FIBO v2 Spec Automation Steering Call

*29 March 2019*

# Attendees

* Mike Bennett
* Jim Logan
* Cory Casanave
* Larry Johnson

Apologies:

* Pete Rivett
* Jason McColm Smith

# Topics

* LaTeX Doc Creation
* CCM Diagrams
* JIRA Co-ordination
* URIs alignment

This week we only covered CCM Diagrams issues.

There is also an update via email on the LaTeX activity.

# LaTeX

**Jason:** LaTeX update: delivered example target to Pete for converting from OWL/RDF to LaTeX template. Will advise as needed, and deliver spec template next week.

# CCM / Diagrams

## Diagram Styles

Mike to work out proposals on those. Can ask Jim for help if needed.

## FIBO Project Repository

There are a number of requirements to be co-ordinated in terms of the FIBO Repository for diagrams:

* End user requirements
	+ e.g. Fannie Mae need to not be caught out by the apparent need for SysML for saving copies of the repository file locally
* Generation of diagram content to the EDM Council site (formerly Al Fresco, now Cameo Collaborator Publisher)
* Maintenance of repository with diagrams kept up to date (currently have to re-do whenever a new repository is created)
* Change control on the repository e.g. binary in GitHub
* Collaborative working:
	+ Balance the need for this (lightly used) with nature of a stand-alone off-line file that would not allow for this

### CCM FIBO Repository Current position:

Currently we have a FIBO-Master CCM Repository on Teamwork Cloud.

This is significantly out of date. In order to support ingest of more recent OWL, all changes to URIs and modular structure (including e.g. classes moved to another ontology) have to be carried out by hand before ingesting. We do not have the information needed to do this without potential error.

Also earlier versions of CCM did not support the practice whereby (a) properties are in a different module (package) than the ones for the class, or (b) a property is introduced in one module but e.g. range declarations are made in another repository.

On (a), CCM now supports this for OWL export i.e. you can maintain properties in a different ontology to the class that is the domain, but we may have ingested OWL under versions where this was not the case, before we were aware that this was a practice in the formal OWL.

On (b), not clear whether or when this is supported in CCM.

This means that there may already be cases in the current FIBO-Master repository in which properties, or features of properties such as the range declarations, are duplicated in different parts of the model repository.

For these reasons as well as the modular changes, we want to move on from the current FIBO-Master repository, discard it completely, and create a fresh file by ingesting the OWL. Note that this will also finally remove any unused or deleted content from upper ontology and previous explanatory / conceptual material, to reflect only what is in the OWL.

### SysML Dependency

JL clarifies that this only comes into play once you try to write into Cameo Collaborator Publisher.

**Process:** Outputs to CCP are to be via the creation of a separate throwaway copy of the Repository, local to one person’s machine (e.g. Mike Bennett’s machine, which already has the SysML component in place).

## Repository Options

We want to do a controlled shift to a new CCM repository file, using the current OWL as the basis for its content.

This is straight forward but one or two moving parts need to be controlled as we do this:

* We need to support the end user requirements (no SysML dependency) as listed above
* We need to be able to generate Cameo Collaborator Publisher diagrams to the EDM Council spec site, replacing what is there
* We also need to be able to maintain and output diagrams for the FIBO v2 Specification
	+ Whether in the same repository as for the EDMC diagrams or a separate one TBD

**Action:** Cory will supply a list of the formal requirements for end users, based on Fannie Mae..

## CCM Repository Options

The options for this are:

* Server based
* Offline CCM Repository File(s)

To get there:

* **Server:** Empty and repopulate the existing MDZip Project File on TWC
* **Offline:** Create new local file and ingest

As soon as you use Cameo Collaborator Publisher in either mode it then adds SysML – causing local save issue. The local save / SysML dependency issue does not arise if the project has not been published to CCP.

### Server Based

For **server based**, it is not clear how to create a new Project Repository on the server – this may require permissions we do not have.

We can work around this by deleting everything in the current server repository and re-importing (though this may get messy e.g. there are currently multiple copies of the annotations material, other odd folders)

### Offline Repository File

We have wanted to do this for some time, got some push-back from the FND group but need to move because of the issues with maintenance of the TWC repository model content, described above.

For offline file,

* We gain flexibility and are not dependent on the server being updates to a given version.
* However we lose the ability to have people work in parallel and combine their changes
	+ This was perceived as not working during 2018 but we did not identify an issue in formal tests. We do know that people trying to do this had issues.
* The file would be maintained as a binary file in a Github repository
	+ We would need to define exactly where
	+ There may also be an option (not discussed today) to maintain it as textual XMI

### Offline: Single File v Separate components

There may be a third option, involving maintenance of (off-line) components of the repository independently.

See notes from Cory from after the meeting (last section of these notes) which includes these proposals. We only touched on these briefly in today’s call as we were trying to lock down all the variables for discussion, but this may turn out to be the optimum resolution. This requires careful design.

## CCM Versions

### Server Based Repository (TWC)

**Current status:** we were to move the TWC Repository based file to CCM 19.0 SP1 (from CCM 19.0). However, some participants needed on line help to do this, and we were unable to arrange a call due to people travelling. Otherwise we were to simply choose a date and switch over on that date.

It’s not clear why people still feel the need for this since almost nobody accesses the TWC project file anyway.

SP2: See recent announcement. Available now or soon.

**Decision:** Given the announcement, this is either available or will be soon, so for the server file anyway until we finalize these decisions, we should wait and move to SP2.

### Offline Project Repository

If we move to an offline file, the versioning issue goes away.

* MB adds later: Or it changes - presumably all users who are going to load, edit and save the repository MUST promise to use the same version as all other users, just like for the TWC version;
* However the file is not damaged if they don’t, it just becomes unreadable for anyone who is out of step.

#### CCM Version for Offline Repository

Given offline repository is starting fresh we can use SP2 (if already publicly available); or wait until it is. See recent announcement.

**Decision:** Given the announcement, this is either available or will be soon, we should wait and move to SP2 for the creation of an off-line project file.

## Creation of New Offline FIBO Repository for Diagrams

Things to figure out at the point where we move to the proposed separate off-line CCM repository for FIBO Master, for diagrams, for OWL round-tripping etc.

### Modular Configuration

Creating a new repository requires some one-off intense work rearranging the OWL ontology packages into the modular structure. This should only need to be done once – but any future URI / Module / Namespace changes should be made in the repository modular structure ahead of ingesting updated OWL with such changes.

### OWL Source

For any new offline Repository, we need to determine whether to use GitHub or the spec Website.

**Answer:** should be GitHub – this has already been established as policy.

BUT the local catalog is not working.

Cory uses his own generated catalog for this. However, for canonical populating of the repository from the OWL, we should use a clearly defined catalog (scope etc.).

#### OWL Source: Dev v Prod

The single off-line repository should use the Dev variant of the RDF/OWL. This is the one with all the OWL in 3 statuses: Release, Provisional and Informative

### Local file maintenance:

This would be maintained as a binary file in GitHub.

In the case of a local MDzip then you can’t have parallel working and merges.

### SysML Dependency

If you don’t touch Cameo Collaborator, then you don’t get the SysML dependency.

If only one person does that e.g. Mike, then if you save TWC to local Zip and THEN publish to Cameo Collaborator then it will not cause the SysML dependency for other project file users.

This would resolve the dependency issue for Fannie Mae and other end users.

**Note:** If we were to continue to use the TWC on-line variant, this would continue to make it unusable for end users like Fannie Mae as they cannot copy the latest version to a separate off-line file for further use.

# Outcomes

* We know we will move to a stand-alone file for FIBO diagrams in the near future
	+ This will be the origin for diagrams in the FIBO v2 Specification
	+ Also needs to be the point of control for round tripping of changes made to the model content using CCM as designed
	+ This usage is not currently in play because of earlier maintenance issues but is still the intended use of the or a CCM Repository
* This will be created using CCM 19.0 SP2 when that is available
* OWL will be ingested from the EDM Council GitHub repository
* This will be all the OWL, via the ‘Development’ status tag
* The catalog there is currently broken; this to be fixed before we do this
* One-off activity will be needed to arrange ontologies into modular structure
* We will not have parallel working on diagrams, but people can check it into and out of GitHub – these will need to be coordinated
* Diagrams for the FIBO v2 formal Specification – we did not discuss this week, but presume these are output as a report from the new offline repository project file in a form that can be digested by the LaTeX – per previous (15 March) call notes
	+ This report may need to be developed, ideally as a simple adaptation of an existing report
	+ This is not the same as the CCP report with the SysML dependency, so in effect anyone can run this
* This new repository will never directly output to Cameo Collaborator Publisher
	+ Therefore the dependency problem that prevents people from saving a copy locally will not arise
	+ Instead, a separate throwaway clone of the repository will be created on one machine (Mike Bennett, plus one backup) in which SysML has been installed; this will be used to update the web content on spec.edmcouncil.org for diagrams
* Cory will supply a list of other requirements that may impact or have impacted end users
	+ Done - see below
* Mike will work on defining formal diagram types, contexts, method (autolayout etc.) for repeatability, and define consistent naming rules (per 15 March notes), for the formal FIBO v2 Specification in LaTeX. This will either not be, or only coincidentally be, like the business facing diagrams.
* **Not discussed:** perhaps the same OWL can be used to originate 2 repositories, one for each kind of diagram, and all future updates of OWL be ingested to both in lock step (automatically?)?
	+ Then one would have hand curated diagrams, the other be and only be diagrams for the Specification.
	+ The difficulty otherwise is that reports will output all diagrams that are present, unless some are put in a sub-folder that is excluded from the report scope, but then those diagrams can’t be maintained without moving them back

# Topic: Availability of UML version of FIBO and diagrams (again)

## From Cory Casanave

*Via email, after the call:*

Goal: Effective use of CCM (and other UML tools if we get them) to extend, manage and diagram FIBO by internal and external users.

Don’t expect a “big bang” of diagramming everything, allow diagrams and curated viewpoints to grow.

Requirements:

* UML models must exactly match and be in sync with published OWL – no extra or experimental stuff.
* Models should mirror FIBO ontology structure (it would be nice if import did this!)
* Versions of FIBO (e.g. pub, dev), should be in separate models – pub imported into dev.
* Element identities between versions must never change such that other models using these resources (such as diagrams or extensions) don’t break. So, the models must be refreshed from scratch ONCE and then always updated from that point forward.
* Published user downloadable .xmdzip and .xmi files for FIBO corresponding exactly to each release must be available.
	+ We should have versions for pub and dev
	+ Interim version based on interim OWL can also be available in git
* The files should be under version control with checkin-checkout.
* For CCM users, these models should contain a growing set of curated diagrams. There should be a way for users to contribute new diagrams and views.
* Generic UML 2 XMI should also be available
* Recognize that enterprise users operate behind firewalls and can’t update their software easily, they may only have “base” magicdraw & CCM
* If possible, also generate diagrams for everything (like NIEM-UML did).

Options:

* File based
	+ GIT locking is used
	+ .mdzip file is updated “transactionaly” , the internal team would be informed when this happens (based on new diagrams or updating to the master OWL branch)
	+ New contributed diagrams could be made in a separate model that imports the master, changes to be imported into the mainline would be moved in during a transaction. This also allows control over acceptance of a diagram set.
* Server based
	+ Start with a clean workspace!
	+ Test that the server can export a “clean” set of linked .mdzip files (e.g. for pub/dev) without sysml, never “dirty” the file with dependencies
	+ The internal team can update from OWL and create diagrams directly in server
	+ External users would contribute diagrams in the same way as file based
* Diagram locations and other choices
	+ I have mixed feelings about having the diagrams in with the model elements or in separate view hierarchies. CCM makes some assumptions about diagram colors and locations, but keeping views separate may make sense and allow for different perspectives. Perhaps some “foundational” or generated diagrams are in with the model. Having views separate would also allow file based version control to be easier – as the model files would not need to be read/write.
	+ There is also the questions of one MDZIP or multiple – at this point I think one is better.
	+ Since we have different URIs in published and git FIBO, there is a question which should be used. I have been using published FIBO but need a way to update it in the interim – which requires the “publish” stack. We do need to be able to work with the git version of FIBO.

In that very few update these models and that there are sometimes issues with model servers, I would use file based to start. Files can always be loaded into servers.