anti-patterns in the modeling of part-whole relations. In *21st European Conference in Advances in Databases and Information Systems ADBIS 2017, Nicosia, Cyprus, September 24-27, 2017*, pages 28–41, 2017.

[59] Ansgar Scherp, Thomas Franz, Carsten Saathoff, and Steffen Staab. F–a model of events based on the foundational ontology dolce+dns ultralight. In *Proc. of the Fifth International Conference on Knowledge Capture*, K-CAP '09, pages 137–144, New York, NY, USA, 2009. ACM.

[60] Mohamed Turki, Gilles Kassel, Inès Saad, and Faïez Gargouri. A core ontology of business processes based on dolce. *Journal on Data Semantics*, 5:165–177, 2016.

[61] Michaël Verdonck, Frederik Gailly, Robert Pergl, Giancarlo Guizzardi, Beatriz Martins, and Oscar Pastor. Comparing traditional conceptual modeling with ontology-driven conceptual modeling: An empirical study. *Information Systems*, 2018.

[62] Yair Wand, Veda C. Storey, and Ron Weber. An ontological analysis of the relationship construct in conceptual modeling. *ACM Trans. Database Syst.*, 24(4):494–528, 1999.

[63] Yair Wand and Ron Weber. An ontological analysis of some fundamental information systems concepts. In *Proceedings of the 9th International Conference on Information Systems, ICIS 1988, Minneapolis, Minnesota, USA, 1988*, page 35, 1988.

[64] Yair Wand and Ron Weber. An ontological evaluation of systems analysis and design methods. In *Information System Concepts: An In-depth Analysis, Proceedings of the IFIP TC8/WG8.1 Working Conference on Information System Concepts: An In-depth Analysis (ISCO 1989), Namur, Belgium, 18-20 October 1989.*, pages 79–108, 1989.

[65] Yair Wand and Ron Weber. An ontological model of an information system. *IEEE Trans. Software Eng.*, 16(11):1282–1292, 1990.

[66] Ron Weber and Yanchun Zhang. An ontological evaluation of niam's grammar for conceptual schema diagrams. In *Proceedings of the 12th International Conference on Information Systems, ICIS 1991, New York, NY, USA, December 16-18, 1991*, pages 75–82, 1991.

[67] Christopher A. Welty and Nicola Guarino. Supporting ontological analysis of taxonomic relationships. *Data Knowl. Eng.*, 39(1):51–74, 2001.

[68] Eduardo Zambon and Giancarlo Guizzardi. Formal definition of a general ontology pattern language using a graph grammar. In Maria Ganzha, Leszek A. Maciaszek, and Marcin Paprzycki, editors, *Proceedings of the 2017 Federated Conference on Computer Science and Information Systems, FedCSIS 2017, Prague, Czech Republic, September 3-6, 2017.*, pages 1–10, 2017.