Chapter 9

Collected Research Questions Concerning Ontology Design Patterns

Karl Hammar, Jönköping University and Linköping University, Sweden

Eva Blomqvist, Linköping University, Sweden

David Carral, Wright State University, Dayton, Ohio, USA

Marieke van Erp, Vrije Universiteit Amsterdam, The Netherlands

Antske Fokkens, Vrije Universiteit Amsterdam, The Netherlands

Aldo Gangemi, CNR-ISTC, Italy

Willem Robert van Hage, Vrije Universiteit Amsterdam, The Netherlands

Pascal Hitzler, Wright State University, Dayton, Ohio, USA

Krzysztof Janowicz, University of California, Santa Barbara, USA

Nazifa Karima, Wright State University, Dayton, Ohio, USA

Adila Krisnadhi, Wright State University, Dayton, Ohio, USA and Universitas Indonesia

Tom Narock, Marymount University, USA

Roxane Segers, Vrije Universiteit Amsterdam, The Netherlands

Monika Solanki, University of Oxford, U.K.

Vojtech Svatek, University of Economics, Prague, Czech Republic

9.1. Introduction

This chapter lists and discusses open challenges for the ontology design pattern (ODP) community in the coming years, both in terms of research questions that will need to be answered, and in terms of tooling and infrastructure that will need to be developed to increase adoption of ODPs in academia and industry. The chapter is organised into four sections: Section 9.2 focuses on issues pertaining to the patterns themselves, including understanding their features and qualities, and developing pattern languages and standards. Section 9.3 concerns the evaluation and development of methods for using, constructing, and extracting ODPs. Section 9.4 focuses on tooling and infrastructure development, including Ontology Engineering environments that support pattern use, pattern repository development, and sustainability and versioning issues. Finally, Section 9.5 lists some additional research challenges related to pattern-based ontology modularization.

9.2. Patterns and Standards

9.2.1. Quality and Evaluation

Many ontologies are rarely, if ever, reused by third parties. This flies in the face of one of the original main motivations for creating ontologies, namely as shared conceptualizations. However, the exact reasons why ontologies see so little reuse are at this point very poorly understood. Anecdotal evidence indicates that ontologies often are poorly designed and constructed, difficult to understand, insufficiently documented and maintained, too specific or too broad (or both). Social dynamics and findability may also play a role. As ontology design patterns are created and used, it behooves the community to really understand the causes for the lack of reuse.

Without quality documentation, it is nearly impossible for ontology engineers to discover existing reusable ODPs, whether by browsing manually or using some search tool. Thus understanding how to best document ODPs (which documentation components are needed, in which order should they be displayed, using which graphical layout, etc) and developing benchmarks for evaluating ODP documentation are important research directions.

Further, the ODP community needs to address possible causes related to poor design, to understandability issues, and to finding an appropriate balance for specificity versus generality. A thorough and evidence-supported understanding of these issues, e.g. what exactly “poor design” and “good design” are, appears to be necessary to advance on these fronts; investigations into ontological anti-patterns fall into this realm [19, 20]. In particular, there seems to be a lack of user-centric evaluations addressing the ontology reuse issue, with only a few notable exceptions, e.g. [10].

In addition to the above, even if a greater understanding of ODP design and documentation are developed, we still need to challenge and evaluate our basic assumptions. We often take as a given that properly supported, ODP use will foster reuse of ontologies, help clean up messy data and facilitate data integration, when in fact, these claims are to a large degree still unverified hypotheses. An important step forward for the ODP community would be to show the validity of such claims. This could be done via empirical studies such as [10] and the development and proper documentation of useful applications which make use of the ODP paradigm.

9.2.2. Pattern Languages and Standards

Currently, ontology design patterns are mostly presented in the form of OWL files, with some minimalistic accompanying documentation. While OWL can indeed be used to express the core of a pattern, it does not provide native ways to represent additional information which would be helpful for reuse or organization of patterns.

For example, it would seem to be important to understand in depth (and be able to represent formally) how different patterns relate to each other. This is important to understand compositionality aspects, e.g. how to create an ontology based on existing ontology design patterns. Key relations are specialization, generalization, and composition and are discussed as in [17], but often relationships are more complex. For example, there are notions of views or shortcuts, see [9, 12], which can be understood as a type of temporary simplification of a pattern.

Using OWL to express axiomatizations provided with patterns is also limiting because the open-world assumption underlying OWL does not cater for expressing non-monotonic constructs like constraints, and because some patterns call for an axiomatization of, say, transitive closure or general rules not expressible in OWL DL. Some first steps towards developing such a representation language have already been undertaken [5, 7] but more work remains to be done.

Another challenge concerns naming in ODPs – currently, the lexical aspect of patterns is rather ad hoc. There has already been some work specifically devoted to ontological naming conventions [21] and even mentioning cross-entity structures [23], but little long-term systematic work has been carried out, and the respective category of the ODP portal is, consequently, empty.[1] The naming aspect is important both for content patterns (which are a kind of mini-ontologies with their own vocabulary) and for logical patterns, where structural transformation from one modeling variant (e.g., reification or meta-modeling) may also involve transformation of naming such as generation of derived word forms of different parts of speech.

9.3. Method Development

9.3.1. Usage Methods

Currently there is only one published ontology engineering methodology that explicitly mentions the use of ODPs: the eXtreme Design (XD) methodology [1, 14, 15]. If we compare this situation with that in software engineering, we find that in that discipline, almost regardless of what methodology you use, you will take design patterns into account in some way or another. Following this example, in ontology engineering it would probably be beneficial to be able to use ODPs in any ontology engineering methodology, and not only in XD. This raises the question of what ODP usage would look like in other kinds of ontology

[1] `http://ontologydesignpatterns.org/wiki/Category:NamingOP`

engineering methodologies, and if different variations of the ODP concept may be needed there.

We also note that there are still open questions relating to the improvement and further development of the XD methodology. Improvement suggestions include providing ways of tailoring the methodology to different project settings such as the staffing situation, duration of the project, the overall goal of the project (e.g. type of ontology, how much focus is put on correct concept definitions and terminology vs. logical structure and reasoning capabilities), and also detailing existing steps of the methodology to better fit realistic project settings. The latter includes the question of reuse – how do you reuse an existing knowledge structure (potentially an existing ontology) and extend that by using ODPs? How do you know an ODP fits your existing structure, so that you know what ODPs to select for your extensions?

9.3.2. ODP Development

The advance of ontology design patterns is caught in a catch-22: In order to provide convincing evidence for the added value of ODPs, the community requires access to a well-organized, well-documented, and well-maintained set of interlinked high-quality ontology design patterns. At the same time, however, there is a lack of incentive (and funding) to provide these, as long as this added value has not yet been convincingly demonstrated.

A joint effort to create, document, and properly catalogue key ontology design patterns will be needed, as well as a discussion on which patterns are needed and how they should be provided. A pattern creation effort is ongoing for the geosciences domain, in form of the U.S. GeoVocamps [9], but the process is still very ad-hoc, and a broader organized effort will be needed.

Good, and highly visible, use cases can be powerful drivers of such community efforts. While use cases for pattern-driven ontology engineering exist (e.g., the recent [11, 13]), they need to be catalogued, made accessible, and assessed regarding the relevance of ODPs to the scenario. Potential additional use cases could be developed with focus on data publishing and reuse, e.g. in the realm of linked data [18], which has seen much work in recent years. A regular challenge or competition event at the WOP workshops could be set up to facilitate this development.

One recent area where the development of more ODPs could fill a clear need would be when applying some non-standard features in ontologies. There are many tutorials, courses, modelling guidelines and ODPs that can help when using standard OWL. However, every year new extensions to the standards are developed, e.g. in terms of managing and reasoning over streaming data, in terms of modelling and reasoning over uncertainty etc. Nevertheless, there are so far few ODPs for such “extensions” to standard OWL, although this is an area where help and support for the ontology engineer would be really badly needed.

Another challenge concerns methods and best practices for the development of new patterns, which is currently undertaken in a rather ad-hoc manner. Just as guidelines are available for publishing ontologies and linked data, best practices must be developed that guide pattern developers in a systematic way to address the problem of developing and publishing a design pattern. These guidelines must empower the pattern developers to ascertain when their pattern axiomatisation is indeed a pattern and when it moves into the realm of a full fledged ontology before they publish it. Best practices may include defining the scope of the pattern, the minimum number of use cases that need to be made available as examples of the pattern application, the level and kind of axiomatisation that is appropriate and the possible extension points for future exploitation of the pattern amongst others, etc. The inclusion of patterns in the ontologydesignpatterns.org patterns repository should be governed by validating these criteria.

In addition to traditional OWL-based ODPs, we should consider that ontology patterns (especially content/knowledge patterns) exist in more or less explicit forms in datasets and models developed within diverse disciplines beyond ontology engineering and the Semantic Web, including formal and computational linguistics, lexical resources, data modelling, stylesheets, microformats, web pages, HCI, tables, etc. Making these implicit patterns explicit, and finding links, is a big challenge, which we have just started. Cf. [6] for a related manifesto. In particular, the patterns emerging from NLP and linguistic resources, and those from ontology engineering and the Semantic Web have important relations, in terms of grounding, empirical evidence, cognitive relevance, etc. (cf. the NLP formal integration work enabled by FRED [16], and the correspondences found in FrameBase).

9.3.3. ODP Extraction

The development of Software Engineering design patterns has conventionally followed a bottom-up approach where recurring patterns are extracted from existing software. This also serves as a validation of the usage of the pattern. However, the development of ODPs has largely happened in a top-down manner. This was justified in the early days of ontology development due to a lack of critical mass. However, strategies are now required that encourage bottom-up development and enable automated extraction of patterns from a corpus of ontologies.

Such a corpus, that is as of yet relatively untapped, in fact exists: many well known and widely used ontologies were developed before ODPs were introduced. Some of these ontologies are upper level such as BFO and SUMO, while others such as SNOMED-CT and CIDOC-CRM are relatively large. These ontologies potentially encode a number of ODPs, a fact which is implicitly validated by their widespread usage. Analyzing and restructuring these ontologies to identify patterns using traditional manual methods is a non-trivial task requiring significant support, which is often not available. The development of automated extraction strategies and tools could however uncover many currently hidden ODPs, thereby bridging the gap between the demand and availability of ODPs, and overcoming the catch-22 discussed in Section 9.3.2.

9.4. Tooling and Infrastructure

9.4.1. Pattern-Capable Ontology IDE:s

Which kind of tool support that is needed and best suited for ODP development and use, is an open question. Some tools have been proposed (such as plugins for the NeOn toolkit, and more recently for WebProtégé [8]) but they do not cover all possible tasks where ODPs can be applied, and in particular there are currently no tools supporting the construction or extraction of patterns, and very few that support sharing, discussing and collecting them – while some online portals do exist, there are no actual client-side tools.

Also, the tooling that does exist is very traditional, introducing the ODPs as a small variation of the normal workflow. An interesting question is whether there are any opportunities for more disruptive changes in tooling that could be supported by ODPs. For instance, (content) ODPs are sometimes viewed as small puzzle pieces, but so far there have not been any attempts to create tools that truly operate on the level of ODPs as their primitives (rather than OWL languages primitives). One could imagine taking ontology engineering to an entirely new level, where the underlying logical language is hidden behind a pattern language.[2]

Additionally it is important for the modeler to become aware of the availability of a suitable pattern when trying to model a certain state of affairs – currently there is a lack of pattern discovery services in general, as observed in [2] as one of the ODP adoption barriers. While keyword search may be of some use for content patterns it is insufficient for logical-structural patterns. The proposed development of an expressive pattern representation language is important but does not suffice by itself if modeling situations are not described as well. Ideally, the retrieval of patterns should be closely interconnected with the support of its reuse. When the modeling challenge mainly consists of expressivity constraints of the target language (e.g., related to n-ary relationships or meta-typing), the modeled situation can possibly be described using first-order logic or a dedicated language such as PURO [22], and starting from this representation, the reuse of the pattern might be invoked semi-automatically for the target ontology [4]. If the problem is of more complex nature, some kind of interactive navigation over some "ontology of ontological modeling problems" to a node referencing adequate patterns might suit better, and the reuse itself is then likely to be more manual as well. In any case, such supportive technologies are currently either missing or immature.

9.4.2. Repository Development

Ontology patterns can be needed in many different lexical and structural contexts. An advanced semantic search capability would boost their usage, but e.g. linking competency questions to potentially useful patterns is not trivial, especially if the structure of the question has to be taken into account. There is room here to join in with research on question-answering on linked data.

[2]See also the discussion on pattern language in Section 9.2.2.

9.4.3. Sustainability Challenges

The provision, maintenance and documentation of ontology design patterns are currently done in a very ad-hoc manner. While the community has a rudimentary portal,[3] it lends only limited support and structure. Versioning aspects and other software engineering related issues need to be addressed, and better tool support for pattern-driven ontology engineering needs to be developed. Sustainability related research questions are about finding out how to best implement infrastructure and apply structured and qualitative software engineering principles to ODPs. What would a tool for the efficient support of ODP-driven ontology-engineering look like? How should documentation be provided? What does versioning mean for ontologies or ODPs on the Web?

9.5. Challenges Related to Pattern-Based Ontology Modularization

Usually, ontology design patterns are merely thought of as building blocks for traditional ontologies. However, it can be argued that there may also be added value in using ODPs for other than this very specific purpose.

One example would be the possible modularization of ontologies [11]. The idea, in this case, would be to somehow preserve the underlying pattern structure, in the form of connected modules, as part of the final ontology. The potential advantages of this appear to be rather self-evident: Modularization would simplify documentation, understanding, and thus reuse. At the same time, however, it would also make it easier to modify parts of an ontology, in particular if such a part be contained within a specific module. Furthermore, repurposing of an ontology may be much easier if modularized, because replacement of modules may be a more straightforward means of adapting an ontology to a new purpose.

Current ontology languages, however do not provide suitable means for representing such modular ontologies and their dependencies. There is also little experience in the ramifications arising from module updates, and of the role of axiomatisations with respect to modularization and module modification. We also do not know about concrete studies which would verify the more obvious promises of such modularization.

Explicit preservation of ODPs used for ontology construction (e.g., as modules) also have the potential to simplify ontology engineering tasks which usually come later in the development pipeline. For example, a systematic reuse of ODPs in ontology construction should rather obviously make ontology alignment a much more tangible task, and may be a practical path forward to overcoming the current bottlenecks in that research area [3]. In fact, it is conceivable that popular and often-reused ODPs could be enhanced with additional information tailored towards making alignments even simpler. Such information could, for example, take the form of annotated natural language sentences which act as linguistic samples for pattern content, and which thus can serve as hooks for linguistic techniques to be used for alignment. Such information could also be used to

[3] http://ontologydesignpatterns.org

enhance ontology population systems, in particular with respect to ontology population from texts. Systematic studies to verify these promises and which would show how to realize such systems have not been persued so far, to the best of our knowledge.

Acknowledgements

Some of the authors acknowledge partial support by the National Science Foundation through the project *EarthCube Building Blocks: GeoLink – Leveraging Semantics and Linked Data for Data Sharing and Discovery in the Geosciences.* Parts of this chapter appeared already in [2].

Bibliography

[1] E. Blomqvist, K. Hammar, and V. Presutti. Engineering ontologies with patterns – the eXtreme design methodology. In A. Gangemi, P. Hitzler, K. Janowicz, A. Krisnadhi, and V. Presutti, editors, *Ontology Engineering with Ontology Design Patterns: Foundations and Applications*, Studies on the Semantic Web. IOS Press, 2016.

[2] E. Blomqvist, P. Hitzler, K. Janowicz, A. Krisnadhi, T. Narock, and M. Solanki. Considerations regarding ontology design patterns. *Semantic Web*, 7(1):1–7, 2015.

[3] M. Cheatham and P. Hitzler. String similarity metrics for ontology alignment. In H. Alani, L. Kagal, A. Fokoue, P. T. Groth, C. Biemann, J. X. Parreira, L. Aroyo, N. F. Noy, C. Welty, and K. Janowicz, editors, *The Semantic Web – ISWC 2013 – 12th International Semantic Web Conference, Sydney, NSW, Australia, October 21-25, 2013, Proceedings, Part II*, volume 8219 of *Lecture Notes in Computer Science*, pages 294–309. Springer, 2013.

[4] M. Dudáš, T. Hanzal, V. Svátek, and O. Zamazal. Obowlmorph: Starting ontology development from puro background models. To be published via Springer LNCS.

[5] R. Falbo, M. Barcellos, F. Ruy, G. Guizzardi, and R. Guizzardi. Ontology pattern languages. In A. Gangemi, P. Hitzler, K. Janowicz, A. Krisnadhi, and V. Presutti, editors, *Ontology Engineering with Ontology Design Patterns: Foundations and Applications*, Studies on the Semantic Web. IOS Press, 2016.

[6] A. Gangemi and V. Presutti. Towards a pattern science for the semantic web. *Semantic Web*, 1(1, 2):61–68, 2010.

[7] G. Guizzardi. Ontological patterns, anti-patterns and pattern languages for next-generation conceptual modeling. In E. S. K. Yu, G. Dobbie, M. Jarke, and S. 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.

[8] K. Hammar. Ontology design patterns in webprotege. In S. Villata, J. Z. Pan, and M. Dragoni, editors, *Proceedings of the ISWC 2015 Posters & Demonstrations Track co-located with the 14th International Semantic Web*

Conference (ISWC-2015), Bethlehem, PA, USA, October 11, 2015., volume 1486 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2015.

[9] P. Hitzler, K. Janowicz, and A. A. Krisnadhi. Ontology modeling with domain experts: The GeoVocamp experience. In C. d'Amato, F. Lécué, R. Mutharaju, T. Narock, and F. Wirth, editors, *Proceedings of the 1st International Diversity++ Workshop co-located with the 14th International Semantic Web Conference (ISWC 2015), Bethlehem, Pennsylvania, USA, October 12, 2015.*, volume 1501 of *CEUR Workshop Proceedings*, pages 31–36. CEUR-WS.org, 2015.

[10] C. M. Keet. The use of foundational ontologies in ontology development: An empirical assessment. In G. Antoniou, M. Grobelnik, E. P. B. Simperl, B. Parsia, D. Plexousakis, P. D. Leenheer, and J. Z. Pan, editors, *The Semantic Web: Research and Applications - 8th Extended Semantic Web Conference, ESWC 2011, Heraklion, Crete, Greece, May 29-June 2, 2011, Proceedings, Part I*, volume 6643 of *Lecture Notes in Computer Science*, pages 321–335. Springer, 2011.

[11] A. Krisnadhi, Y. Hu, K. Janowicz, P. Hitzler, R. A. Arko, S. Carbotte, C. Chandler, M. Cheatham, D. Fils, T. W. Finin, P. Ji, M. B. Jones, N. Karima, K. A. Lehnert, A. Mickle, T. W. Narock, M. O'Brien, L. Raymond, A. Shepherd, M. Schildhauer, and P. Wiebe. The GeoLink modular oceanography ontology. In M. Arenas, Ó. Corcho, E. Simperl, M. Strohmaier, M. d'Aquin, K. Srinivas, P. T. Groth, M. Dumontier, J. Heflin, K. Thirunarayan, and S. Staab, editors, *The Semantic Web – ISWC 2015 – 14th International Semantic Web Conference, Bethlehem, PA, USA, October 11-15, 2015, Proceedings, Part II*, volume 9367 of *Lecture Notes in Computer Science*, pages 301–309. Springer, 2015.

[12] A. A. Krisnadhi. *Ontology Pattern-Based Data Integration*. PhD thesis, Wright State University, 2015.

[13] A. A. Krisnadhi, Y. Hu, K. Janowicz, P. Hitzler, R. A. Arko, S. Carbotte, C. Chandler, M. Cheatham, D. Fils, T. Finin, P. Ji, M. B. Jones, N. Karima, K. A. Lehnert, A. Mickle, T. Narock, M. O'Brien, L. Raymond, A. Shepherd, M. Schildhauer, and P. Wiebe. The geolink framework for pattern-based linked data integration. In S. Villata, J. Z. Pan, and M. Dragoni, editors, *Proceedings of the ISWC 2015 Posters & Demonstrations Track co-located with the 14th International Semantic Web Conference (ISWC-2015), Bethlehem, PA, USA, October 11, 2015.*, volume 1486 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2015.

[14] V. Presutti, E. Blomqvist, E. Daga, and A. Gangemi. Pattern-based ontology design. In M. C. Suárez-Figueroa, A. Gómez-Pérez, E. Motta, and A. Gangemi, editors, *Ontology Engineering in a Networked World.*, pages 35–64. Springer, 2012.

[15] V. Presutti, E. Daga, A. Gangemi, and E. Blomqvist. extreme design with content ontology design patterns. In E. Blomqvist, K. Sandkuhl, F. Scharffe, and V. Svátek, editors, *Proceedings of the Workshop on Ontology Patterns (WOP 2009) , collocated with the 8th International Semantic Web Conference (ISWC-2009), Washington D.C., USA, 25 October, 2009.*, volume 516 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2009.

[16] V. Presutti, F. Draicchio, and A. Gangemi. Knowledge extraction based on discourse representation theory and linguistic frames. In A. ten Teije, J. Völker, S. Handschuh, H. Stuckenschmidt, M. d'Aquin, A. Nikolov, N. Aussenac-Gilles, and N. Hernandez, editors, *Knowledge Engineering and Knowledge Management - 18th International Conference, EKAW 2012, Galway City, Ireland, October 8-12, 2012. Proceedings*, volume 7603 of *Lecture Notes in Computer Science*, pages 114–129. Springer, 2012.

[17] V. Presutti and A. Gangemi. Content ontology design patterns as practical building blocks for web ontologies. In Q. Li, S. Spaccapietra, E. S. K. Yu, and A. 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 128–141. Springer, 2008.

[18] V. Rodríguez-Doncel, A. A. Krisnadhi, P. Hitzler, M. Cheatham, N. Karima, and R. Amini. Pattern-based linked data publication: The linked chess dataset case. In O. Hartig, J. Sequeda, and A. Hogan, editors, *Proceedings of the 6th International Workshop on Consuming Linked Data co-located with 14th International Semantic Web Conference (ISWC 2105), Bethlehem, Pennsylvania, US, October 12th, 2015.*, volume 1426 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2015.

[19] T. Sales and G. Guizzardi. Ontological anti-patterns. In A. Gangemi, P. Hitzler, K. Janowicz, A. Krisnadhi, and V. Presutti, editors, *Ontology Engineering with Ontology Design Patterns: Foundations and Applications*, Studies on the Semantic Web. IOS Press, 2016.

[20] T. P. Sales and G. Guizzardi. Ontological anti-patterns: Empirically uncovered error-prone structures in ontology-driven conceptual models. *Data & Knowledge Engineering*, 99:72–104, 2015.

[21] D. Schober, I. Tudose, V. Svatek, and M. Boeker. OntoCheck: verifying ontology naming conventions and metadata completeness in Protégé 4. *Journal of Biomedical Semantics*, 3(2):1, 2012.

[22] V. Svátek, M. Homola, J. Kluka, and M. Vacura. Mapping structural design patterns in OWL to ontological background models. In V. R. Benjamins, M. d'Aquin, and A. Gordon, editors, *Proceedings of the 7th International Conference on Knowledge Capture, K-CAP 2013, Banff, Canada, June 23-26, 2013*, pages 117–120. ACM, 2013.

[23] V. Svátek, O. Sváb-Zamazal, and V. Presutti. Ontology naming pattern sauce for (human and computer) gourmets. In E. Blomqvist, K. Sandkuhl, F. Scharffe, and V. Svátek, editors, *Proceedings of the Workshop on Ontology Patterns (WOP 2009) , collocated with the 8th International Semantic Web Conference (ISWC-2009), Washington D.C., USA, 25 October, 2009.*, volume 516 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2009.