

WS2 Session Notes Thursday 29 September

Back to [Technical Modeling Framework](#)

Back to [start](#)

Additional Metadata as OWL Annotation Properties

Discussion on this topic covers two distinct sets of decisions to be made:

- What metadata?
- How to render the OWL Annotation Properties for this metadata, in FIBO

What Metadata

The first question has been discussed, and we will (off line) finalize what are the specific metadata terms we need. These elements fall into the following categories:

- Contextual Metadata
 - Classification Facets
 - Context markers of various types
- 2. Semantic Provenance Metadata
 - Existing: Term Origin, Definition Origin
 - Additional:
 - Cross reference to (non origin) standards by term and definition
 - Adapted definitions and the like
- 3. Change Management Metadata
 - These are handled by the OMG
 - see [this spreadsheet](#) for current details
- 4. Archetypes

We did not discuss these further this week. The full set of metadata terms for Semantic Provenance will be defined off line. The terms required for change management will be adopted from the OMG work. The terms for contextual metadata will be dealt with in a future session.

Of these terms, many of those in the contextual and the provenance space will be fulfilled by terms from Dublin Core, SKOS, and either ISO 1087 directly or SBVR terms for those ISO 1087 terms.

We will derive a normative set of DC, SKOS and ISO 1087/SBVR terms to use, and then see what if anything is left, that is what additional metadata we will require that is not covered by terms from those standards.

The bulk of this work will be concluded off line from these sessions.

Implementation:

- DC Terms
- SKOS terms
- SBVR terms
- New OMG change management metadata

Much of what we are looking for is in the spreadsheet that Elisa sent. Development additional metadata terms to support this.

Can't round trip informal stuff.

Definition: should be in metadata.

Develop some template for adding this stuff. We would need to add something to tooling so we want to be able to take what's in the OWL Annotation Property for "Definition", and automatically clone this into the UML "Notes" construct so that people can see it.

We don't care so much about having "Further Notes".

Action: Look into how to get the things in Annotation Properties and get them into the Notes construct. Would also potentially be able to make ways of accessing the content of e.g. "Further Notes" visible.

Consensus: What if we can only clone the definition to the Notes field, and the rest is accessible via the official Adaptive interface.

Avoid tool specific integrations where possible. Apart from the Notes field, which maps into "UML Comment" and is owned by the elements and so can be round tripped.

Rendering

The ODM standard defines a number of allowed UML Base Classes (metaclasses) which may be used for any given OWL construct.

Our practice to date has been to define one UML Base Class from this set.

We should take the same approach with OWL Annotation Property.

The following UML Base Classes are defined for OWL Annotation Property in ODM:

- AssociationClass
- Association
- Property
- Class

MB: Recommend AssociationClass EK: Recommend AssociationClass

In previous sessions, there was some resistance to this choice. Discussed this now.

The rationale behind using the UML Association Class construct is that it allows for a hierarchy of such terms to be created.

What we envisage is some kind of defined hierarchy of metadata, as a hierarchy of OWL Annotation Classes. Is this a reasonable, realistic idea, or should we simply tabulate the metadata terms by name?

There was no clear consensus on this.

It was suggested that we should define the metadata as an additional Profile within the FIBO model repository. This is in line with what MB has envisaged, so we seem to be in agreement.

Action: Draft profile to be added and presented to the group, with the metadata terms we know of to date. This will be brought up to date with the final OMG decisions on Change Management terms and on a recommended set of DC and SKOS (and ISO 1087/SBVR?) terms to use.

From this, it seems that we will be looking to use OWL Annotation Property.

Discussion

We know we are going to render them as OWL Annotation Properties.

How are we going to render the OWL Annotation Class.

Definition of property or the instance of it?

- definition of the annotation

What would we write. How we import SKOS and how the instance data looks.

Not clear how to render it.

Elisa's example: uses Association Class

Consensus: Alternative views? Concerns are how to express the facts.

- that is not currently in the ODM spec. Want to update with the 1.1. ODM release, have done a lot of experiments, works consistently with built-in annotation like definedBy and stuff like DC and SKOS.

EK will show it to MB. Can mimic this. Provide profiles.

Migrating our profiles. Sam at EA has these profiles.

This will be part of our big bank profile migration.

For common metadata...

For the specific set we are going to use, we should add a specific tag. (M assumed that's what we would see). Would be an additional profile, not part of RDF OWL profile. Express in ways that can be reused not only in ODM but in other things.

Action: EK will have a crack at this, present and get feedback from the group.

This may not address terms in SBVR. Would need SKOS interpreted as RDF Schema, SKOS in SBVR and so on.

The OWL and RDF Profiles Sam Mancarella is working on. Sam is migrating those into EA. At that point, we could use those profiles.

Meanwhile, the way of representing OWL Annotation Properties is implementation sent to Sam - this is why we would have to do the profile migration before we use the set of OWL annotation properties that we are talk.

Using the new EA Profiles

Suggest people use the single base class we have defined, in doing FIBO content. This should not of course limit what other people do with ODM in other contexts.

We would was, for people editing, using or submitting FIBO stuff:

a. Only use the single base class defined for FIBO; or b. Use any base class defined as valid ODM.

Consensus: From Adaptive POV as long as there are precise rules defined going from ontology to UML then this would not be a problem.

Clearly if more than one base class is used for a given OWL construct, then there will be more than one OWL construct to a given UML base class.

Most of the issues of round tripping here would not be a problem if the stereotype is read when transforming. Previous informal transformations have not read these and have made inferences from e.g. the range.

Processing an import: hard to know if something is in the box or not, you can't see in XMI, you only see that there are properties and what they are owned by.

Display: somewhere you can display datatype properties either as a relationships pointing at the literal, or have in the box. If in the box, presumably it has the Literal shown as its datatype? This is something I may not have done very tidily and would be happy to refactor.

How people want it to look when people import into their tools, of course the diagrams won't get imported with that.

In order to get the diagrams:

See OMG Diagram Definition standard.

There won't be tool support for this just yet, but since that standard exists, presumably there will be in future?

UML specification would have to take advantage of the DD Spec before we will see it in UML. UML 2.5 will possibly take advantage of the DD standard but what we are producing now will not.

A note on OWL Annotation Properties:

These are not really very different to OWL Object Properties, or indeed any kind of OWL Property. The distinction is that reasoners know to ignore these for processing.

That is, an OWL Annotation Property has a Domain and a Range.

Therefore, to define a set of properties with a domain and a range, and to be able to arrange these in a hierarchy, we should use the Association Class base class from UML.

We will draft something for next week in the FIBO EA Repository and see how this looks, then fine tune it.

Support for Sybase Powerbuilder and Powerdesigner

Powerdesigner is the UML tool. It does not support the latest UML. API is in VB and tough to use.

It is used in some shops. Once IMM is out, then a number of our tame UML tools will be able to support those thing. That won't affect whether someone continues to use Powerdesigner in their shop.

There is some SBVR support in MagicDraw.

Do we need a statement saying what it takes to play in this standard space. For instance, for the UML, would say "UML 2.x compliant". Similarly for SBVR, would support any SBVR compliant tool for SBVR experts, and any OWL compliant tool.

SBVR should be added to the set of tooling that we support. This will not necessarily be in the current version. Not in this release but on the roadmap for the near future.

What's next?

Start with the Business entity.

Create the stand-alone piece that's the source for the standard.

Look at eliminating dependencies.

Dependencies - eliminate as many as possible while retaining the meaning.

AOB?

Sessions next week one on one for the Annotation Properties and other things.

From:
<https://www.omgwiki.org/OMG-FDTF/> - **Financial Services DTF wiki**

Permanent link:
https://www.omgwiki.org/OMG-FDTF/doku.php?id=ws2_thursday_29_september

Last update: **2011/10/10 07:46**

