

## Seeing through Vocabularies

Kevin Ford

---

“Ontologies” are popular in library land. “Vocabularies” are popular too, but it seems that the library profession prefers “ontologies” over “vocabularies” when it comes to defining classes and properties that attempt to encapsulate some realm of knowledge. Bibframe, MADS/RDF, BIBO, PREMIS, and FRBR are well-known “ontologies” in use in the library community.<sup>1</sup> They were defined either by librarians or to be used mainly in the library space, or both. SKOS, FOAF, Dublin Core, and Schema are well known “vocabularies.”<sup>2</sup> They are used widely by libraries though none were created by librarians or specifically for library use. In all cases, those ontologies and vocabularies were created for the very purpose of publication for broader use, which is one of the primary objectives behind creating one: to define a common set of metadata elements to facilitate the description and sharing of data within a group or groups of users.

Ontologies and vocabularies are common when working with RDF (Resource Description Framework), a very simple data model in which information is expressed as a series of triple statements, each consisting of three parts: a subject, a predicate, and an object. The types of ontologies and vocabularies referred to here are in fact defined using RDF—Thing A is a Class and Thing Z is a Property. Those using any given ontology or vocabulary employ the defined classes and properties to further describe their Things, for a lack of a better word.

It is useful to provide an example. The first block of triples below represents Class and Property definitions in RDF Schema (RDFS), which provides some very basic means to define classes and properties and some relationships between them, such as the domains and ranges for properties. The second block is instance data.

```
ontovoc:Book      rdf:type      rdfs:Class
ontovoc:authoredBy  rdf:type      rdf:Property
ontovoc:authorOf   rdf:type      rdf:Property

ex:12345          rdf:type      ontovoc:Book
ex:12345          ontovoc:authoredBy  ex:abcde
```

`ontovoc:Book` is defined as a Class and `ontovoc:authoredBy` is defined as a Property. Using those declarations, it is possible to then assert that `ex:12345`, which is an identifier, is of type `ontovoc:Book` and was authored by `ex:abcde`, an identifier for the author. Is the first block—the definitions—an “ontology” or a “vocabulary?” Putting aside the question for now, air quotes—in this case literal quotes—have been employed around “ontologies” and “vocabularies” to suggest that these are more terms of art than technical distinctions, though it must also be acknowledged that there is a technical distinction to be made.

---

**Kevin Ford** ([kevinford@loc.gov](mailto:kevinford@loc.gov)) is Librarian, Linked Data Specialist in the Library of Congress’s Network Development and MARC Standards Office. He works on the Library’s Bibframe Initiative, and similar projects, such as MADS/RDF, and is a member of the ITAL Editorial Board. The ideas and opinions expressed here are those of the author and do not necessarily reflect those of his employer.



Ontologies in the RDF space frequently, if not always, use classes and properties from the Web Ontology Language (known as OWL) to define a specific realm's classes and properties and how they relate to each other within that realm of knowledge. This is because OWL is a more expressive definition language than basic RDFS. Using OWL, and considering the example above, `ontovoc:authoredBy` could be defined as an inverse of `ontovoc:authorOf`.

```
ontovoc:authoredBy owl:inverseOf ontovoc:authorOf
```

In this way, and given the little instance data above (the two triples that begin `ex:12345`), it is then possible to *infer* the following bit of knowledge:

```
ex:abcde ontovoc:authorOf ex:12345
```

Now that the `owl:inverseOf` triple/declaration has been added to the definitions, it's worth re-asking: Do the definitions represent an "ontology" or a "vocabulary?"

A purist might answer "not an ontology," but only because those statements have not been combined in a document, which itself has been given a URI and declared to be an `owl:Ontology`. That's the actual OWL Class that says, "This is an OWL Ontology." But let's say those statements had been added to a document published at a URI and declared to be an `owl:Ontology`. Is it an ontology now? Perhaps in a strict sense the answer is "yes." But in a practical sense few would view those four declarations, wrapped neatly in a document that has been given a URI and called an Ontology, as an "ontology." It doesn't quite rise to the occasion—"ontologies" almost always have a broader scope and employ more formal semantics—making its use a term of art, often, rather than a real technical distinction.

Yet, based on the same narrow definition (a published document declaring itself to be an `OWL:Ontology`) combined with a far more extensive set of class and property definitions with defined relationships between them, it is possible to describe FOAF as an ontology.<sup>3</sup> But it is widely known as, and understood as, a "vocabulary." (There is also an experimental version of Schema as OWL.<sup>4</sup>)

And that gets to the crux of the issue in many ways. Putting aside the technical distinction that can be argued to identify something as an "ontology" versus a "vocabulary," there are non-technical semantics at work here—what was earlier described as a "term of art"—about when, how, and why something is deemed an "ontology" versus a "vocabulary." The library community appears to think of their creations as "ontologies" and not "vocabularies," even when the documentation tends to avoid the word "ontology." For example, the opening sentence of the Bibframe and MADS/RDF documentation very clearly introduces each as a "vocabulary," as does FRBR in RDF.<sup>5</sup> On the surface they may be presented as "vocabularies," which they are of course, but despite this prominent self-declaration they are not seen in the same light as FOAF or Schema but instead as something more exacting, which they also are. It is worth contemplating why they are viewed principally as "ontologies" and to examine whether this has been beneficial. Perhaps the ideas behind designating something a "vocabulary" are, in fact, more in line with the way libraries operate, whereas "ontologies" represent an ideal (and who doesn't set their sights on the ideal?), striving toward which only exposes shortcomings and sows confusion.

The answer to "why" is historical and probably derives from a combination of lofty thinking, traditional standards practices, and good ol' misunderstanding. Traditional standards practices favor more formal approaches. Libraries' decades-long experience with XML and XML Schema

contributed significantly to this mindset. XML Schema provides a way to describe the precise construction of an XML document *and* it can then be used to validate the XML document. XML Schema defines what elements and attributes are permitted in the XML document and frequently dictates their order. It can further constrain the values of an element or attribute to a select list of options. In many ways, XML Schema was the very expression of metadata quality control. Librarians swooned. With the right controls and technology in place, it was impossible to produce poor, variable metadata.

In the case of semantic modelling, OWL is certainly a more formal approach. It's founded in description logics whose expressions take the form of occult-like mathematics, at least as viewed by a librarian with a humanities background. OWL can be used to declare domains and ranges for properties. One can also designate a property as a Datatype Property, meaning it takes a value such as a string or a date, as its value, or an Object Property, which means it will reference another RDF resource as its object. But these declarations are actually more about inferencing—deriving information by applying the ontology against some instance data—and not about restrictions, constraints, or validation. To be clear, there *are* ways to apply restrictions in OWL—“wine can be either red or white”—but this is a form of advanced OWL modelling that is not well understood and not often implemented, and virtually never in ontologies designed by librarians. Conversely, indicating a domain for a property, for example, is easy, relatively straightforward, and seductive because it gives the appearance that the property can only be used with resources of a specific class. Consider: The domain of `ontovoc:authoredBy` is `ontovoc:Book`. That does not mean that the `ontovoc:authoredBy` can only be used with a `ontovoc:Book` resource. It means that whatever resource uses `ontovoc:authoredBy` must therefore be a `ontovoc:Book`. Defining that domain for that property is not restricting its use only to books; it allows one to derive the additional knowledge that the thing it is used with must be a book even if it doesn't identify itself as one. This may seem like a subtle distinction and/or it may seem like tortured logic, but if it does it may suggest that one's point of view, one's mindset, favors constraints, restrictions, and validations.

And that's OK. That's library training and conditioning, completely reinforced in our daily work. It's what has been taught in library schools for decades and practiced by library professionals even longer. *Names should be entered “last name, first name” and any middle initial, if known, included. The data in this field should only be a three-character language code from this approved list of language codes.* These rules and the consistency resulting from these rules are what make library data so often very high quality. Google loves MARC records from our community for this very reason.

Wishing to exert strong control at the definition level when creating a model or metadata scheme with an eye to data quality, it is a natural inclination for librarians to gravitate to a more formal means of defining a model, especially one that seems to promise constraints. So, despite these models self-describing at a high-level as vocabularies, the models themselves employ a considerable amount of OWL at the technical level, which becomes the focus of any users wishing to implement the model. Users comprehend these models as something more than a vocabulary and therefore view the model through this more complex lens. Unfortunately, because OWL is poorly understood (sometimes by creators and sometimes by users, and sometimes by both), this leads to various problems. On the one hand, creators and users believe there are technical restrictions or constraints where there are, in fact, none. When this happens, the “constraint” is

either identified as a problem (“Consider removing the range for this property”) or—and this is more damaging—the property (read: model/vocabulary/ontology) is avoided. Even when it is recognized that the “constraint” is not a real restriction (just a means to infer knowledge), forging ahead can generate new issues. When faced with a domain and range declaration, for example, forging ahead can result in inaccurate, imprecise, or simply undesirable inferences. Most of the currently open “issues” (about 50 at the time of writing) about Bibframe follow a basic pattern: 1) there is a declaration about this Property or this Class that makes it difficult to use because of how it has been defined with OWL; 2) we cannot really use it presently because it would cause potential inferencing issues; 3) consider altering the OWL definitions.<sup>6</sup> Pursuing an (OWL) ontology, while formal and seemingly comforting because it feels a little like constraining the metadata schema, can result in confusion and a lack of adoption. Given that vocabularies and ontologies are developed and published to encourage users to describe their data in a way that fosters wide consumption by others, this is unfortunate to say the least.

It is notable that SKOS, FOAF, Dublin Core, and Schema have very different scopes and potentially much wider user bases than the more library-specific ontologies (Bibframe, MADS/RDF, BIBO, etc.). There is something to be learned here: the smaller the domain, the more effective an ontology might be; the larger the universe, a more general approach may be better. It is further true that FOAF, Dublin Core, and Schema define specific domains and ranges for many of their properties, but they have strived for clarity and simplicity. The creators of Schema, for example, eschewed the formal semantics behind RDFS and OWL and redefine domain and range to better match their needs and (perhaps unexpectedly) most users’ automatic understanding.<sup>7</sup> What is generally true is that each of the “vocabularies” approached the creation and defining of their models so as to minimize the use of formal semantics, and promoted this as a feature. In this way, they limited or removed altogether the actual or psychological barriers to adoption. Their offering was more accessible, less fussy. Bearing in mind the differences in scale and scope, they have been rewarded with a wider adopter base and passionate advocates.

The decision to create a “vocabulary” or an “ontology” is a technical one and a political one, both of which must be in alignment. It’s a mindset and it is a statement. It is entirely possible to define the model at a technical level using OWL, making it by definition an ontology, but to have it be perceived, and used, as a vocabulary because it is flexible and not strictly defined. Likewise, it is not enough to call something a vocabulary, but in reality be a model burdened with formal semantics that is then expected to be adopted and used widely. If the objective is to fashion a (pseudo?) restrictive metadata set with rules that inform its use, and which is strongly bonded with a specific community, develop an “ontology,” but recognize that this may result in confusion and lack of uptake. If, however, the desire is to cultivate a metadata element set that is flexible, readily useable, and positioned to grow in the future because it employs fewer rules and formal semantics, create a “vocabulary.” That’s really what is being communicated when we encounter ontologies and vocabularies. Interestingly, the political difference between “vocabulary” and “ontology” appears, in fact, to be understood by librarians: library models self-identify as “vocabularies.” But once past those introductory remarks, the truth is exposed quickly in the widespread use of OWL, revealing beyond doubt that it is not a flexible, accommodating vocabulary but a strictly defined model. To dispense with the air quotes: as librarians we’re creating ontologies and calling them vocabularies. We really want to be creating vocabularies that are ontologies in name only.

**ENDNOTES**

- <sup>1</sup> “Bibframe Ontology,” Library of Congress, accessed May 21, 2020, <http://id.loc.gov/ontologies/bibframe.html>; “MADS/RDF (Metadata Authority Description Schema in RDF),” Library of Congress, accessed May 21, 2020, <http://id.loc.gov/ontologies/madsrdf/v1.html>; “Bibliographic Ontology Specification,” The Bibliographic Ontology, accessed May 21, 2020, <http://bibliontology.com/>; “PREMIS 3 Ontology,” Premis Editorial Committee, accessed May 21, 2020, <http://id.loc.gov/ontologies/premis3.html>; Ian Davis and Richard Newman, “Expression of Core FRBR Concepts in RDF,” accessed May 21, 2020, <https://vocab.org/frbr/>.
- <sup>2</sup> Alistair Miles and Sean Bechhofer, editors, “SKOS Simple Knowledge Organization System Reference,” W3C, accessed May 21, 2020, <https://www.w3.org/TR/skos-reference/>; Dan Brickley and Libby Miller, “FOAF Vocabulary Specification 0.99,” accessed May 21, 2020, <http://xmlns.com/foaf/spec/>; “DCMI Metadata expressed in RDF Schema Language,” Dublin Core™ Metadata Initiative, accessed May 21, 2020, <https://www.dublincore.org/schemas/rdfs/>; “Welcome to Schema.org,” Schema.org, accessed May 21, 2020, <http://schema.org/>.
- <sup>3</sup> “FOAF Ontology,” xmlns.com, accessed May 21, 2020, <http://xmlns.com/foaf/spec/index.rdf>.
- <sup>4</sup> See “OWL” at “Developers,” schema.org, accessed May 21, 2020, <https://schema.org/docs/developers.html>.
- <sup>5</sup> See “Bibframe Ontology” and “MADS/RDF (Metadata Authority Description Schema in RDF)” above.
- <sup>6</sup> “Issues,” Bibframe Ontology at GitHub, accessed 21 May 2020, <https://github.com/lcnetdev/bibframe-ontology/issues>.
- <sup>7</sup> R.V. Guha, Dan Brickley, and Steve Macbeth, “Schema.org: Evolution of Structured Data on the Web,” *acmqueue* 15, no. 9 (15 December 2015): 14, [https://dl.acm.org/ft\\_gateway.cfm?id=2857276&ftid=1652365&dwn=1](https://dl.acm.org/ft_gateway.cfm?id=2857276&ftid=1652365&dwn=1).