VCOI WG Call Notes

*28 September 2020*

# Attendees

* Mike Bennett
* Richard Robinson
* Rob Nehmer
* Pete Rivett
* Bobbin Teegarden
* Claude Baudoin
* Frederic De Vaulx

# Agenda

* Continue the discussion from last week: URLs, wiki page outputs
* Other items (depending who is available)
  + Detailed output items per spreadsheet prototype
  + Example concepts for GovDTF
  + Ontology usage and questions
  + SKOS / MVF / Tooling?

# Meeting Notes

*From last time:*

### URI for concept

We want to have this either way.

In (1) it needs to be a # for the line

In (3) it is the wiki page URI/URL

What if:

We did (3) for the URI

Then do a Wiki table with the details (one page for all the definitions) with URI link to those pages for more info; (that also the q about what level of detail to include)

### Comments

PR: We seem to be focusing on the presentation requirements rather than what the machine readable thing should be like.

MB: That’s intentional – requirements precede machinery

Are there needs for machine readable as well?

Yes – someone might want to define a reference and have a machine read it in some other context completely.

* This is a new requirement to add to our requirements
* Example?
  + Suppose we had an ontology-style URI for each element it would be easy to define and incorporate it in an end point so that in some other document on some other website, or in some other application, grab the definition and have it pop up when the user hovers the mouse over the term
    - So a still human end point requirement
  + SPARQL end points?
    - Yes, or Graph QL end point or just a simple REST end point without a query.

Which brings us back to context – this is a different context.

To do this, need to divorce the content from the presentation.

So all these wiki etc. requirements are end presentation of content from some central (e.g. SKOS) vocabulary resource.

e.g. in FIBO ontology => SKOS vocabular automatically generated from that.

Then the wiki pages etc. would be generated from that.

Then you have 2 things (each with a URI):

* A Concept
  + URI that is e.g. a SKOS URI or an RDF or OWL URI in the Concept model
* A Term (word or set of words)
  + URI that is the Term in this Context for this TF or SIG – on this wiki page

For Pete’s use case, do we really want to reference the Term, or the Concept e.g. an OWL URI.

In MVF terminology it would be a Vocabulary element linked to a Community (the TF or SIG) and would reference a proxy for the concept, that is something in the ontology.

What MVF does v SKOS does (and indeed what SBVR does) – do we need a presentation on that? Or a simple explanation today.

## MVF Overview

PR shares MVF metamodel to give us an idea of what it does or doesn’t do for our purposes.

See Fig 1.

**MVF stands for:** Multiple Vocabulary Facility

**Driver:** allow model elements in any model e.g. UML or BPMN model to be translatable to different elements with different definitions, e.g. customer account

**Status:** Work in Progress.

**Revised submission** expected March 2021

**Wiki:** <http://www.omgwiki.org/mvf-rfp/doku.php>

**Active participants:** Pete Rivett, Evan Wallace, Elisa Kendall and Fred Cummings.

**Contributions** from Cory Casanave, Ed Barkmeyer and Robert Lario.

Diagram, schematic

Description automatically generated

**Figure 1: MVF Metamodel Current draft (work in progress)**

### MVF Features

MVFEntry is the proxy for the Concept. Proxy for what can be an ontology element, a SKOS element. Can be anywhere you like, can also be a definition on a web page. Is a proxy for the concept, and has an associated meaning. Not reinventing OWL or SKOS or others.

2 relationships: Generalization, Context

Context is one kind of SKOS Broader. Generalization is another.

This is the relation called EntryContext

Core thing (original motivation) is to apply meaning to models. Other uses will emerge from this.

MVFDictionary - a package for entries

Workspace – consists of a number of dictionaries or vocabs (bounds any lookups).

e.g. to display a name and definition, search space bounded by workspace.

Vocabulary (in workspace)

Vocab Entry has

* Term
* Definition

Vocab is associated with a Community.

Also abbrev, notes, status.

Can import other Vocabularies.

Each Vocab entry has exactly one MVF Entry i.e. one meaning.

Concept can have many terms associated with it for different community or purpose.

Displaying model element – get its MVF entry, find the Vocab entry in the current workspace.

e.g. Polish Accounting Community. Would get the term as understood in the context of that community.

Algo exists to find the correct term – using preferred and precedence to have a order for import. For a given model element may be many candidate elements that might apply.

Can also use this to define a se of vocab entries for different communities, linked to external SKOS or other. Need not only be applied to model elements, but you can. Once people go beyond what this group is doing, can extend this out.

The Term concept (Vocabulary Entry) is associated with a Definition (rather than MVFEntry having the definition). There was some discussion on this.

Why: when defining, use the language of the community to define the term to ensure it makes sense for the members of that community.

### Questions

Comparing with SKOS

SKOS has SKOS-XL that promotes term to be a more first class thing rather than a label.

MVF has more on the word / term already.

What SKOS XL does not do is associate the definition with the label.

SKOS doesn’t have quite the same capability for talking about the concept and talking about the word / term as 2 different things.

So we can use MVF to do things that SKOS doesn’t do.

Suggestion:

Make MVFEntry = SKOS Concept

No – make MVFEntry *reference* SKOS Concept (is intended to be a proxy for those)

### Things to Think About

Where the external source is already an OWL ontology – would we want to pull everything into a single SKOS resource first, or just use MVF to refer out directly to ontologies? What additional value would we get from SKOS in this case?

Do we really need SKOS in the middle since MVF can refer to whatever is the source of meaning?

Can the MVF EntryContext relation be specialized into kids of broader and narrower, like the recommended SKOS extensions?

### Recommended SKOS Extensions or BroaderThan / NarrowerThan

SKOS Broader / Narrow kinds of relation include:

* Generalization
* Type/instance
* Inclusion
* Parthood
* Topic (aboutness)
* ? other

These define the full kinds of ‘Taxonomy’ (as distinct from what we usually mean here by Taxonomy which is exclusively based on Generalization)

There are ‘recommended’ SKOS extensions for SOME of these.

Examples:

* Nairobi is narrower than Kenya (inclusion)
* French grammar is narrower than French (topic)

SKOS doesn’t even separate out the Generalization one.

MVF does separate Generalization

Can it in principle extend ‘EntryContext’ for these kinds of relation?

Can we mix and match and use some SKOS extensions within MVF??

What kind of relation is EntryContext? MVF is a metamodel and EntryContext is an association within that metamodel – does not imply any UML Association or other semantics – just the semantics of the MOF Association.

**Purpose of this:** separate out different terms to help people find the right MFV entry, e.g. if they have an model element called Account, then (to reuse something) user has to work out the right concept to associate the right model element for account e.g. banking, a story, ledger account v customer account and so on. So the aim is the able to reflect ontological differences without providing an ontology model – by selecting the right concept for the desired context.

### Comments?

For this group, want the URI for the Vocabulary, which in turn links to the MVF entry, which links to the Ontology.

The element Vocabulary Entry in MVF has a URI.

Then we have the same question as with FIBO: Given a URI can we show one thing to humans (web browsers) and another to machines (in FIBO’s case, OWL URI)?

Can that URI in MVF be associated with a wiki page, or have some link? Can implement content negotiation if the server supports is. In REST, can get back the Term and the Definition and some examples, usage etc. and plug that into the UI of another site.

## AoB

No.

All to ponder on these issues.