Towards Authoring of Vocabularies and Application Profiles using Dataspecer

Jakub Klímek[1,*], Štěpán Stenchlák[1], Adam Polický[1], Petr Škoda[1] and Martin Nečaský[1]

[1]*Charles University, Faculty of Mathematics and Physics, Department of Software Engineering, Malostranské náměstí 25, 118 00 Praha 1, Czechia*

Abstract

With the recent introduction of the Common European Data Spaces, a need for an approach to developing data specifications ensuring cross-domain data interoperability was identified. This means coming to an understanding of the meaning of common terms collected in core vocabularies and applying these vocabularies in domain-specific settings, with different contexts and extensions in each domain, formalized as application profiles. Although there are tools focused on authoring individual data specifications describing vocabularies, a systematic approach and tooling to assist in a consistent development of an entire distributed ecosystem of these vocabularies and application profiles is missing. In this demonstration, we show our novel extension to our Dataspecer tool, which supports an approach to authoring vocabularies and their application profiles. The extension enables data specification editors to author vocabularies and application profiles using a straightforward, web-based editor, leveraging a linked data-based description of the contents of the data specifications to foster their smooth profiling.

Keywords

vocabulary, data specification, application profile, model-driven development, linked data

Demo submission, demo and video available at https://dataspecer.com/papers/iswc2024/.

1. Introduction

The Common European Data Spaces are dependent on being able to, first and foremost, describe the data existing in them. A key building block to be used for this task is DCAT-AP [1] - a European application profile of the W3C Data Catalog Vocabulary [2], widely used mostly in open data catalogs. However, since DCAT-AP does not aim to fulfill the needs of every nation and domain, but rather focuses on the most commonly overlapping concepts, there are many extensions - *application profiles (APs)*, which further specify how exactly to apply DCAT-AP in a narrower context. These APs are, for instance, national ones such as DCAT-AP-CZ, DCAT-AP_IT, etc., but also domain-specific ones such as GeoDCAT-AP, StatDCAT-AP, etc., catering to the needs of the individual domains. A key property of a proper AP is that it complies with rules and restrictions of the profiled vocabularies and APs. Therefore, these APs form a hierarchy

Posters, Demos, and Industry Tracks at ISWC 2024, November 13–15, 2024, Baltimore, USA

*Corresponding author.

jakub.klimek@matfyz.cuni.cz (J. Klímek); stepan.stenchlak@matfyz.cuni.cz (Š. Stenchlák); adam.policky@gmail.com (A. Polický); petr.skoda@matfyz.cuni.cz (P. Škoda); martin.necasky@matfyz.cuni.cz (M. Nečaský)

0000-0001-7234-3051 (J. Klímek); 0000-0003-4843-2470 (Š. Stenchlák); 0000-0002-2732-9370 (P. Škoda); 0000-0002-5186-7734 (M. Nečaský)

where, e.g., the metadata using GeoDCAT-AP must be valid against DCAT-AP and, in turn, against DCAT. Naturally, the issue of managing all these profiles arises when, for example, a new DCAT version is published, and all the changes between the two DCAT versions need to be somehow propagated to all the APs, while maintaining their interoperability. For each AP, this requires a working group to meet frequently to first decide how to propagate the changes and what other changes to make to their respective specifications. Secondly, the working group needs to implement the changes in their specification itself, often consisting of multiple artifacts such as a human-readable specification, an RDFS [3] vocabulary, a set of SHACL [4] shapes, JSON Schemas, examples, etc. All of this is currently a mostly manual and error-prone process with limited tooling support.

There is also a key modeling aspect to profiling vocabularies, which ontology or vocabulary authoring tools such as Protégé [5], Chowlk [6] or LODE [7] do not implement, and that is the concept of an entity (class, property) profile. It is a reaction to a common vocabulary reuse problem that can be illustrated in the example of `dcat:Distribution` class, on which `dcterms:title` is re-used[1]. In DCAT, the name of the property is `title` and the definition is *A name given to the distribution.* However, this is in contrast to the DCMI Metadata Terms, which define `dcterms:title` as having a name `Title` and definition *A name given to the resource.* Although it is easy to write a different definition in a specification document, the question is how to handle this difference in data. Attaching both definitions to `dcterms:title` would be ambiguous. Creating a subproperty of `dcterms:title` would limit reuse of existing terms.

The concept of *Entity profile*, described in the SEMIC Style Guide[2] as *Reuse with terminological adaptations*, and discussed in a blog post detailing the approach to application profiling[3], tackles this issue. It was developed by the Semantic Interoperability Community (SEMIC) initiative of the European Commission, which is also responsible for DCAT-AP and its various APs. SEMIC's approach to defining core vocabularies and APs, supported by the OSLO Toolchain[4], is that vocabularies are just simple lists of terms, their IRIs, names, and definitions, to facilitate their reuse in the broadest possible context. APs consisting of entity profiles then specify the usage of terms selected from different vocabularies in a defined context, including terminological adjustments and validation expectations. Many current vocabulary specifications combine a vocabulary and an AP in one document. We call this AP *Default application profile.*

Our contributions presented in this demonstration are (1) a preliminary version of the Data Specification Vocabulary (DSV) used to capture entity profiles and (2) an extension to Dataspecer, our data specification editing tool [8], which enables specification editors to leverage entity profiles and to reuse published data specifications easier.

2. Data Specification Vocabulary (DSV)

A key aspect of our approach is the machine-readable representation of an application profile. Current specifications such as DCAT [2] usually contain an RDFS [3] representation of the

[1] https://www.w3.org/TR/vocab-dcat-3/#Property:distribution_title
[2] https://semiceu.github.io/style-guide/1.0.0/clarification-on-reuse.html
[3] https://joinup.ec.europa.eu/collection/semic-support-centre/application-profiles-what-are-they-and-how-model-and-reuse-them-properly-look-through-dcat-ap
[4] https://joinup.ec.europa.eu/collection/oslo-open-standards-linked-organisations-0

defined classes and properties. APs such as DCAT-AP [1] usually contain SHACL shapes for the validation of defined constraints such as domain/range constraints, cardinality constraints, code list constraints, etc. However, a machine-readable representation of lots of the information contained in the human-readable part of the AP is missing; e.g., labels and definitions of the reused classes and properties changed to reflect the narrower context of the AP, as shown in the `dcterms:title` example above, but also:

1. Information on the level of an entire application profile
    - Identification of the AP and the vocabularies and APs profiled by the AP
    - Relation to other APs, such as previous/next version
    - Explicit location of technical artifacts of the AP, e.g., human-readable specification, diagrams, SHACL shapes, JSON Schemas, JSON-LD contexts, examples, etc.
    - Change logs to be interpreted downstream
2. Information on the level of individual entity (class or property) profiles
    - Identification of the profiled entity and/or entity profile
    - Identification of properties (labels, notes, cardinalities, ...) changed in the AP compared to the profiled entity or entity profiles
    - Changed labels, definitions and usage notes
    - Changed domains, ranges and cardinalities

Representing the information contained in an AP in a machine-readable way, without the need for the introduction of explicit subclasses and subproperties, which is often viewed as too heavy-weight, is one of the goals of the Data Specification Vocabulary (DSV).[5] It is a work in progress used in this demonstration. Each entity profile is represented as a separate entity with its own IRI, links to the profiled class, property, or their profiles, and contains a list of changes made in the profile. In future, this representation can be used for generation of SHACL shapes and other technical artifacts as well as for checking of validity of the profiling relations and, finally, to facilitate change propagation throughout the profiling hierarchy. The DSV vocabulary and its default AP reuse The Profiles Vocabulary [9], ADMS [10], OWL [11], RDFS [3], RDF [12] and Dublin Core Terms vocabularies.

3. Demonstration Scenario

In this section, we go over a three-part scenario, which illustrates the extension to Dataspecer supporting the authoring of application profiles. The detailed steps of the demonstration together with a link to our demo instance are described on our demo landing page https://dataspecer.com/papers/iswc2024/.

3.1. Vocabulary Creation and Publication

We start by creating a simple vocabulary, a subset of DCAT [2], using Dataspecer. In a graphical editor of what resembles the UML Class Diagram notation, we create a few DCAT classes and

[5] https://w3id.org/dsv#

properties including their IRIs, names, definitions, domains, and ranges. Based on that, a human-readable specification document is generated using Respec[6] and a machine-readable RDFS/OWL file is generated and can be previewed in the tool. Before publication, the document template can be supplemented with metadata and static texts, which will remain in the specification even when the data model changes. Finally, the specification artifacts can be downloaded and published directly on a web server such as GitHub Pages.[7]

3.2. Application Profile

For the second part of the demonstration, we create a subset of the DCAT Default AP, reusing terms from existing vocabularies, combining them into a conceptual model that describes the application context. We start with creating an AP based on the vocabulary specification from subsection 3.1, and we use the Entity profiling concept implemented in Dataspecer to create entity profiles of classes and properties of DCAT and FOAF, adjusting titles and definitions, and specifying cardinalities. This again results in a previewable and adjustable Respec-based specification document and a data representation of the AP using DSV, which can be published on any web server. A notable advantage of our approach are links from the concepts in the AP documentation to the documentation of the DCAT vocabulary.[8]

3.3. Application Profile of an Application Profile

Finally, to illustrate that an AP can be profiled further, we can create an AP project based on the default AP from subsection 3.2, see the vocabularies and class and property profiles ready to be reused further, for example to create DCAT-AP [1], a European profile of DCAT and its default profile. The steps are similar; only now, the entity profiles will profile not terms from the DCAT vocabulary, but the entity profiles from the DCAT default profile, forming a hierarchy.[9]

4. Current Limitations and Future Work

Our current implementation has some known limitations, such as an insufficient edge routing algorithm in the graphical editor, missing undo/redo functionality, etc. In addition to these, there are features that we are currently working on. These include export of validation SHACL shapes [4] derived from the DSV representation of the APs, layout assistant for the graphical editor, storage of Dataspecer projects in Solid Pods, machine-processable specification change logs using LDES [13] and support of change propagation to APs based on the change logs.

Acknowledgments

The work was supported by the project no. CZ.02.01.01/00/23_014/0008787, by the Charles University project GAUK no. 262823 and by SVV project number 260 698.

[6] https://respec.org/docs/
[7] See https://mff-uk.github.io/demo-specifications/test1/ for a sample of vocabulary specification
[8] See https://mff-uk.github.io/demo-specifications/test1-dap/ for a sample of default application profile specification
[9] See https://mff-uk.github.io/demo-specifications/test1-ap/ for a sample of AP of AP specification

References

[1] J. D. Cock, M. Dekkers, P. Fragkou, A. Schiltz, A. Sofou, DCAT-AP 3.0, Technical Report, European Commission, 2024. URL: https://semiceu.github.io/DCAT-AP/releases/3.0.0/.

[2] A. Perego, R. Albertoni, D. Browning, S. Cox, P. Winstanley, A. G. Beltran, Data Catalog Vocabulary (DCAT) - Version 3, W3C Recommendation, W3C, 2024. URL: https://www.w3.org/TR/2024/REC-vocab-dcat-3-20240822/.

[3] R. Guha, D. Brickley, RDF Schema 1.1, W3C Recommendation, W3C, 2014. https://www.w3.org/TR/2014/REC-rdf-schema-20140225/.

[4] H. Knublauch, D. Kontokostas, Shapes Constraint Language (SHACL), W3C Recommendation, W3C, 2017. URL: https://www.w3.org/TR/2017/REC-shacl-20170720/.

[5] M. A. Musen, The protégé project: a look back and a look forward, AI Matters 1 (2015) 4–12. doi:10.1145/2757001.2757003.

[6] S. Chávez-Feria, R. García-Castro, M. Poveda-Villalón, Chowlk: from UML-Based Ontology Conceptualizations to OWL, in: P. Groth, M. Vidal, F. M. Suchanek, P. A. Szekely, P. Kapanipathi, C. Pesquita, H. Skaf-Molli, M. Tamper (Eds.), The Semantic Web - 19th International Conference, ESWC 2022, Hersonissos, Crete, Greece, May 29 - June 2, 2022, Proceedings, volume 13261 of *Lecture Notes in Computer Science*, Springer, 2022, pp. 338–352. doi:10.1007/978-3-031-06981-9_20.

[7] S. Peroni, D. M. Shotton, F. Vitali, Making ontology documentation with LODE, in: S. Lohmann, T. Pellegrini (Eds.), Proceedings of the I-SEMANTICS 2012 Posters & Demonstrations Track, Graz, Austria, September 5-7, 2012, volume 932 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2012, pp. 63–67. URL: https://ceur-ws.org/Vol-932/paper12.pdf.

[8] S. Stenchlák, M. Nečaský, P. Škoda, J. Klímek, DataSpecer: A Model-Driven Approach to Managing Data Specifications, in: The Semantic Web: ESWC 2022 Satellite Events - Hersonissos, Crete, Greece, May 29 - June 2, 2022, Proceedings, volume 13384 of *LNCS*, Springer, 2022, pp. 52–56. doi:10.1007/978-3-031-11609-4_10.

[9] N. Car, The Profiles Vocabulary, W3C Working Group Note, W3C, 2019. URL: https://www.w3.org/TR/2019/NOTE-dx-prof-20191218/.

[10] J. D. Cock, M. Dekkers, P. Fragkou, A. Schiltz, A. Sofou, ADMS Vocabulary, Technical Report, SEMIC, 2024. URL: https://semiceu.github.io/ADMS/releases/2.00/.

[11] OWL 2 Web Ontology Language Document Overview (Second Edition), W3C Recommendation, W3C, 2012. https://www.w3.org/TR/2012/REC-owl2-overview-20121211/.

[12] M. Lanthaler, R. Cyganiak, D. Wood, RDF 1.1 Concepts and Abstract Syntax, W3C Recommendation, W3C, 2014. URL: https://www.w3.org/TR/2014/REC-rdf11-concepts-20140225/.

[13] W. Slabbinck, R. Dedecker, S. Vasireddy, R. Verborgh, P. Colpaert, Linked Data Event Streams in Solid LDP containers, in: D. Graux, F. Orlandi, E. Niazmand, G. Ydler, M. Vidal (Eds.), Proceedings of the 8th Workshop on Managing the Evolution and Preservation of the Data Web (MEPDaW) co-located with the 21st International Semantic Web Conference (ISWC 2022), Virtual event, October 23rd, 2022, volume 3339 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2022, pp. 28–35. URL: https://ceur-ws.org/Vol-3339/paper4.pdf.