Identification of Semantic Anti-Patterns in Ontology-Driven Conceptual Modeling via Visual Simulation

Tiago Prince SALES[a], Pedro Paulo Favato BARCELOS[b] and Giancarlo GUIZZARDI[c]
Ontology and Conceptual Modeling Research Group (NEMO)
[a, c] *Computer Science Department, Federal University of Espírito Santo, Brazil*
[b] *Electrical Engineering Department, Federal University of Espírito Santo, Brazil*

Abstract. The construction of large-scale reference conceptual models and ontologies is a complex engineering activity. To develop high quality models, a modeler must have the support of expressive engineering tools such as theoretically well-founded modeling languages and methodologies, ontological patterns and computational environments. Patterns and Anti-Patterns are known to be an efficient way to reuse knowledge from experts' successful past experiences. This paper proposes a set of Semantic Anti-Patterns for ontology engineering. These anti-patterns capture error prone modeling decisions which can result in the creation of models that allow for unintended model instances (representing undesired state of affairs). The anti-patterns presented here have been empirically elicited through an approach of ontology conceptual models validation via visual simulation.

Keywords. Ontology-Driven Conceptual Modeling, Semantic Anti-Patterns, OntoUML, Conceptual Model Visual Simulation

Introduction

Conceptual modeling is a complex activity. In [1], an analogy is made between the construction of large ontologies (as reference models) and the programming of large computer systems, quoting the famous E. W. Dijkstra´s ACM Turing lecture entitled "*The Humble Programmer*". In both cases, we have an acknowledgement of the limitations of the human mind to address the large and fast increasingly intrinsic complexity of these types of activities. For this reason, human conceptual modelers and ontologists should make use of a number of suitable complexity management engineering tools to maximize the chances of a successful outcome of this enterprise. As discussed in [1], among these tools, we have modeling languages and methodologies, patterns and automated supporting environments.

In recent years, there has been a growing interest in the use of Ontologically Well-Founded Conceptual Modeling languages to support the construction and management of these complex artifacts. OntoUML is an example of a conceptual modeling language which metamodel [2] has been designed to comply with the ontological distinctions and axiomatization of a theoretically well-grounded foundational ontology later dubbed UFO (Unified Foundational Ontology) [3]. This language has been successfully employed in a number of industrial projects in several different domains, such as

Petroleum and Gas [4] and News Information Management [5]. In fact, recently, it has been considered as a possible candidate for contributing to the OMG SIMF (Semantic Information Model Federation) standardization *request for proposal* [6] after a significant number of successful applications in real-world engineering settings [7]. Besides the modeling language itself, this approach also comprises a number of model-based environments for model construction, verbalization, code generation, formal verification and validation ([8], [9]). In particular, the validation strategy employed there makes use of an approach based on visual model simulation [9]. Finally, the OntoUML approach also comprises a number of ontological design patterns [1].

A Design Pattern describes a standard solution to a recurrent problem, and a system of patterns (or a Pattern Language) defines a system of primitives of a higher-level of granularity to talk and reason about design [10]. Design Patterns have been successfully employed for years in areas such as software and data engineering and, in recent years, there has been a growing interest in defining patterns for ontology engineering ([1], [3] and [11]). However, as discussed in [1], most of the available so-called ontology patterns fall in one single category, namely, *transformation (or codification) patterns*, i.e., patterns that provide a standard solution for coding high-expressive ontology conceptual models in languages of lower expressivity (e.g., OWL or RDFS).

The term *Anti-Pattern* was coined in [12] with the following definition: "*[An] Anti-pattern is just like pattern, except that instead of a solution it gives something that looks superficially like a solution, but isn't one*". An anti-pattern is a recurrent decision for a specific scenario that usually results in more negative consequences than positives ones. In this paper, we are interested in one specific sort of Anti-Patterns, namely, model structures that, albeit producing syntactically valid ontology conceptual models, are prone to result in unintended domain representations. In other words, we are interested in configurations that when used in a model will typically cause the set of valid (possible) instances of that model to differ from the set of instances representing intended state of affairs in that domain [3]. We name here these configurations *Semantic Anti-Patterns*.

This paper contributes to the identification of Semantic Anti-Patterns in Ontology-Driven Conceptual Modeling by carrying out an empirical qualitative approach. We do that by employing the visual simulation capabilities embedded in OntoUML editor [9]. By taking advantage of this capability of the OntoUML supporting tools, we illustrate our Semantic Anti-Patterns here using this language. However, most of the configurations elicited here can be produced in other modeling languages such as UML, ER and the Web Ontology Language (OWL).

The remainder of this paper is organized as follows: in Section 1, we briefly elaborate on the modeling language OntoUML; Section 2 discusses (also briefly) the approach for model validation via visual simulation embedded in the OntoUML editor; Section 3 presents the methodological steps used in this research; Section 4 presents the elicited *Semantic Anti-Patterns* with their undesired consequences and possible solutions; Finally, Section 5 presents some final considerations of this work.

1. OntoUML

The OntoUML metamodel [2] contains: (i) elements that represent ontological distinctions prescribed by an underlying foundational ontology; (ii) constraints that govern the possible relations that can be established between these elements. These points are illustrated below using ontological distinctions among the categories of object types (*Kind*, *Subkind* and *Roles*), trope types (*Relator*) and relations (*formal* and *material* relations). For an in depth presentation, formal characterization and empirical evidence for a number of the ontological categories underlying OntoUML, the reader is referred to [3].

In a simplified view we can state that: *Kinds* and *Subkinds* are types that aggregate all the *essential* properties of their instances and, for that reason, all instances of a given *Kind/Subkind* cannot cease to instantiate it without ceasing to exist (a meta-property known as *rigidity*). A *Kind* defines a uniform principle of identity which is obeyed by all its instances; *Subkinds* are rigid specializations of a *Kind* and inherit that principle of identity supplied by that unique subsuming *Kind*. A *Role*, in contrast, represents a number of properties which instances of a *Kind* have contingently and in a relational context. A stereotypical example can be appreciated when contrasting the *Kind* Person, the *Subkinds* Man and Woman (specializing Person) and the *Role* Student (also specializing Person).

A *Relator* is the objectification of a relational property (i.e., a relational trope). Relators are existentially dependent on a multitude of individuals, thus, *mediating* them. Examples of relators are Enrollments and Marriages. Relators are the foundation and truth-makers of the so-called *material* relations in the way, for instance, that the marriage between John and Mary founds (is the truth-maker of) the relation *is-married-to* between John and Mary (but also the relations *being-the-husband-of*, *being-the-wife-of*), or in the way that the Enrollment between Mick and the London School of Economics founds the relation *studies-at* between these two individuals. Contrary to material relations, *formal* relations hold directly between entities without requiring any intervening (connecting) individual. Examples include the relations of existential dependence and parthood but also *being-taller-than* between two individuals.

Regarding (i), OntoUML incorporates modeling constructs that represent all the aforementioned ontological categories (among many others) as modeling primitives of the language. Regarding (ii), the metamodel embeds constraints that govern the possible relations to be established between these categories. These constraints are derived from the very axiomatization of these categories in the underlying foundational ontology. Examples include: a *Role* (and a *Subkind*) must be a subtype of exactly one ultimate *Kind*; a role cannot be a supertype of a *Kind*; a relator must bear mediation relations to at least two distinct individuals, among many others[1].

2. Model Validation via Visual Simulation

As discussed in [3], the only grammatically correct models of OntoUML are ontologically consistent models. In other words, by incorporating ontological constraints in its metamodel, OntoUML proscribes the representation of ontologically non-admissible states of affair in domain ontologies represented in that language. However, as discussed in [9], the language cannot guarantee that, in a domain ontology represented in that language, only model instances representing *intended state of affairs* are admitted. This is because the admissibility of domain-specific states of affairs is a matter of factual knowledge, not a matter of consistent possibility [1].

---

[1] Formal proofs for these constraints can be found in [3].

To illustrate this point, take the following fictitious medical domain ontology representing the procedure of a transplant. In this case, we represent domain concepts such as Person, Transplant Surgeon, Transplant, Transplanted Organ, Organ Donor, Organ Donee, etc. The (obviously incomplete) model of Figure 1 capturing aspects of this domain does not violate any ontological rule; it would have done so, for example, had we placed Organ Donor as a supertype of Person, or had we represented the possibility of a Transplant without participants [3]. These two cases can be easily detected and proscribed by an editor such as the one proposed in [8]. However, there are still unintended model instances (according to a conceptualization assumed here for this domain) which are represented by valid instances of this model. One example is a state of affairs in which the Donor, the Donee and the Transplant Surgeon are one and the same Person. Please note that this state of affairs is only considered inadmissible due to domain-specific knowledge of social and natural laws. Consequently, it cannot be ruled out *a priori* by a domain independent system of ontological categories.

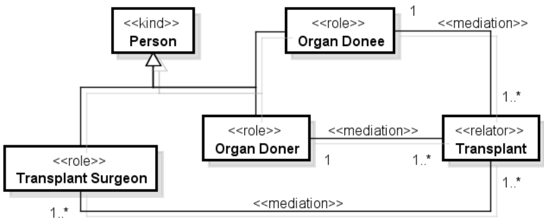


Figure 1. A fragment of an ontology for the transplant domain in which unintended instances are admitted

Guaranteeing the exclusion of unintended states of affairs without a computational support is a practically impossible task for any relevant domain. In particular, given that many fundamental ontological distinctions are modal in nature, in order to validate a model, one would have to take into consideration the possible valid instances of that model in all possible worlds.

In [9], the authors propose an automated approach for OntoUML which offers a contribution to this problem by supporting conceptual model validation via visual simulation. In the proposed tool, the ontologies are translated into Alloy [13], a logic language based on set theory, which is supported by an analyzer that exhaustively[2] generates possible instances for a given specification and also allows automatic checking of assertions' consistency. The generated instances of a given ontology are organized in a branching-time temporal structure, thus, serving as a visual simulator for the possible dynamics of entity creation, classification, association and destruction [9]. In [9], the modeler is then confronted with a visual representation of the snapshots in this world structure. These snapshots represent model instances that are deemed admissible by the ontology's current axiomatization. This enables modelers to detect unintended model instances (i.e., model instances that do not represent intended state of affairs) so that they can take the proper measures to rectify the model.

The comparison between admissible model instances, generated by the Alloy Analyzer, and the intended ones, obtained from domain experts or the ontology documentation, highlights possibly erroneous modeling decisions. The recording and categorization of these decisions for a set of OntoUML ontologies served as a basis for identifying the semantic anti-patterns proposed in this paper.

---

[2] To be precise, the Alloy analyzer exhausts the possible models given a finite context [13].

3. Methodology

The approach used in this work for the identification of the proposed set of anti-patterns was an empirical qualitative analysis. The idea was to simulate existing ontologies built in OntoUML ontologies by employing the approach described in section 2. In analysis reported here, we have decided to study the recurrence of these anti-patterns across: (i) different domains; (ii) different levels of modeling expertise in Ontology-Driven Conceptual Modeling; (iii) models of different sizes, maturity and complexity.

Regarding domain diversity, we have selected models in the following areas:

1. A Conceptual Model that describes a Brazilian Health Organization
2. A Conceptual Model that describes the Organizational Structure of Brazilian Federal Universities
3. A Conceptual Model that describes a Domain of Online Mentoring Activities
4. An Ontology representing the domain of Transport Optical Network Architectures
5. An Ontology in the Biodiversity Domain
6. A Heart Electrophysiology Reference Ontology
7. An Ontology in the Domain of Normative Acts
8. An Ontology of Public Tenders
9. An Ontology in the Domain of Brazilian Federal Organizational Structures

Regarding levels of expertise, we have classified as “beginners”, those modelers with less than one year of experience with OntoUML and its foundations. In contrast, we classified as “experienced”, those modelers that had worked with the language for two or more years and had applied the language in large-scale complex domains. In all the analyzed cases, the modelers involved in the creation of the models had a significant experience in traditional conceptual modeling approaches.

Finally, regarding scale and complexity, the investigated models can be characterized as follows: models (1-3) were produced by graduate students as final assignments of an *“Introduction to Ontology Engineering”* 60-hours course offered by the Graduate School on Computer Science of the Federal University of Espírito Santo[3]. These models cannot truly be considered as ontologies in the meaningful sense of the word, since they can hardly be said to represent the consensus of a community. Nonetheless, especially (1) and (2) were created by modelers with vast experience in the respective domains, despite having a low experience in ontology-driven conceptual modeling, in general, and OntoUML, in particular. Model (3) was produced by a modeler with low experience in OntoUML but with a significant experience in ontology modeling using an alternative approach, namely, the language LINGO extended with first-order logic Constraints; Model (4) was produced by experienced modelers in an industrial project. Moreover, the modelers had accessed to domain experts as well as a supporting international standard of the domain (ITU-T G.805). Finally, the resulting ontology was published in an important scientific forum in the area of Telecommunications [14]; Model (5) was developed in the Brazilian National Center for Amazon Research in collaboration with domain experts [15]; Model (6) was published in a renowned international journal in the area of Bioinformatics in a special issue of Biomedical ontologies [16]. In that publication, the authors demonstrate that

---

[3] These models can be found in their original in http//nemo.inf.ufes.br/courses/ontologyengineering**/**

the model is expressive enough to represent in a satisfactory manner a number of international standards (e.g., HL7, MIT-BIH); Models (7-8) were produced in a large-scale industrial project for the Brazilian Regulatory Agency for Land Transportation (ANTT). The models were produced by experienced modelers with vast experience in the language. Moreover, the modelers had constant access to normative documentation and to domain experts. Information regarding this project can be found in [17]. However, due to contractual confidential constraints, the models themselves could not be disclosed; finally, model (9) was produced by a group of modelers in the Brazilian Ministry of Planning [18]. The modelers were taught a 40 hours course in OntoUML and had a beginner's level expertise with the language and its foundations. However, this group was formed by experts in the domain who had a professional-level experience in traditional conceptual modeling.

Table 1 below summarizes the characterization of these ontologies.

Table 1. Properties of the ontologies subjects to the analysis and identification of the semantic anti-patterns

Ontology	Context of Model	Level of OntoUML Modeling Experience	#Class	#Assoc.	Additional Support for Validation
1	Graduate Course Assignment	Beginner	24	12	Additional Documentation in Graduate Course Assignment.
2	Graduate Course Assignment	Beginner	31	30	Additional Documentation in Graduate Course Assignment.
3	Graduate Course Assignment	Beginner	30	16	Additional Documentation in Graduate Course Assignment.
4	Industrial Project (Private Sector)	Experienced	190	122	Contact with Model Creators; Documentation of the domain is an International Standard; Associated Publication.
5	Research Project	Experienced	194	60	Extensive Additional Documentation in an associated Master Dissertation; Associated Publication.
6	Research Project	Experienced	46	69	Extensive Additional Documentation in an associated Master Dissertation; Associated Publication.
7	Industrial Project (Government)	Experienced	74	41	Contact with Model Creators; Domain is regulated by Federal Norms.
8	Industrial Project (Government)	Experienced	44	29	Contact with Model Creators; Domain is regulated by Federal Norms.
9	Government Interoperability Effort	Beginner	15	7	Additional Documentation in Project Report.

Our strategy for identifying anti-patterns across the sample of ontologies above was conducted as follows. For each of these cases, we started by simulating the model at hand using the approach described in the previous section. This process resulted in a number of model instances (automatically generated by the Alloy Analyzer) which, thus, represented the *possible models instances* of the ontology. We then contrasted the set of possible instances with the set of *intended instances* of the ontology, i.e., the set of model instances that represented intended state of affairs according the creators of the ontology. When a mismatch between these two sets was detected, we analyzed the ontology representation in order to identify which structures in the model were the causes of such a mismatch. Finally, we catalogued as anti-patterns those model structures that recurrently produced such mismatches, i.e., modeling patterns that would repeatedly produce model instances which were not intended ones. To be more precise, we considered as anti-patterns the error prone modeling decisions which occurred in at least one third of the validated ontologies.

In order to detect the mismatch between possible and intended models, we carried out the simulation-based validation process with a constant interaction with the model creators (when available). Moreover, when the model creators were not available, the mismatches were detected by inspecting the textual documentation accompanying the ontology.

4. Semantic Anti-Patterns

This section presents a set of Semantic Anti-Patterns elicited via the method discussed in the previous section. We discuss, for each presented anti-pattern, their configuration, undesired consequences and possible solutions. Moreover, for each anti-pattern we provide an example found in one of the subject ontologies. Notice that the purpose of this paper is not to judge whether the ontological categories used to classify the elements in the shown examples are the proper ones. Instead, our goal is to analyze the consequences given those modeling choices.

A summary of the occurrence of each of these anti-patterns in the ontologies discussed in section 2 is presented in Table 2 below.

Table 2. A summary of the number of occurrences (#) for the following anti-patterns in each of the investigated ontologies: Generic Cycle (GC), Relation Between Overlapping subtypes (RBOS), Relation Specialization (RS), Imprecise Abstraction (IA), Type-Reflexive Relationship (TRR), Pseudo-Anti-Rigid (PAR)

Ontology	#GC	#RBOS	#RS	#IA	#TRR	#PAR
1	1	1	0	1	0	0
2	1	1	1	3	0	0
3	3	2	0	1	0	0
4	9	1	3	3	4	1
5	2	2	11	3	3	0
6	2	0	2	2	0	2
7	8	3	0	3	0	0
8	2	4	1	0	0	0
9	2	0	2	1	2	1
Total	**30**	**14**	**20**	**17**	**9**	**4**
Percentage	**100%**	**77.78%**	**66.67%**	**88.89%**	**33.33%**	**33.33%**

4.1. *The "Generic Cycle" Anti-Pattern*

The most recurrent anti-pattern found in our investigations is what we have named here *The Generic Cycle*. This anti-pattern was identified 30 times, occurring in 100% of the simulated ontologies. As depicted in Figure 2(a), it corresponds basically to a sequence of classes connected by domain associations of any type, configuring a cycle. The issue here is the following. Conceptual models such as the one in Figure 2(a) are always defined at the type level. At that level, the associations in the model work as a typing mechanism that prescribe which type of entities can occur in an association path. However, by itself, the structure of the diagram cannot proscribe unintended links that can be established between instances of these types at the instance level. In particular, suppose a sequence of types $T_1T_2 \ldots T_nT_1$ connected in a cycle such as the one of Figure 2(a). Moreover, suppose an OCL-like navigational statement $x.T_2$ that allows us to move from an instance x of type T_1 to an instance of type T_2 in a model (i.e., to retrieve the instances of T_2 which are associated with x). For instance, in the model of Figure 2(b), if we have *h* as an instance of Heart, then the mapping *h.HeartCells* give us the collective of *HeartCells* associated with *h*. In general, $x.T_2 \ldots T_n$ will then give us the instance of T_n associated with x (instance of T_1). Now, under this configuration, we found to be very common that users want to express the following restriction: for an arbitrary instance x of T_1, we have that $x.T_n \subseteq x.T_2 \ldots T_n$.

This constraint should have been included, for example, in the Heart Electrophysiology Ontology depicted in Figure 2(b). There, the domain experts would like to reinforce that: for a generic instance *h* of Heart, *h.Atrium* = *h.HeartCells.AtriumContraction.Atrium*. Notice, that without this constraint, the model of Figure 2(b) is still syntactically valid. However, it has at least one unintended model instance, namely, one in which the collective of cells from someone's heart contracts the atrium of someone else's heart!

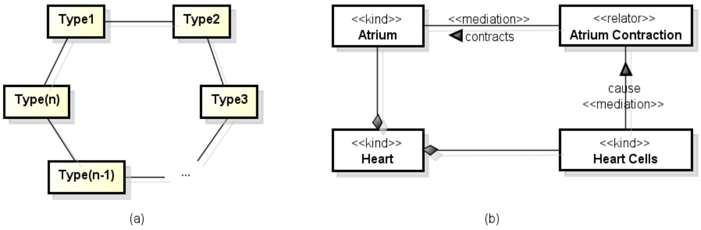


Figure 2. (a) Configuration of a Generic Cycle. **(b)** A generic cycle at the electrophysiology ontology.

4.2. *The "Relation Between Overlapping Subtypes" Anti-Pattern*

The following kind of cycle, due to its frequent recurrence (14 occurrences, in 77.78% of the ontologies), has been here categorized under a separate label. The "*Relation Between Overlapping Subtypes*" anti-pattern occurs when we have a relation R between overlapping (i.e., non-disjoint) subtypes $T_1 \ldots T_n$ of a common supertype T, as depicted in Figure 3(a). In this model, an instance of R will be an n-uple $<x_1 \ldots x_n>$ such that every x_i is an instance of type T_i (a subtype of T). Now, we have found that, typically, users want to include the following constraint for this model configuration: for a set of types $A = \{T_i \ldots T_j\}$, such that $A \subseteq T_1 \ldots T_n$, we have that the same instance x of T cannot appear in an n-uple of R instantiating more than one type in A.

An example of this anti-pattern was illustrated in Figure 1, although the material relation at hand derived from the Transplant relator is not made explicit there. Another example is shown in Figure 3(b), which was extracted from the Biodiversity Ontology. Here, according to the domain conceptualization, a Person can be the responsible for a Collection and a person can be an Assistant in a collection. However, the same person cannot play both roles for the same Collection.

The representation of this kind of constraint in the diagram is not possible, since the constraint must restrict the relation between instances, not between types.

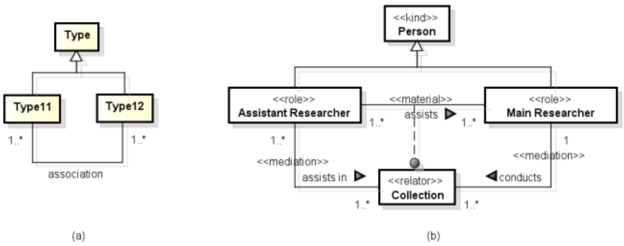


Figure 3. (a) Relation Between Overlapping Subtypes anti-pattern. **(b)** An example adapted from [13]

4.3. The "Relation Specialization" Anti-Pattern

Another particular case of a cycle is shown in Figure 4(a). This anti-pattern was identified in 66.67% of the ontologies in a total of 20 occurrences. The *Relation Specialization* is characterized by the representation of a relation R between two types T_1 and T_2 such that their respective supertypes ST_1 and ST_2 are also associated by a relation R_S. Now, we found to be commonly the case that users want an intended representation in which R (the relation between the specializing types) is a subset, specialization or redefinition [18] of the relation R_S between their supertypes.

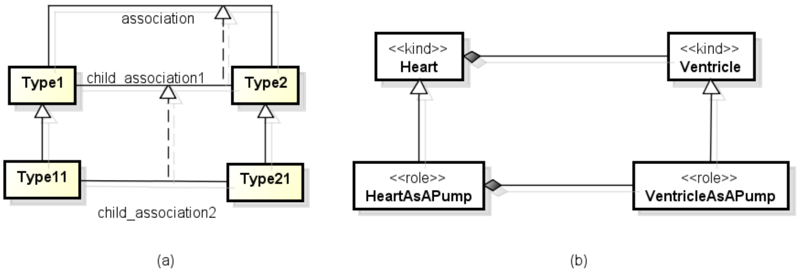


Figure 4. (a) The Relation Specialization anti-pattern. **(b)** An example found in the ECG Ontology.

An example of this anti-pattern is illustrated in the model of Figure 4 (b), adapted from the electrophysiology ontology. In this domain, Hearts are composed of Ventricles; moreover, when a Heart is working as a pump, it must be composed of ventricles that also work as pumps. However, as represented in Figure 4(b), this model does not proscribe a situation in which Heart H_1 is composed of ventricle V_1, Heart H_2 is composed of ventricle V_2, but when working as pumps, hearts H_1 and H_2 end up being composed of ventricles V_2 and V_1, respectively.

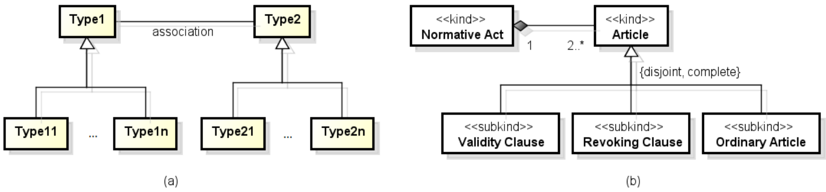
The solution for this anti-pattern is simple and works in all cases, namely, by explicitly representing the inclusion constraint between the two involved relations.

Standard UML 2.0 presents three modeling options for representing this inclusion constraint: association specialization, association subsetting and association redefinition. Moreover, as discussed in depth in [16], OntoUML presents clear guidelines for how each of these options should be chosen in particular modeling situations.

4.4. The "Imprecise Abstraction" Anti-Pattern

In the sample set of ontologies analyzed, the so-called *Imprecise Abstraction* anti-pattern was found in 88.89% of the cases in a total of 17 different occurrences. The generic configuration of this anti-pattern is depicted in Figure 5(a). In this configuration, we have two types T_1 and T_2, of which at least one (e.g., T_2) has subtypes ($T_{21} \ldots T_{2n}$). Moreover, T_1 and T_2 are related by association R. The source of the inconsistency resides in the representation of one single association R between T1 and T_2, thus, abstracting the more concrete associations between the (implicit) subtypes of T_1 ($T_{11} \ldots T_{1n}$) and (the respective subtypes of) T_2. This form of abstraction often leads to simpler and more concise models. However, it can also decrease the level of precision of these models. This occurs, for example, due to the fact that, in its more abstract version, the model can hide specific constraints (e.g., cardinality constraints) for the more specialized relations.

An occurrence of this anti-pattern (extracted from the Brazilian Normative acts ontology) is depicted in Figure 5(b). In this domain, each act is composed by Articles, which can have different purposes. These different sorts of Article are described by the *Subkinds* in the model of Figure 5(b), namely, Ordinary Article, Validity Clause, and Revoking Clause. The abstraction of all the compositions into a generic one indeed results in a cleaner model, because generally all the articles compose a normative act and a normative act is indeed composed by different types of articles. However, in that model, the composition of a Normative Act must follow a specific set of rules. For instance, every act must have exactly one Validity Clause and at most one Revoking Clause.



(a) (b)

Figure 5. (a) The Imprecise Abstraction anti-pattern. **(b)** Example found in the normative acts ontology

As an additional example, suppose a Genealogy ontology in which an offspring is said to have a maximum of two parents. Now, we have the subtypes Father and Mother as specialization of Parent. Although someone can have two parents, it is not the case that (in the assumed conceptualization) someone can have two Fathers or two Mothers.

4.5. The "Type-Reflexive Relationship" Anti-Pattern

The last of the particular types of cycled anti-patterns is called *Type-Reflexive Relationship*. It was identified 9 times, in 33.33% of the validated ontologies. The simplest of all, this anti-pattern consists in one concept and one association which association ends are both connected to the same type. It holds much resemblance to the *Relation Between Overlapping Subtypes* anti-pattern, since it implicitly represents two different roles of the concept, one for each end. The unintended consequences that arise from modeling a type-reflexive relationship are typically due to the missing definition of formal associations meta-properties (e.g., (in, anti) transitivity, symmetry (asymmetry, anti-symmetry), and (ir) reflexivity) that would explicitly restrict the intended semantics, eliminating interpretation problems.

Examples of this anti-pattern are found in the Biodiversity ontology which include a relation of *is-spatially-contained-in* defined between geographical regions and a relation of *component-of* defined between Ecosystems. In both cases, these associations are defined between two instances of the same type (Geographical Region and Ecosystem, respectively). However, in both cases, the two related instances cannot be same. Actually, both associations should be defined as partial order relationships (irreflexive, asymmetric and transitive). Without the explicitly representation of these meta-properties, one cannot guarantee that the intended semantics of these relationships will be preserved across different interpreters (humans and machines alike).

It is important to highlight that the occurrence of this anti-pattern can also manifest an occurrence of the *Imprecise Abstraction* anti-pattern. This can be the case if the type participating in the type-reflexive relationship R has subtypes and, hence, possible hidden constraints on how the instances of these subtypes can be related via R.

4.6. The "Pseudo-Anti-rigid" Anti-Pattern

In approaches such as OntoUML and OntoClean, a type T is considered as anti-rigid iff for any individual x that instantiates that type in any given world w, there exist at least one alternative world w' in which x does not instantiate T. Examples of anti-rigid ontological categories in OntoUML are *phases* (e.g., Living Person, Teenager), *roles* (e.g., Student, Husband, Professor) and *role mixins* (e.g., Customer, Voting Member) [3]. The *Pseudo-Anti-rigid* pattern (identified in 33.33% of the ontologies, with a total of 4 occurrences) does not have a distinct structural configuration. Instead, it is characterized by the presence of a type T which is defined as anti-rigid but, due to other constraints in the model, it turns out that the instances of T cannot cease to instantiate it in any possible world. In other words, although represented as anti-rigid, in this situation, T would logically be a rigid type.

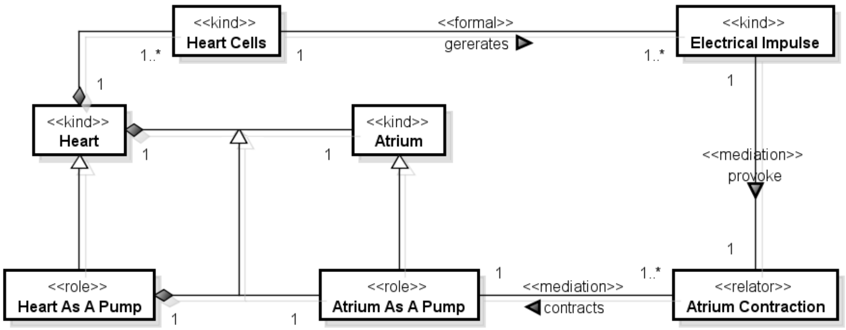


Figure 6. Adapted fraction of the electrocardiography ontology, described in [14]

Adapted from [14], in Figure 6 we have an example of this anti-pattern. In this domain, Hearts are composed of Atriums and Heart Cells. The Heart Cells collective generate Electrical Impulses, which contract the Atriums, thus, making them pump blood. The contraction of the Atriums makes the entire Heart also work as a pump.

In this model fragment, there are two classes stereotyped as roles, which were supposed to be anti-rigid. However, the simulation of this model shows that all Hearts and all Atriums must necessarily works as pumps, contradicting their anti-rigid meta-property. To show that, consider an instance α of the class Heart. We have that α is composed of collectives of HeartCells, which can generate (at least one) Electrical Impulse; each Electrical Impulse is responsible for precisely one Atrium Contraction; each Atrium Contraction necessarily contracts exactly one Atrium which, in turn, composes exactly one Heart. Since the latter heart must be exactly that heart α, we have that α's Atrium is always subject to contraction and, as a consequence, both α´s Atrium and α must always play the role of pump! In other words, α´s Atrium plays the role of a pump when mediated by an Atrium Contraction. However, following the constraints in Figure 6, this atrium is always subject to at least one contraction. Moreover, α plays the role of pump when composed by an Atrium working as a pump, which as just argued is necessarily the case. Thus, we have that the distinction between Heart (as a *Kind*) and *Heart as a Pump* (as a *Role* played by a heart) is non-informative. The same can be said for the distinction between *Atrium* and *Atrium as a Pump*.

Let us assume that the underlying intention is to express that *Heart as a Pump* and *Atrium as a Pump* are indeed anti-rigid, i.e., both hearts and atriums can exist without playing these respective roles. This can be made possible in this model by relaxing the appropriate cardinality constraints. For example, the relation between Heart Cells and Electrical Impulses could be modified to allow Heart Cells to exist without causing Electrical Impulses.

5. Final Considerations

The construction of large-scale reference ontologies is a complex engineering activity. To develop high quality models, an ontologist must be supported by expressive engineering tools such as theoretically well-founded modeling languages and methodologies, ontological patterns and computational environments.

This paper makes a contribution in this direction by presenting a number of empirically elicited Semantic Anti-patterns that were identified as recurrent in a pool of ontologies ranging from academic exercises to real-world reference ontologies. We believe that the process of anti-patterns identification and proactive fixing results in more precise models, which avoid implicitly representing unintended model instances.

The approach of anti-pattern elicitation via visual simulation conducted here has been proven to be fruitful and promising. We intend to pursue this approach with a larger and more diverse pool of reference ontologies as we believe that the continued assessment of these models via this approach can elicit additional semantic anti-patterns. Moreover, once a catalog of these patterns is identified, we intend to incorporate them in the OntoUML editor as an attempt to proactively detect and avoid their occurrence in particular OntoUML models.

Acknowledgments. This research is supported by FAPES (PRONEX grant #52272362) and CNPq (productivity grant #311578/2011-0).

References

[1] G. Guizzardi, Theoretical Foundations and Engineering Tools for Building Ontologies as Reference Conceptual Models, *Semantic Web Journal* **1** (2010), 3-10.

[2] R. Carraretto, *A Modeling Infrastructure for OntoUML*, Technical Report, Ontology and Conceptual Modeling Research Group (NEMO), Federal University of Espírito Santo, Vitória, Brazil, 2010.

[3] G. Guizzardi, *Ontological Foundations for Structural Conceptual Models*, PhD Thesis, Telematica Instituut Fundamental Research Series, Enschede, 2005.

[4] G. Guizzardi, M. Lopes, F. Baião and R. Falbo, The Role of Foundational Ontologies for Domain Ontology Engineering: An Industrial Case Study in the Domain of Oil and Gas Exploration and Production, International Journal of Information Systems Modeling and Design, 1(2010), 1947-8186.

[5] F. Carolo and L. Burlamaqui, Improving Web Content Management with Semantic Technologies, *Semantic Technology Conference (SemTech)*, San Francisco, 2011.

[6] Object Management Group, Semantic Information Model Federation (SIMF): Candidates and Gaps, Available: http://www.omgwiki.org/architecture-ecosystem/. [Accessed June 25th 2012].

[7] U.S. Department of Defense, Data Modeling Guide (DMG) for an Enterprise Logical Data Model, Available: http://www.omgwiki.org/architecture-ecosystem/lib/exe/fetch.php?media=dmg_for_enterprise_ldm_v2_3.pdf. [Accessed June, 25th, 2012].

[8] A. B. Benevides and G. Guizzardi, Model-Based Tool for Conceptual Modeling and Domain Ontology Engineering in OntoUML, *Enterprise Information Systems* **24**, Berlin, Springer, 2009, pp. 528-538.

[9] A. B. Benevides, G. Guizzardi, B. Braga and J. Almeida, Validating Modal Aspects of OntoUML Conceptual Models Using Automatically Generated Visual World, *Journal of Universal Computer Science* **16** (2010), 2904-2933.

[10] E. Gamma, R. Helm, R. Johnson and J. Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*, Addison-Wesley, United States, 1995.

[11] Ontology Design Patterns, Available: http://ontologydesignpatterns.org/wiki/Main_Page. [Accessed 25th June 2012].

[12] A. Koenig, "Patterns and Antipatterns" Journal of Object-Oriented Programming 8 (1995), 46-48.

[13] D. Jackson, *Sofware Abstractions: Logic, Language an Analysis*, The MIT Press, Massachusetts, 2006.

[14] P. P. F. Barcelos, G. Guizzardi and A. M. Garcia, Ontological Evaluation of the ITU-T Recommendation G.805, in 1*8th International Conference on Telecommunications*, Cyprus, 2011.

[15] A. C. F. Albuquerque, (Development of a Biodiversity Domain Ontology (*Desenvolvimento de uma Ontologia de Domínio para Modelagem de Biodiversidade – in portuguese*), MSc Thesis, Federal University of Amazonas, Vitória, 2011.

[16] B. Gonçalves, G. Guizzardi and J. G. Pereira Filho, Using an ECG Reference Ontology for Semantic Interoperability of ECG Data, *Journal of Biomedical Informatics* **44** (2011), 126-136.

[17] C. A. M. Bastos, A. Salles, L. S. d. Rezende, S. Mecena, M. L. Sanchez and M. A. F. Caldas, Building up a Model for Management Information and Knowledge: the Case-Study for a Brazilian Regulatory Agency, in *2nd International Workshop on Software Knowledge (SKY 2011)*, Paris, 2011.

[18] Brazilian Ministry of Planning, Budget and Management, Conceptual Modeling Outline for the Brazilian Government Organizational Structures and the SIORG (*Esboço de Modelagem Conceitual para Estruturas Organizacionais Governamentais Brasileiras e o SIORG – in portuguese*), Available:http://vocab.e.gov.br/2011/09/org.pdf. [Accessed June 25th 2012]

[19] D. Costal, C. Gomez and G. Guizzardi, Formal Semantics and Ontological Analysis for Understanding Subsetting, Specialization and Redefinition of Associations in UML, in *30th International Conference on Conceptual Modeling (ER 2011)*, Brussels, 2011.