CoreFiling feedback on the XBRL Taxonomy related to the EBA final draft Implementing Technical Standards on Supervisory Reporting Requirements

Introduction

This document sets out CoreFiling's feedback on the Draft XBRL Taxonomy provided with the consultation papers.

Domains for QNameItemType concepts are not in the DTS of the entry points in the taxonomy package

Item concepts with schema type xbrli:QNameItemType have model:domain and model:hierarchy attributes which determine the starting domain member and role for the Dimensional Relationship Set defining the domain for the fact.

Using the entry points specified in the .taxonomyPackage.xml file, almost every concept fitting this description has an empty domain. This prevents applications from presenting users with the correct list of choices for facts with QNameItemType.

We suspect this is accidental. There are definition relationships defining useful domains for these concepts, but they have been omitted from the DTS of the taxonomy entry points. For example, for

<xs:element name="ei326" type="xbrli:QNameItemType" substitutionGroup="xbrli:item" id="eba_ei326" xbrli:periodType="instant" nillable="true" model:domain="eba_exp:SC" model:hierarchy="http://www.eba.europa.eu/xbrl/crr/role/dict/dom/SC/SC1" model:creationDate="2013-09-18" />

the domain is defined in the files

www.eba.europa.eu/eu/fr/xbrl/crr/dict/dom/sc/hier{-defl-prel}.xsd

but these are unreferenced from elsewhere in the taxonomy so cannot be discovered from any of the DTS entry points.

refPeriodStartDate/refPeriodEndDate type issue

We think we have identified an issue with the two parameters refPeriodEndDate and refPeriodStartDate defined in www.eurofiling.info/eu/fr/xbrl/val/params.xml. These declare the data type of the parameter as "xs:date", but because they select period values which are typed by XBRL to xbrli:dateUnion then the actual value could be an xs:dateTime.

It is currently not well defined in the Variables specification what form of automatic conversion should be done in this situation, and processors may raise xbrlve:parameterTypeMismatch error owing to the data type mismatch.

Our colleague Richard Ashby has raised a bug against the XBRL Formula Specification to track this issue:

http://bugzilla.xbrl.org/show_bug.cgi?id=516

To avoid ambiguity we would recommend explicitly converting the value by wrapping the entire select statement with the xs:date() constructor function.

i.e. replace:

max(//xbrli:period/(xbrli:instant | xbrli:endDate))

with:

xs:date(max(//xbrli:period/(xbrli:instant | xbrli:endDate)))

and similarly for the other parameter.

Versioning

The published taxonomy package contains a number of files that are not obviously subject to a versioning approach:

- www.eurofiling.info/eu/fr/xbrl/*
- www.eba.europa.eu/eu/fr/xbrl/crr/dict/*
- www.eba.europa.eu/eu/fr/xbrl/crr/fws (fws.xsd, fws-lab-en.xml)

As we understand it:

- The eurofiling.info domain contains content that may be shared between EBA and EIOPA
- The dict/ directory contains concept/member definitions used elsewhere in the taxonomy
- fws.xsd has one concept for the two frameworks (COREP, FINREP)

As no version number or date is used to identify these files we are concerned as to how changes to these files will be managed.

Our concerns are primarily practical rather than theoretical and may affect XBRL implementations using these taxonomies.

As background, we assume that robust XBRL implementations will configure their XBRL processors with fixed, local copies of XBRL taxonomy documents rather than rely on Internet access.

Unversioned content causes problems using multiple versions of the taxonomy

Leaving aside the eurofiling.info content, a CRD IV taxonomy release consists of a versioned and unversioned part: the framework and the dictionary.

As there is no version number in the file locations we assume that the dictionary will be added to over time but never removed from or otherwise incompatibly modified.

Even with this approach, there is still an issue using multiple versions of the taxonomy in an XBRL processor. Consider two CRD IV taxonomy releases, the second adding new concepts to the dictionary. If an XBRL processor was naively configured to use both taxonomies then there would be conflicting copies of the dict files. If the old copy was chosen for a new filing then errors would result. If the new

copy was chosen for a filing against an old framework release then there is a risk that content might be accepted that should not be.

It might be argued that the framework hypercubes would preclude new dictionary concepts being used with an older framework, and that correct configuration of an XBRL processor would use the newer dictionary files (as they are backwards compatible) with both the old and new framework. This is not easy to achieve with the files as published for the consultation. If mixing and matching of the dictionary and framework is necessary for these deployment scenarios then they should be released as separate packages and both the framework and dictionary should have an associated version number on the package. A given framework release should require a minimum dictionary version.

The above issues would be avoided if the dictionary was versioned alongside the framework with a version number or date in its file locations. This does not necessitate using a version number in the schema namespaces (though this would be necessary to enable users to create a single document that refers to both versions, which may or may not be a desirable usecase).

Unversioned shared eurofiling.info content may cause problems using Solvency II and the CRD IV taxonomy together

A similar issue exists with eurofiling.info content, but this is further complicated by a lack of clarity around updates and release due to the shared nature of this content.

As there is no version number in the file locations we assume that they will be added to over time but never removed from or otherwise incompatibly modified.

Section 4 of *Representation of XBRL in the Data Point Model* suggests these shared files may not be included in the final release. For users who want to work with known-good local copies it would be helpful for Eurofiling to provide releases of these files. A CRD IV taxonomy release should be documented as depending on a minimum versioned (or dated) release of these files.

In-place updates of eurofiling.info content during the development of Solvency II and the CRD IV taxonomy have caused Solvency II instances that were valid against the June Solvency II release to become invalid when using the latest eurofiling.info content. At the time of the June Solvency II release, filing indicators used find:table. This is no longer in the current filing indicators schema, so documents produced at the time of that release are now invalid against the new schema. We understand that the intent is not to make this kind of backwards incompatible change when the taxonomies are no longer at draft stage, but a versioning approach based on in-place updates carries inherent risks of unintentionally invaliding old content.

Redistribution of xbrl.org and eurofiling.info artefacts in the CRD IV taxonomy package

In addition to the eurofiling.info artefacts mentioned above, the CRD IV taxonomy package contains documents from xbrl.org. A single .taxonomyPackage.xml file provides remappings for all three domains. Since eurofiling.info and xbrl.org are not directly controlled by the EBA, the offline copies provided by the EBA for convenience would be better placed in separate taxonomy packages, allowing applications to substitute alternative packages (with more recent content) as appropriate.

Filing indicators

(Substantially similar feedback was previously provided on CWA 1 itself but we understand that change in this area would need to be driven by EBA / EIOPA, hence its inclusion here.)

It seems very odd to model all tables of data in the filings using one technology (Table Linkbase) and then resort to a completely different technology (tuples) for defining what is effectively a table reporting which other tables have been reported.

The use of an inconsistent technology is itself unhelpful, as it requires software developers to support two different ways of working with data, but the particular approach being taken presents a number of practical problems:

1. It is not extensible. There has been some discussion of adding "description" text to the filing indicators to explain why a particular table is or is not being reported. As there is no way for users of the taxonomy (European and national regulators) to alter the definition of the find:table element, implementing this has led to discussion of even more obscure corners of the spec in the form of footnotes, which of course don't benefit from data typing, and are difficult to address using technology such as Formula.
2. The use of tuples makes merging and splitting instances documents harder. It is often convenient to split a filing into separate instance documents for preparation and review, before merging into a single document prior to submission. For the most part this can be achieved with a simple XBRL merge, but the use of a single parent tuple for filing indicators means that the filing indicator data requires special treatment during the merge.
3. Tuples are increasingly seen as a legacy technology, and we expect to see support for them reduced in the future. For example, the Table Linkbase provides no specific constructs for handling tuples.
4. The proposed approach violates CWA 1 Filing Rule 2.24 regarding duplicate facts because the facts representing different tables would also be considered duplicates according to the XBRL specification. This may cause problems when representing the filing indicator XBRL in a database or other format that does not cope with inconsistent duplicate values.

Our recommendation would be treat this filing data in the same way as all other data in the filing, and model it as a table using the Table Linkbase. Failing that, the find:fIndicators parent should simply be removed as it is serving no useful purpose at all.

Use of the verbose label role for table codes

The taxonomy provides Table Linkbase table resources with standard and verbose generic labels. The standard label includes the table name and a more descriptive label. The verbose label is simply the table code. For example for C_06.00:

Standard:	C06.00 (GS) Group Solvency
Verbose:	C_06.00

This does not seem to fit with the intent of a verbose label, which should generally provide more detail than a standard label. Consider using a terse label or swapping the standard and verbose label values.

Further, section 8.5 of *Representation of XBRL in the Data Point Model* says:

"Identification of templates on find:filingIndicator facts is made using codes. These codes are the documentation label of a table:table resource (in case a template is reflected by a single individual table) or the common label of a set of tables. "

Very few documentation labels (role http://www.xbrl.org/2008/role/documentation) are present in the taxonomy and they do not appear to provide table labels as described above, whereas verbose labels (role http://www.xbrl.org/2008/role/verboseLabel) do.

A specialised label role akin to rc-code may be the most appropriate way to record this information.

Use of non-standard arcroles and attributes

The DTS contains arcroleType definitions for the following arcroles:

- http://www.eurofiling.info/xbrl/arcrole/complete-breakdown
- http://www.eurofiling.info/xbrl/arcrole/partial-breakdown
- http://www.eurofiling.info/xbrl/arcrole/superset-breakdown
- http://www.eurofiling.info/xbrl/arcrole/model-properties
- http://www.eurofiling.info/xbrl/arcrole/group-table
- http://www.eurofiling.info/xbrl/arcrole/applies-to-table

All except superset-breakdown are used in the taxonomy, but documentation is limited and we are not aware of any conformance suites to ensure consistent processing.

Interoperability would be improved if the arcroles were included in XBRL.org's Link Role Registry, or incorporated into an XBRL.org specification.

A similar issue exists for the custom attributes in the http://www.eurofiling.info/xbrl/ext/model namespace.

In the meantime, it would be helpful if model.xsd included a link to comprehensive documentation produced by the Eurofiling group. For maximum discoverability, links to the documentation could also be served at the URIs corresponding to the arcroles and the model namespace. The W3C follow this approach for many of their namespaces, including that of the XML Schema specification:

http://www.w3.org/2001/XMLSchema