

Innovations in XBRL Formula validation

EuroFiling, Frankfurt 13th June 2023

Paul Warren, Technical Director, XBRL International
Mark Goodhand, Head of Research, CoreFiling

Overview

- Future of XBRL Formula
- Optimising DPM-based XBRL Formula Rules

Future of Formula

XBRL Formula Development

- Embedding validation rules in a taxonomy is very powerful
- Backed by great infrastructure to ensure interoperability (conformance, certification)
- Faces multiple challenges
- Arrived at a fork in the road:
 - Evolve existing platform
 - Make a more radical change



Formula challenges

- Syntax is cumbersome and hard to read/write
 - Tied to XML (xBRL-XML) – we now have xBRL-JSON, xBRL-CSV and OIM
 - Performance issues in some environments
 - Can't write all the rules we want to:
 - Limited taxonomy access
 - Limited fact alignment features
 - New ESG requirements
- } US data quality rules rely on these

Formula solutions

XF – text based syntax for Formula

OIM Formula – remove XML dependency

XULE – alternative rules approach from

XBRL US

"Table based formula" – optimise Formula for CSV?

Evolve Formula – more filters, more functions, XPath 3, "chaining"?



Not so much a fork in the road....

Oxford Formula strategy meeting (March 2023)



Oxford Formula Plan



Short term: Formula 2.0

- **XF** – text-based formula
- **OIM Formula** specification

XF – text-based formula



Short-term – "Formula 2.0"

- Published as 2018 Working Group Note
- No new functionality, just a different syntax for XBRL Formula
- Straightforward transformation between text and linkbase syntax
- Rapid formalisation as full specification planned
- New syntax for referencing XF files directly from taxonomies

```
namespace eg = "http://example.com/taxonomy";
assertion NonNegativeRevenue {
    variable $v1 {
        concept-name eg:Revenue;
    };
    test { $v1 ge 0 };
};
```

OIM Formula



Short-term – "Formula 2.0"

- Current specs tell us:
 1. How to get from xBRL-JSON/xBRL-CSV to xBRL-XML
 2. How to apply XBRL Formula to xBRL-XML
- In theory, you can run existing Formula on xBRL-CSV/xBRL-JSON
- But, it's desirable to be able to run rules directly on xBRL-CSV/xBRL-JSON

OIM Formula



Short-term – "Formula 2.0"

- Create a specification that prevents XBRL Formula Rules from using functionality that requires an XML document/model:
 - XPath node navigation
 - Certain Formula functions
 - Define meaning of "context node" in all cases
- Goal: no ambiguity about whether rules in a taxonomy can be evaluated without an XML model
 - "This taxonomy is OIM/xBRL-CSV compatible"
- Goal: allow processors to switch into "OIM mode" early (no XML DOM), and raise errors if Formula rules attempt prohibited operations

OIM Formula - status

- PWD expected in coming weeks
- Will include new registry functions (e.g. f:period, f:entity-identifier)
 - Consistent with OIM model and terminology
 - Avoid relying on XML syntactic constructs (e.g. contexts)
- Existing registry functions will be classified as:
 - Supported – can be used in OIM Mode
 - Unsupported – cannot be used in OIM Mode
 - Deprecated – can be used in OIM Mode, but there's a better way now.

OIM Formula



Long-term – "XBRL Rules 3.0"

- Agreement that the next major version of Formula should not be based on current foundations
- Considered modernising with XPath 3.0, but this seems like an evolutionary dead-end
- Fundamental issues with the Formula processing model still remain
- Formula 3.0 plans to make a clean break, and expected to draw heavily on the experience gained with XBRL US "XULE" language
- Natively OIM-compatible
- Will drive the development of OIM Taxonomy model

XULE

- Developed by XBRL US to support "Data Quality" rules
- Has strong support for:
 - Checking (extension) taxonomy contents
 - Driving rules based on taxonomy metadata (simplifies rule maintenance)
- Text-based syntax
- Multiple implementations exist, but not backed by a formal standard or conformance suite

Future of Formula - summary



"XBRL Formula 2.0"

- Rapid formalisation of XF
- Completion of OIM Formula

"XBRL Rules 3.0"

- New language
- Text-based
- Natively OIM compatible
- Built on XULE experience

Alignment with European requirements

- XBRL Europe now has a Standard Roadmap Working Group
 - Chaired by Thomas Klement and Mark Goodhand
 - Aims to ensure European XBRL reporting requirements are addressed by international standards
- EBA TFERF group identified several concerns with XBRL Formula
 - Validation time and hardware requirements
 - Complexity of implementing validation rules
 - Complexity of debugging validation rules
 - Coverage of use cases by validation rules
- DPM architecture has certain special features
 - Three models in one: datapoints, templates and semantic dimensions

Formula evolution for European reporting

Problem	Solution
Validation time and hardware requirements	XF <ul style="list-style-type: none">• Reduces taxonomy size OIM Formula <ul style="list-style-type: none">• Reduces report model memory requirements• Avoids CSV to XML conversion cost
Complexity of debugging validation rules	XF <ul style="list-style-type: none">• Assertions are comprehensible to humans
Coverage of use cases by validation rules	Formula 3.0 enhancements
Complexity of implementing validation rules <ul style="list-style-type: none">• Business experts think in terms of templates	<i>Template-centric assertions with XF?</i>

Template-centric assertions

- Load xBRL-CSV into an OIM model aligned with business templates
 - Template, row, column and sheet codes as dimensions
- Comprehensible assertions using an international standard
 - No need for custom DPM-XL and DPM-ML languages
 - Concise expression of rules across rows and columns
 - Simpler dimensional model (vs full semantic dimensions)
- Works with XBRL Formula as it stands today
- Benefits from future enhancements to XBRL Formula
- Reaffirms taxonomy as the Single Source of Truth for validation

```
"dp454790": {  
  "decimals": "$decimalsMonetary",  
  "dimensions": {  
    "concept": "eba_tmpl:mi",  
    "eba_tmpl:T": "F 36.01.a",  
    "eba_tmpl:R": "150",  
    "eba_tmpl:C": "70",  
  },  
  "eba:documentation": {  
    "CellCode": "{F 36.01.a, r0150, c0070}",  
    "logicalDataPointId": "41728"  
  }  
}
```

DPM-XL transformed to abbreviated XF

with {tC_90.00, r*}: {c0070} * {c0080} = {c0010}



```
1 namespace dpm = "http://www.eurofiling.info/xbml/ext/data-point-model";
2 cover all except t, s, r, c
3
4 assertion v10101_m {
5     unsatisfied-message (en) "with {tC 90.00, r*}: {c0070} * {c0080} = {c0010}"
6     with t[C_90.00]
7     test {c[0070]} * {c[0080]} = {c[0010]}
8 }
```

54 Characters (*1,1)

Get involved

- Open questions
 - How slow are things now?
 - How fast is fast enough?
 - Which tables & rules cause the most trouble?
- What we need
 - Realistic test data
 - Performance testing
 - Assistance with prototyping
 - Assistance with specification development

Regulators



Software vendors

