Mapping Structural Design Patterns in OWL to Ontological Background Models

Vojtěch Svátek, Miroslav Vacura
University of Economics
W. Churchill Sq. 4, 13067 Prague, Czech Rep.

Martin Homola, Ján Kľuka
Comenius University
Mlynská dolina, 84248 Bratislava, Slovakia

ABSTRACT

The concerns of efficient data management and logical inference on the Semantic Web often lead to disconnection between the surface structure of RDF/OWL data/ontologies and the background state of affairs. The PURO ontology background model language allows to explicitly capture the mapping between the foreground and background modeling layers. The background modeling primitives are intentionally kept analogous to those of RDF/OWL, namely, derived from the particular-universal and relationship-object distinctions. We project the PURO framework onto the W3C CPV family of structural design patterns, thus providing additional insights into them and possibly facilitating their selection and reuse.

1. INTRODUCTION

With the growing popularity of Semantic Web technology, a large portion of new ontologies, such as Linked Data (LD) vocabularies, has been directly authored in OWL [5], and thus influenced by its limited expressiveness and efficient reasoning requirements, but also by pre-existing data models and application needs. Adherence to the deeper ontological nature of the modeled entities is not necessarily observed; for instance, universal entities (types) or complex relationships are often syntactically represented by individuals (objects). When such models are visualized, reused, matched or transformed, such modeling discrepancies may become troublesome.

Consider the diagram on the left-hand side of Fig. 1, depicting the complex fact of a business entity (resource 3) offering exemplars (i.e., 'some items') of a certain musical album (resource 1) as product for sale, in a certain region. The fact refers to two LD vocabularies: the e-commerce ontology GoodRelations (GR)[1] and the Music Ontology (MO).[2] The remaining two instance-level resources in the diagram (2 and 4) are the 'offering' itself and the value '90' (minutes) understood as 'typical' and thus modeled as a resource rather than literal.[3] In the right-hand side diagram we approximate the *ontological background* of this fragment (omitting the entities that would be *types* in both diagrams, for easier readability). Among other things we see that the object property mo:release_type becomes an additional type of the product offered and that the 'offering' is now a relationship (with arity >2) rather than a self-standing object.

[1]http://purl.org/goodrelations/v1

[2]http://purl.org/ontology/mo/

[3]Such kind of modeling is not common in MO, but, rather, in GR-compliant ontologies, cf. http://www.ebusiness-unibw.org/ontologies/opdm/#ontologies.

K-CAP '13, June 23-26, 2013, Banff, Canada.

Obviously, modeling the ontological background for each individual data fragment is unfeasible. The mapping thus has to be established at the level of entity *types*, which means, indirectly. On the one side of the mapping is an *ontological foreground model* (OFM), i.e., the structure of an RDFS/OWL ontology; on the other side is an analogous *ontological background model* (OBM). OBM models should be represented in a suitable *OBM language* of modeling primitives (OBML). Examples of such languages include OntoClean [3] and PURO [8], the latter employed in this paper.

The aim of *pattern-based* ontology design is to build ontologies more rapidly, avoid common pitfalls and achieve better interoperability. Available pattern collections include: (1) a W3C one;[4] (2) the Rome *OntologyDesignPatterns.org* portal; (3) the Manchester ODP catalog.[5] All patterns from the W3C and Manchester collections, and also some from the Rome portal, are so called *structural patterns*, concerned with efficient modeling of a generic subject given the expressiveness and reasoning capabilities of OWL.

Similarly to actual ontology design, the developers of an ontology pattern have to decide which pattern element is to be modeled by which particular construct. Given that the patterns have to be finally instantiated in some ontology language with limited expressiveness and reasoning trade-offs, the patterns themselves may intentionally diverge from the strict ontological nature of things. This divergence may not be apparent to the pattern's user, as it is not typically discussed when patterns are defined. Also, multiple patterns may cover same modeling problem, and the ontology developer can choose one of them according to her needs and circumstances; the W3C's 'Classes as Property Values' (CPV) document [4] analyzed in this paper offers a family of five distinct patterns for seemingly one modeling problem. Hence, even following the best practices in ontology design (here, the applicable patterns) may lead to ontologies covering similar domains, but diverging from the ontological background in various structurally incompatible ways—causing problems, e.g., in data integration.

In this paper, we use the PURO OBML to annotate and analyze the five W3C's CPV patterns. By making the OBM explicit, we are able to introduce three additional patterns covering the same modeling problems and to evaluate the suitability of each pattern depending on the application. The paper thus brings new insights to the problem of modeling classes as property values and respec-

[4]http://www.w3.org/2001/sw/BestPractices/

[5]http://www.gong.manchester.ac.uk/odp/html/

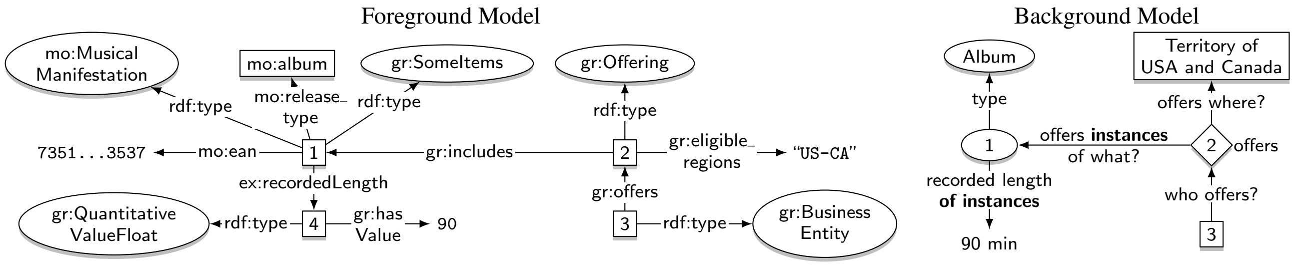

Figure 1: OFM and OBM of the same example data

tive ontology patterns, while in the same time it demonstrates the applicability of the PURO OBML in ontology pattern design.

2. THE PURO OBML

PURO ontological background models [8] make two basic distinctions: between ontological *particulars* and *universals*, and between *objects* and *relationships*. Particular and universals are distinguished by the possibility of instantiation. As for the R-O distinction, objects are entities with their own identity, while a relationship cannot be considered without the entities on which it depends. Furthermore, some ontologies (especially LD vocabularies) model assignments of quantitative data values to individuals. Such assignments are not proper relationships: we distinguish them as a third option within the R-O distinction: *valuation*. The P-U and R-O(-V) distinctions are orthogonal, thus creating a two-dimensional space of the size 2×3, see Fig. 2, where each PURO construct is associated with the OFM entity (in parentheses) through which it can be represented most naturally.

	***O**bject*	***R**elationship*	*Valuation*
***P**articular*	$\mathcal{B}$-object (individual)	$\mathcal{B}$-relationship (*3 possibilities*)	$\mathcal{B}$-valuation (data prop. assert.)
***U**niversal*	$\mathcal{B}$-type (class)	$\mathcal{B}$-relation (*3 possibilities*)	$\mathcal{B}$-attribute (data property)

Figure 2: Basic constructs of the PURO OBML

Higher-order types and *n*-ary relationships cannot be directly represented in OWL (they are typically meta-modeled), however in PURO we can correctly annotate them. The notion of $\mathcal{B}$-relationship is further decomposed into three variants, *$\mathcal{B}$-fact*, *$\mathcal{B}$-instantiation* and *$\mathcal{B}$-axiom*, whose analogous OFM entities are object property assertion, instantiation (rdf:type statement) and T-box axiom (such as subclassing, equivalence or disjointness). Again, the analogy with RDF/OWL (DL) is imperfect, since in the OBM, $\mathcal{B}$-types can be not only declared as instances of (higher-order) $\mathcal{B}$-types but also appear on *one* side of a $\mathcal{B}$-fact. The ontological entities from the example in Fig. 1 would then be labeled as follows:

- the class gr:BusinessEntity as a 1st order $\mathcal{B}$-type (i.e., instantiated by $\mathcal{B}$-objects);
- the mo:release_type property as $\mathcal{B}$-instantiation;
- the class mo:MusicalManifestation and the individual mo:album as 2nd order $\mathcal{B}$-types (instantiated by 1st order $\mathcal{B}$-types);
- Class gr:Offering together with property gr:offers as a (ternary) $\mathcal{B}$-relation instantiated by $\mathcal{B}$-facts;
- the ex:recordedLength and gr:hasValue properties and the gr:QuantitativeValueFloat class as part of $\mathcal{B}$-attribute;
- gr:eligibleRegions as a property valued by $\mathcal{B}$-objects;
- the mo:ean property has no OBM, as it is valued by *identifiers* of individual mo:MusicalManifestation $\mathcal{B}$-types.

Note, in particular, that the mo:album individual 'jumps two orders up' when moving to the OBM view.

PURO was conceived with respect to good mental alignment with OWL modeling primitives and is intended especially for use with LD vocabularies. One may be concerned whether such distinctions can be reliably distinguished in real LD vocabularies. To answer this question, we have performed an annotation experiment on three LD vocabularies [7] where we found out that two annotators acting on common guidelines could reliably agree on these distinctions. Illustrating the issue on the P-U axis, we indeed found out that LD vocabularies (for business, government, geography, private life, etc.) anticipate facts about concrete objects (persons, organizations, items of goods, documents, or the like, often even with some kind of legal status) which can be reliably distinguished from the universals in the domain. Problems would definitely arise in biomedical,[6] or, more generally, scientifically-biased ontologies. These are almost exclusively concept-oriented, although in some cases, individuals (e.g., cells, organs, or chemical processes taking place at some time at some location) acting as 'prototypes' are used, thus blurring the P-U distinction. We remark, however, that PURO model is specifically designed for LD vocabularies, hence these ontologies are not its primary application targets. For more details please refer to [8, 7].

3. ANALYSIS OF THE CPV PATTERN

While we analyzed all structural patterns from the three collections, we only (for lack of space) present the results for the most interesting one; for others see the long version of this paper.[7]

The W3C note [4] provides and exemplifies five modeling options for the fact that an individual (book) has a particular relationship ('having as subject') to an entity (animal taxon, such as Lion) represented by a class embedded in a class hierarchy. Diagrams from [4] are reproduced for reader's convenience in Fig. 3.

Approach 1: Use classes directly as property values. The pattern is suitable for expressing a $\mathcal{B}$-fact between an individual Book ($\mathcal{B}$-object) and a taxon itself ($\mathcal{B}$-type), i.e., when the book deals with the *intension* of the taxon class (e.g., explains by whom or when the Lion taxon was defined). In OWL 2, this can be expressed within the DL fragment, thanks to punning.[8] Using Approach 1 in connection with taxon class *extension* seems less plausible.

Approach 2: Create special instances of the class to be used as property values. Here, again, the relationship behind the dc:subject property corresponds to a $\mathcal{B}$-fact; however, there are $\mathcal{B}$-objects on both sides, the second one representing a topic (such as LionSubject). The original target classes are however redefined by this step (and should also better be renamed as well)—as they

[6] See, e.g., http://bioportal.bioontology.org/.
[7] http://patomat.vse.cz/tr-sdp-2013.pdf
[8] http://www.w3.org/TR/owl2-new-features/

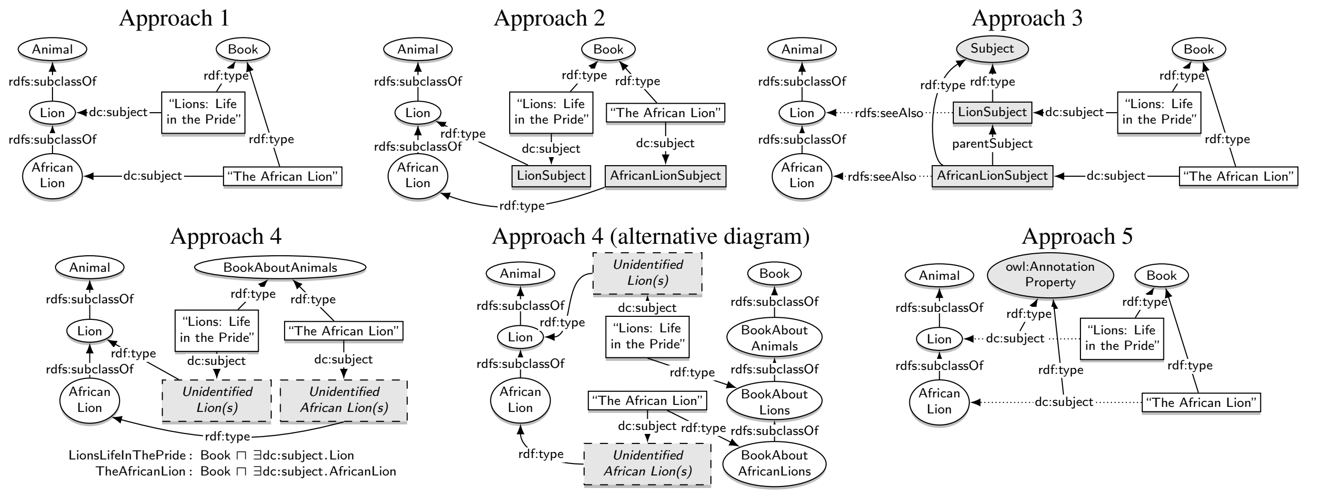

Figure 3: Diagrams of CPV patterns, reproduced from [4] except the alternative diagram for Approach 4

now have topics as instances, they cannot have physical animals at the same time. The original class hierarchy populated by physical animals actually stays apart of the presented pattern fragment.

Approach 3: Create a parallel hierarchy of instances as property values. The background nature of the dc:subject property is the same as in Approach 2. The original class hierarchy is present in the pattern, but it is conceptually disconnected, since the 'animal topic' has no clear 'factual' relationship to actual animals or their class. Also note that a parentSubject property assertion (expressing that a topic is a part of another topic) is partonomic, thus *not* conceptually equivalent to an rdfs:subclassOf relationship.

Approach 4: Create a special restriction in lieu of using a specific value. The dc:subject property links a Book to an (undetermined) physical animal, which is an instance of the target class. The link is expressed indirectly via OWL DL existential restriction. Subclasses of Book for various topics can also be defined explicitly, see the alternative diagram. The OFM intuitively corresponds to the OBM: a $\mathcal{B}$-fact (reflected by a dc:subject assertion) plus a $\mathcal{B}$-instantiation (reflected by a rdf:type assertion). This is analogous to the 'offering instances of an album' situation in Fig. 1.

Approach 5: Use classes directly as annotation property values. This approach is conceptually equivalent to Approach 1. However, one might argue that annotation properties should not be used for conceptual relationships but only for 'technical' purposes.

Let us now add three more approaches (see Fig. 4) not present in [4] but straightforwardly derivable from PURO-based insights:

Meta-Modeling Approach. A taxonomy of individuals meta-modeling[9] the classes themselves (rather than topics, as in Appr. 3) can be built, with a dedicated property, e.g., subtypeOf, connecting sub- and super-types. In the OBM, assertions of this property become $\mathcal{B}$-axioms and blend in with the original subclass hierarchy. This approach allows to express the relationship between an individual and the class it meta-models directly, using an auxiliary property (say, models). Compared to Appr. 5, which uses annotation properties, this relationship is within the sight of a reasoner.

SomeItems Approach. We can borrow the SomeItems class from GR, cf. Fig. 1. Each of its instances is a placeholder for an unspecified amount of real-world objects (items of goods) of a certain class (product or service model) that are offered by a business entity under certain conditions. In the CPV setting this leads to a foreground model that explicitly expresses that the source individual relates to the extension of the target class (e.g., the book is about physical lions), while OWL DL expressiveness is not needed, in contrast to Approach 4. The downside of this kind of modeling is that placeholders representing multiple entities (animals) may become mixed with normal entities in the extension of the class.

Fact-Instantiation Label Approach. This slightly differs from Approach 1 by explicitly revealing the relatedness to the *extension* of the target class. We could do it by linking some specially designed annotation property on the dc:subject assertion, or alternatively (as in Fig. 4) by introducing a new object property such as subjectInstancesOf (to express that the subject of the book is not the class itself but its instances—analogously to the 'offers instances of what?' label in the OBM in Fig. 1).

All eight approaches are summarized in Table 1, in the same order as above (using obvious acronyms for the three last ones). Columns 2–4 assess the suitability of each approach to capture the background relationship of the source individual to different aspects of the target class: an abstract topic[10] derived from the class ('Topic'); its intension ('Int.'); and its extension ('Ext.'). The last column picks up the most important issues identified either by the authors of [4] or by us. We immediately see that the CPV approaches fall into three clusters: those focused on topic modeling (2, 3), those indicating relationship to class intension (1, 5, and MM), and those indicating relationship to class extension (4, SI, and FIL). Within the clusters, the difference lays in presence of various side-effects.

4. RELATED RESEARCH

Although the divergence between ontological foreground and background has been observed before, the amount of literature addressing its practical impact is small. A prominent example is OntoClean [3], which, however, is more concept-oriented (in contrast, PURO can be seen as fact-oriented), and requires the engineer to take into account OBML notions (such as rigidity) that are fundamentally different from what s/he knows from the OFM representation. The ontological distinctions recognized by PURO are also found in many foundational ontologies (FO) such as DOLCE[11] or BFO,[12] which are, however, combined with domain ontologies in

[9]Cf. [2] where such meta-modeling is used for simulating TBox reasoning via cheaper ABox level operations.

[10]That is, without bias towards the intension/extension of a class.

[11]http://www.loa.istc.cnr.it/DOLCE.html

[12]http://www.ifomis.org/bfo

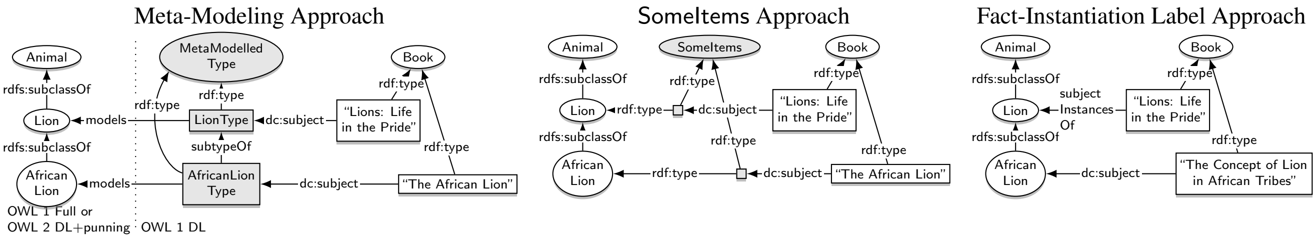

Figure 4: Diagrams of additional CPV patterns

Appr.	Topic	Int.	Ext.	Issues
1	+−	+	+−	OWL Full or punning
2	+	+−	−	Topics mutually unlinked; mixing topics and real individuals
3	+	+−	−	Topics loosely linked to classes
4	−	−	+	OWL DL (use of restriction)
5	+−	+	+−	Use of annotation properties for modeling content
MM	+−	+	+−	Redundant representation (classes and individuals with same meaning)
SI	−	−	+	Mixing placeholders and real individuals
FIL	−	−	+	Meaning fully depends on additional annotation or naming convention

Table 1: Summary of CPV approaches from the background model perspective

the same modeling layer, while the OBML paradigm keeps the OBM structures distinct from the OFM ones; rather than 'injecting' the OBM into the OFM (as for the FO), its constructs are unobtrusively linked to the given ontology via annotations. The ontology patterns that we analyzed were also evaluated by other researchers [1, 6]; they, however, predominantly focused on syntactic and inferential issues, rather than on the ontological background. Regarding other studies of the CPV problem, [9] predates [4] and partially overlaps with it, suggesting five approaches specifically for modeling subjects (with similar, though not identical meaning as the notion of 'topic' used in this paper).

5. CONCLUSIONS AND FUTURE WORK

The notion of ontological background models (OBMs) has until now been implicit in ontological engineering research. In this paper, we have used the PURO OBML [8] to analyze existing W3C 'Classes as Property Values' patterns (analysis of further patterns is in a supplementary material). We have demonstrated that PURO-based analysis brings new insights into OWL structural patterns.

Ongoing work concerns systematic discovery of PURO OBMs behind LD vocabularies. We have already built nearly-complete *PURO annotation sets* of the MO, GR, and FOAF vocabularies, in the form of annotation of individual foreground entities (in particular, classes, individuals, and property ranges) with PURO primitives. This effort is being carried out manually (by expert ontologists), but with the help of an *annotator tool*—a Protégé plugin allowing to create annotations for each vocabulary. A vocabulary entity can be annotated either generically (typically, based on its textual description), or with respect to its use in a *particular dataset*. Rather than being solely textual labels, annotations refer to a PURO (meta-)ontology, which, with the help of meta-modeling allows to automatically check for coherence of the OBM using a conventional reasoner [8] (similar approach is known for OntoClean [2]).

What is needed is more thorough analysis of the *interactions* between entity-level annotations, which could assist the annotation process by suggesting the possible/probable labels. In the 'pattern' thread, the next step will be comparison of the PURO OBMs of vocabularies with the OBMs of structural patterns, leading to the design of new patterns for problems not covered yet.

Acknowledgements. V. Svátek and M. Vacura are partially supported by the EU ICT FP7 under No.257943, LOD2 project. M. Homola and J. Kľuka are supported from the Slovak national VEGA project no. 1/1333/12. This collaboration resulted from the Czecho-Slovak bilateral mobility project registered under no. SK-CZ-0208-11 and no. 7AMB12SK020.

6. REFERENCES

[1] Dodds, L., Davis, I.: *Linked Data Patterns*. Online: `http://patterns.dataincubator.org/book/`

[2] Glimm, B., Rudolph, S., Völker, J.: Integrated metamodeling and diagnosis in OWL 2. In: Proc. ISWC, 2010.

[3] Guarino, N., Welty, C.: An Overview of OntoClean. In: *Handbook on Ontologies*, pp. 151–172. Springer, 2009.

[4] Noy, N. (ed.): *Representing Classes As Property Values on the Semantic Web.* W3C Working Group Note 5 April 2005. `http://www.w3.org/TR/swbp-classes-as-values/`

[5] *OWL 2 Web Ontology Language Document Overview.* W3C Recommendation, 2009.

[6] Rodriguez-Castro, B., Ge, M., Hepp, M.: Alignment of Ontology Design Patterns: Class As Property Value, Value Partition and Normalisation. In: Proc. ODBASE'12.

[7] Svátek, V., Homola, M., Kľuka, J., Vacura, M.: Ontological Distinctions for Linked Data Vocabularies. Technical Report, online `http://patomat.vse.cz/tr-odlv-2013.pdf`.

[8] Svátek, V., Homola, M., Kľuka, J., Vacura, M.: Metamodeling-Based Coherence Checking of OWL Vocabulary Background Models. In: Proc. OWLED'13.

[9] Welty, C.: *The Ontological Nature of Subject Taxonomies.* In: Proc. FOIS, 1998.