ON THE USE OF ONTOLOGY DESIGN PATTERN FOR LEGAL KNOWLEDGE BASE ENGINEERING

Philipp S. Thumfart

Philipp Siegfried Thumfart, Dissertant, Universität Wien, AT
Philipp@Thumfart.info

Keywords: ***Knowledge base engineering, ontology, Ontology Design Pattern, methodology***

Abstract: *In this position paper we present some advantages, that the use of Ontology Design Pattern (ODP) can have for Legal Knowledge Base Engineering and also possibilities, why the widespread use of ODP in the legal domain is still lacking. Ontologies and the knowledge bases based thereon are seldom reusable and interoperable. This results in most ontologies being built from scratch, which in turn increases the cost of ontology engineering. This can be remedied by the increased use of ODP in ontology engineering. ODP are modelling solutions to solve recurrent ontology development problems. The increased use of ODP would increase the reusability and interoperability of the resulting ontologies and would therefore decrease their engineering cost. Unfortunately, only a few ODP exist for the legal domain, since their development can be difficult and their added value has not been convincingly demonstrated yet. The second notion is due to the fact, that too few ODP are publicly available and ready to use. To standardize and accelerate the development of legal ODP, we aim to develop a methodology for creating ODP, tailored to the specific needs of the legal domain. In this paper we will present the basic structure of this methodology, which we will further developed in future work.*

1. Introduction

Since Valente and Breuker [20] proposed ontologies to be the «missing link» between legal theory and Artificial Intelligence and Law, ontologies have been widely used to represent legal knowledge and thus make it accessible for Artificial Intelligence applications.

As Gruber [9] pointed out, the term *ontology* is borrowed from philosophy, where an ontology is a systematic account of existence and for knowledge-based systems, what «exists» is exactly that which can be represented. Although the definition of ontology in computer science differs from the use of the term in philosophy, the definition is heavily influenced from it's origin in philosophy.

Gruber [9] further argues that in such an ontology, definitions associate the names of entities in the universe of discourse (e.g., classes, relations, functions, or other objects) with human-readable text describing what the names mean, and formal axioms that constrain the interpretation and well-formed use of these terms. For him, an ontology is an explicit specification of a conceptualization [8].

This notion, that an ontology is supposed to be a shared conceptualisation of a domain of knowledge, is also fundamental for the expectations we have when dealing with ontologies in the AI domain. As Valente [19] pointed out, especially in the early days of ontologies in AI, ontologies were mainly perceived as mechanism for knowledge sharing and reuse. Since the reusability of ontologies for knowledge sharing was perceived to be the main aspect, it was concluded, that ontologies should therefore be free of application specific or reasoning specific commitments [8].

But as Valente et al. [21] argued, ontologies are never completely neutral. He explains, that ontologies always assume a view of how to model the world and its fundamental parts. This is because all ontologies

are built with a bias toward a specific application, and their very usability depends on commitments that are biased towards the types of reasoning the ontology is supposed to support [19]. This bias toward a specific application can cause various problems, mainly terminological and structural differences between ontologies.

Furthermore, a predominant monolithic approach when engineering ontologies and the lack of a centralised and well developed repository for ontologies still hamper the interoperability and reuse of ontologies and causes most ontologies still being made from scratch [6, 10].

To remedy this problem, Clark and Porter [4] suggested to identify repeated Pattern of axioms in a large theory, and then abstract and reify those Pattern as components in their own right, analogous to the notion of «design Pattern» in object-oriented programming. This idea was quickly adopted and methodologies for the development of ontologies using Ontology Design Pattern (ODP) have been suggested, most notably the methodology by Presutti et al. [15]. However, to our best knowledge, no methodology has yet been suggested to develop ODP themselves and therefore, also no such methodology for legal ODP. Hence, the main contribution of this paper is laying the groundwork for a methodology to develop legal ODP from not formalized or partly formalized knowledge.

The remainder of this paper is organized as follows. In Section 2 we present a short introduction into ODP and outline their general benefits. Then, in Section 3 we describe the problem, why ODP are not used more frequently. In Section 4 we discuss the structure of a possible methodology for developing legal ODP. Finally, we conclude and present future work in Section

2. Ontology Design Pattern

An ODP describes a particular recurring modelling problem that arises in specific ontology development contexts and presents a well-proven solution for the problem [5]. Presutti et al. [16] describe several categories of ODP, differentiated by the problem they solve. These categories include for example *Structural ODP*, which solve structural problems and shape the overall ontology, *Lexico-Syntactic ODP*, which are linguistic structures consisting of a sequence of types of words associated with an assessment of the meaning they express [1], *Reasoning ODP*, which are procedures that perform automatic inference or *Content ODP* which are small ontologies with explicit documentation of design rationales, which can be used as building blocks in ontology design [2].

ODP are used in the ontology engineering process to build a coherent ontology, based on best practices. They provide, amongst others, the following benefits to the ontology engineer:

Interoperability and reusability: Ontology interoperability and the possibility to reuse these ontologies, which results thereof, lower the time and resources needed to develop ontologies [5]. ODP provide a concise implementation of the modelling solutions they provide. It is therefore difficult to be influenced by a certain bias towards an application when using ODP in ontology engineering. The resulting ontology will therefore most likely use the same terminology and follow the same structure as other ontologies that used the same set of ODP.

Computing resources: Hondros [11] already showed the benefits of creating and caching Java representations of compound ontologies to save computing resources. In their example, a cached ontology is a lightweight representation of an ontology serialised as a Java object. Its class and property hierarchies are already calculated, and for many run-time operations it can be used instead of the whole ontology itself, preventing the expensive operation of loading and classifying ontologies each time they are required. Since this requires a modular approach to ontology engineering, I argue, that Content ODP would be a perfect fit to create these ontologies. Since Content ODP naturally encourage a modular approach to ontology engineering, it would be easier to implement a similar approach as suggested by Hondros [11].

Reduce engineering time and mistakes: Since ODP encode modelling best practices, and represent well-proven solutions, they lower the entry barrier for domain experts, who lack ontology engineering practice and naturally increase the quality of the resulting ontology.

These benefits are also indicated by the studies of Blomqvist, Gangemi, and Presutti [2] and Blomqvist et al. [3] which came to the conclusion, that ODP are perceived as useful by ontology engineers, increase the quality of the resulting ontology and allow for faster ontology development.

3. Problem

However, since the introduction of the idea of ODP only a few ODP have been proposed for the legal domain [14, 7, 17]. Of course, other, more general ODP can be used to formalize the parts of legal knowledge, that deal with common knowledge, like the role pattern presented by Krisnadhi [12] or the EventCore pattern[1], but the vast majority of legal knowledge has not yet been formalized into ODP.

Hammar et al. [10] pointed out, that in order to provide convincing evidence for the added value of ODP, the community requires access to a well-organized, well-documented, and well-maintained set of interlinked high-quality ontology design Pattern. 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.

Furthermore, engineering well founded ODP obviously requires extensive domain knowledge and ontology engineering skills. This holds especially true for the legal domain, since contrary to natural sciences, where the truth of a proposition does not depend on who states it, but only empirical data supports it, in law, the truth of a statement is, among other aspects, dependent on who states it (persuasive authority)[18]. Furthermore, the heterogeneity and fragmentation of legal sources further complicates the process of formalizing legal knowledge. Since a legislator cannot lay down every possible case by law, legal norms also have to be formulated in a vague and uncertain manner to allow for a certain degree of interpretation, which further complicates ontology engineering. Therefore, the legal domain can be seen as one of the most challenging domains for knowledge representation.

If we would be able to reduce the knowledge and skill needed to develop ODP without sacrificing quality, more people would be able to contribute. We believe, that the increasing interest resulting from this lowered entry barrier will in turn convince more people to contribute to publicly available ODP, which in turn will increase the community's interest in the field.

4. Solution

As mentioned before, to the best of our knowledge, no methodology for the development of ODP has been suggested yet, neither a general one nor specifically for the legal domain. Due to the problems listed above, we see the necessity to develop such a methodology, specifically for the legal domain. This methodology we are trying to create is aimed at standardizing and accelerating the process of constructing high quality, well documented, legal ODP. The methodology will be structured in 3 stages: source material selection, pattern identification and ODP construction.

4.1. Source material selection

Legal knowledge can be represented by different types of documents, i.e. legislation, court decisions, parliamentary documents, contracts, commentaries or case-law notes. To select the most promising legal sources, we first have to classify them. Similar to Santos et al. [18] we propose a three-dimensional classification

[1] http://ontologydesignPattern.org/wiki/Submissions:EventCore.

scheme for legal sources, to identify the most promising legal sources to develop ODP from. In this scheme, the first criteria is relevance, which can be further divided into authority and legal importance [22]. Authority describes the position of the legal document within the hierarchy of norms, as prescribed by the legal system. Legal importance describes the relevance of the legal document for the legal community. The latter can oftentimes be subjective, but the amount of discourse on a legal document can be an indicator on it's legal importance.

Secondly, the legal sources have to be classified by consensus. Legal source material is vague and uncertain by nature. Since a legislator cannot lay down every possible case by law, legal norms have to be formulated in a vague and uncertain manner to allow for a certain degree of interpretation. In this scheme, consensus can also be further divided into consensus on terminology and consensus on legal consequences. Both criteria are again subjective and hardly quantifiable [18].

Thirdly, legal sources have to be classified by coverage of the specific domain of interest.

4.2. Pattern identification

After classifying the legal sources, pursuant to the three dimensions explained above, and selecting a set of the most promising ones, recurring Pattern have to be extracted from them. Following the MeLOn methodology, suggested by Mockus and Palmirani [13], we propose to extract subject, predicate, object pairs, i.e. triples from the source material. To reassure reusability and interoperability of the resulting ODP, these triples should already follow a consented terminology of terms used in the legal domain and terms used by other ODP. While extracting triples, the domain expert also has to keep track of the number of occurrences of a certain pattern and of the place(s) of occurrence in the source material.

4.3. ODP construction

While the first step of this methodology has to be done mainly by a domain expert, this gradually shifts toward the third step, which has to be done mainly by an ontology engineer. When all possible triples have been extracted from the selected source material, it is advisable to check if they are already covered by ODP, which can be reused. For this purpose, we propose to divide the triples into three categories: already covered by ODP, partly covered by ODP and not covered by ODP. Triples of the first category can be ignored, in the second category, ODP can be amended in order to cover the extracted triples and ODP for the third category have to be developed from scratch.

5. Conclusion

In this paper we presented arguments on why ODP are beneficial for ontology engineering in general. We further presented possible causes for why the widespread use of ODP is still lacking. Namely, a lack of incentive to develop ODP, since their development is difficult and time consuming and their added benefit has not yet been sufficiently proven. We argued, that this can be remedied by standardizing and accelerating the ODP development process by introducing a methodology, tailored to the difficulties of the legal domain. These difficulties are the heterogeneity and fragmentation and the persuasive authority of legal sources and the vagueness and uncertainty of legal terms. Such a methodology would therefore need to structure legal sources and find common Pattern in them while keeping consented legal terms. In this paper we already presented the basic structure of a methodology, which we are planning to further develop in the near future and present along with a usability study.

6. Literatur

[1] Guadalupe Aguado De Cea et al. «Using Linguistic Patterns to Enhance Ontology Development». In: Proceedings of the International Conference on Knowledge Engineering and Ontology Development. 2009.

[2] Eva Blomqvist, Aldo Gangemi, and Valentina Presutti. «Experiments on Pattern-Based Ontology Design». In: Proceedings of the Fifth International Conference on Knowledge Capture. Redondo Beach, California, USA, 2009, pp. 41–48.

[3] Eva Blomqvist et al. «Experimenting with eXtreme Design». In: Knowledge Engineering and Management by the Masses. Ed. by Philipp Cimianoand H. Sofia Pinto. Vol. 6317. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 120–134.

[4] Peter Clark and Bruce Porter. «Building Concept Representations from Reusable Components». In: Proceedings of the Fourteenth National Conference on Artificial Intelligence and Ninth Conference on Innovative Applications of Artificial Intelligence. Providence, Rhode Island, July 1997, pp. 369–376.

[5] Ricardo De Almeida Falbo et al. «Ontology Patterns: Clarifying Concepts and Terminology». In: Proceedings of the 4th International Conference on Ontology and Semantic Web Patterns. Vol. 1188. Sydney, Australia, 2013, pp. 14–26.

[6] Aldo Gangemi and Valentina Presutti. «Ontology Design Patterns». In: Handbook on Ontologies. Ed. by Steffen Staab and Rudi Studer. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 221–243.

[7] Cristine Griffo, João Paulo A Almeida, and Giancarlo Guizzardi. «A Pattern for the Representation of Legal Relations in a Legal Core Ontology». In: 29th International Conference on Legal Knowledge and Information Systems (JURIX 2016). 2016.

[8] Thomas R. Gruber. «A Translation Approach to Portable Ontology Specifications». In: Knowledge Acquisition 5.2 (June 1993), pp. 199–220.

[9] Thomas R. Gruber. «Toward Principles for the Design of Ontologies Used for Knowledge Sharing?» In: International Journal of Human-Computer Studies 43.5-6 (1995), pp. 907–928.

[10] Karl Hammar et al. «Collected Research Questions Concerning Ontology Design Patterns». In: Ontology Engineering with Ontology Design Patterns: Foundations and Applications. Ed. by Pascal Hitzler et al. Vol. 25. Studies on the Semantic Web. Berlin, Germany; Amsterdam, Netherlands: Akademische Verlagsgesellschaft; IOS Press, 2016, pp. 189–198.

[11] Constantine Hondros. «Standardizing Legal Content with OWL and RDF». In: Linking Enterprise Data. Ed. by David Wood. Boston, MA: Springer US, 2010, pp. 221–240.

[12] Adila Krisnadhi. «The Role Patterns». In: Ontology Engineering with Ontology Design Patterns: Foundations and Applications. Ed. by Pascal Hitzler et al. Vol. 25. Studies on the Semantic Web. Berlin, Germany; Amsterdam, Netherlands: Akademische Verlagsgesellschaft; IOS Press, 2016, pp. 313–319.

[13] Martynas Mockus and Monica Palmirani. «Legal Ontology for Open Government Data Mashups». In: 2017 Conference for E-Democracy and Open Government (CeDEM). 2017 Conference for E-Democracy and Open Government (CeDEM). Krems, Austria: IEEE, May 2017, pp. 113–124.

[14] Harshvardhan J. Pandit, Declan O'sullivan, and Dave Lewis. «An Ontology Design Pattern for Describing Personal Data in Privacy Policies». In: WOP@ISWC. 2018.

[15] Valentina Presutti et al. «eXtreme Design with Content Ontology Design Patterns». In: Proceedings of the 2009 International Conference on Ontology Patterns. Vol. 516. Washington, DC, USA, 2009, pp. 83–97.

[16] Valentina Presutti et al. «Pattern-Based Ontology Design». In: Ontology Engineering in a Networked World. Ed. by Mari Carmen Suárez-Figueroa et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 35–64.

[17] Cristiana Santos et al. «Complaint Ontology Pattern – COP». In: Advances in Ontology Design and Patterns. Ed. by Karl Hammar et al. Studies on the Semantic Web vol. 032. OCLC: on1019998480. Berlin, Germany: Akademische Verlagsgesellschaft AKA GmbH, 2017, pp. 69–83.

[18] Cristiana Santos et al. «Reuse and Reengineering of Non-Ontological Resources in the Legal Domain». In: AI Approaches to the Complexity of Legal Systems. Ed. by Ugo Pagallo et al. Vol. 10791. Cham: Springer International Publishing, 2018, pp. 350–364.

[19] André Valente. «Types and Roles of Legal Ontologies». In: Law and the Semantic Web. Ed. by V. Richard Benjamins et al. Red. by David Hutchison et al. Vol. 3369. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 65–76.

[20] André Valente and J. A. Breuker. «Ontologies, the Missing Link between Legal Theory and AI and Law». In: Proceedings of JURIX-94. 1994, pp. 65–75.

[21] André Valente et al. «Building and (Re)Using an Ontology of Air Campaign Planning». In: IEEE Intell. Syst. 14.1 (Jan. 1999), pp. 27–36.

[22] Marc Van Opijnen and Cristiana Santos. «On the Concept of Relevance in Legal Information Retrieval». In: Artificial Intelligence and Law 25.1 (Mar. 2017), pp. 65–87.