Discussion of RDF/OWL Strategy for TC Review 2018-02-15:

TC would like to be involved from Step 3 on. In all likelihood there will be an overlap between TC and the RDF group but as a whole TC would like to be kept closely in the loop sooner rather than later.

**Round-tripping should be addressed in the strategy.** This has been a requirement for some years and that this should be achieved by routing through the model rather than direct binding to binding. As this may effect where and how special vocabularies are used in the RDF binding this requirement should be explicitly addressed in the strategy. Serialization in RDF must contain the same semantic content as other bindings and the need for conceptual equivalence must be stated clearly.

**Don't see anything in this document regarding the approach being used regarding the inclusion of other vocabularies.** Mapping of generated RDF to OWL ontologies in not on this list. There were discussions of this in Dagstuhl. When do we use our own predicates and when do we replace them with our own ontologies? Part of the issue was the level of effort to do this. Part is how far to reflect RDF specific limitations in the model (similar to issues regarding XML).

This modeling process first requires determining the target semantics and then looking around for existing terms which have those semantics. The ease of round-tripping with depend on how creative we get in the re-use.

**Being able to read it into Protégé.** This should be part of the strategy to verify its usability and is one of the review steps that TC will take.

Agreed, but just to set expectations, reading into Protégé will typically catch syntactic errors like malformed class definitions or property restrictions. Mistakes like accidental re-purposing of class names are rarely caught in OWL. We may be able to take some steps (e.g. enumerating disjoint classes) to increase the scope of errors detected by OWL tools.

**Specific questions:**

1 - If package is being represented in RDF this is a problem in terms of references and movement of classes between packages. Packages in the UML model are for the convenience of modelers and should not be reflected in the serializations.

agreed.

2 - We would like to see early drafts of RDF vocabulary especially those used for discovery purposes. TC members are already experimenting with what they can do using the XML and would like to do the same with the RDF. This means TC needs to be involved at Step 3.

3 - We are unsure of the meaning of 4.a. Patterns are not currently in Bindings as they are common underlying designs, not usable classes.

As I understand them, patterns capture commonalities of properties which are then have specialized use in particular classes. OWL is designed for these sorts of use cases in that a property can be defined with the common constraints and class-specific uses of that property can have additional constraints. For instance, {Aninal hasChild Animal} and {Cat hasChild Cat} (ignoring speciation) can be written:

|  |  |
| --- | --- |
| :hasChild rdfs:domain :Animal; rdfs:range :Animal . | # :hasChild is like a Pattern property. |
| :Cat owl:subClassOf [ | # in a specialization for :Cats, |
|  owl:onProperty :hasChild ;  | # every :hasChild value must be a :Cat |
|  owl:allValuesFrom :Cat |  |
| ] . |  |