Piotr Malczak

Formula Meta Description

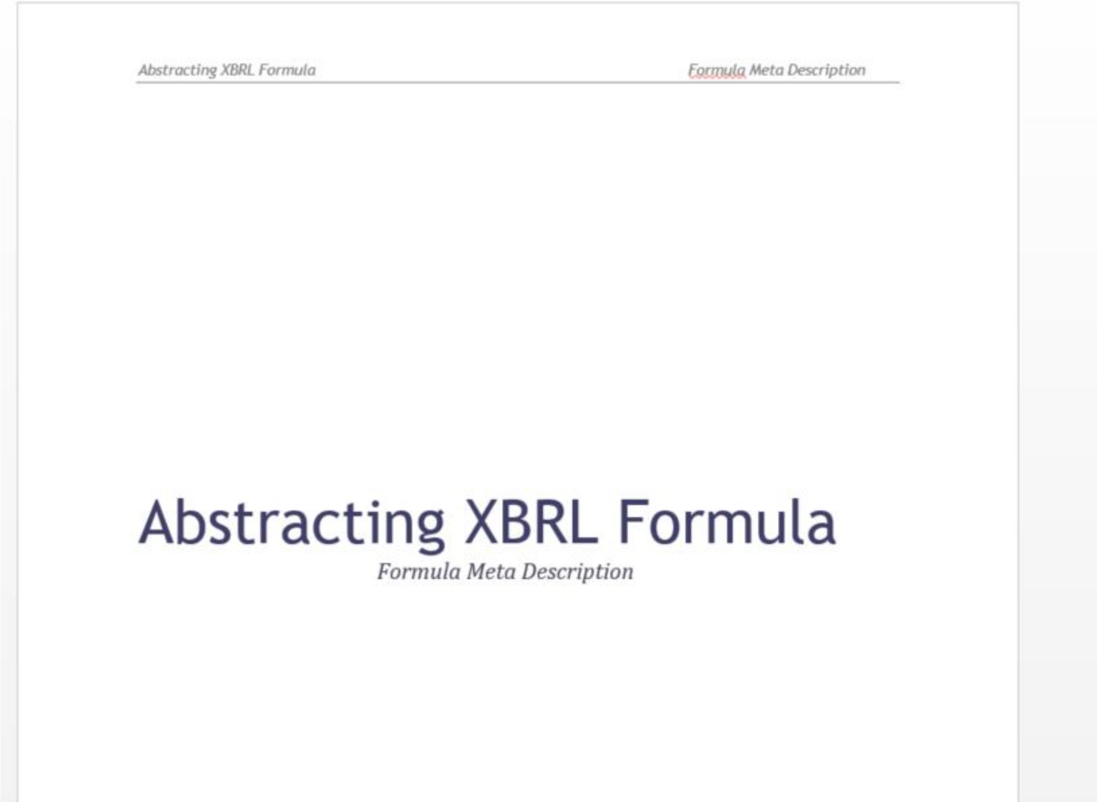
Eurofiling Workshop
User-friendly assertion notations - PANEL

Luxembourg 2013-12-10

GPM

AGENDA

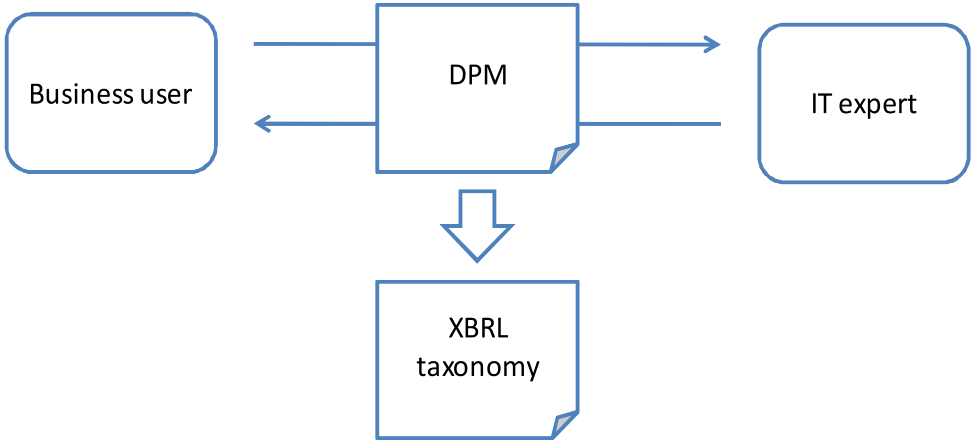
- What is FMD about ?
  - FMD components
    - Validation patterns
    - Addressing schemas
  - Examples
- Data addressing vs table addressing



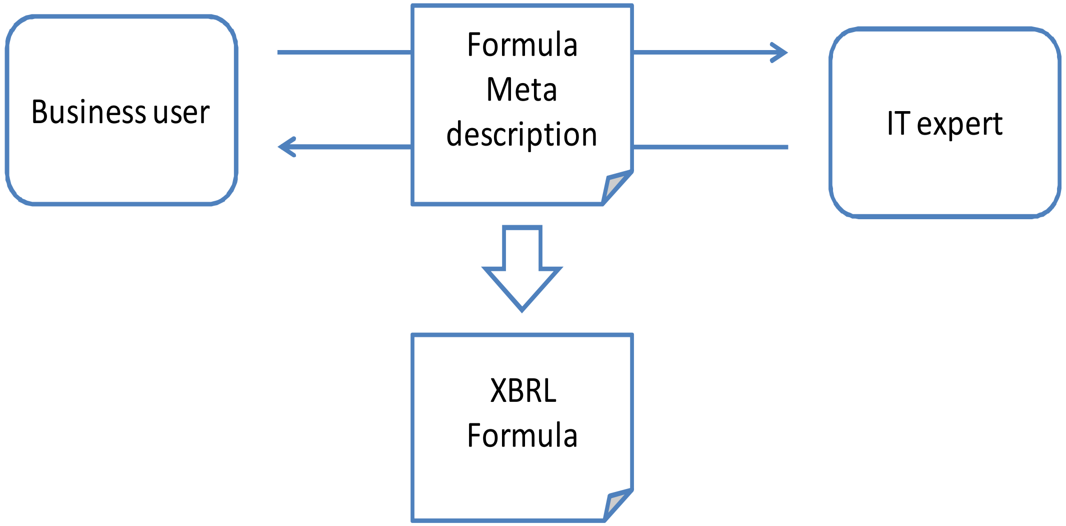
Abstracting XBRL Formula Formula Meta Description

Abstracting XBRL Formula

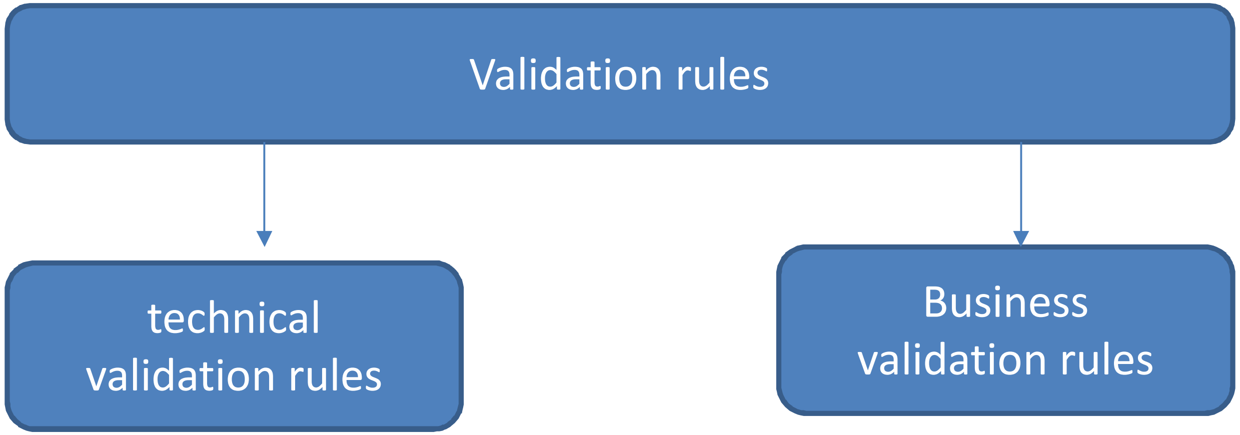
Formula Meta Description



FMD is a kind of tool which allows project participants
to communicate effectively



FMD is for business validation rules



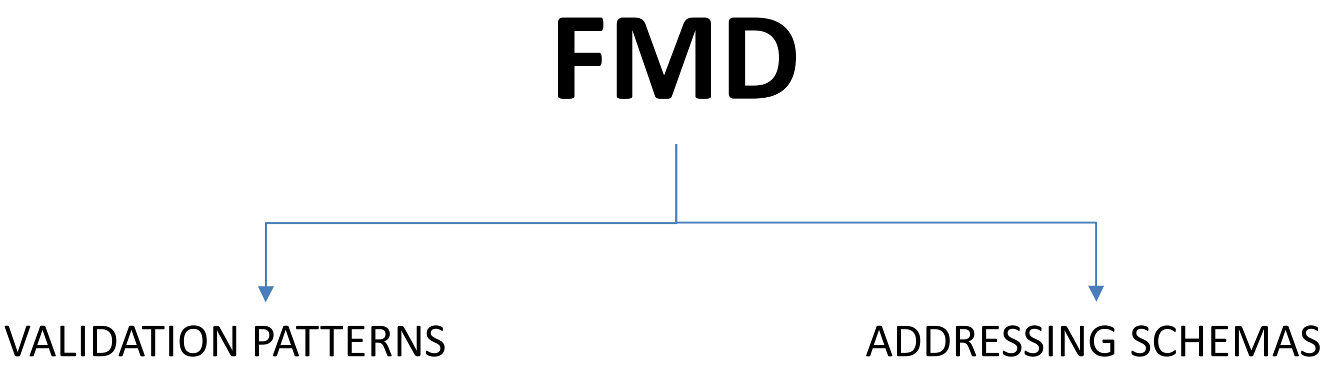
The purpose of technical validation rules is to check technial aspects of XBRL Instance e.g.:

- uniqueness of units
- entities validation
- uniqueness of typed dimensions values

Business rules are about data.

FMD

FMD components



FMD

VALIDATION PATTERNS

ADDRESSING SCHEMAS

- LOGICAL RELATIONSHIP
- EXISTANCE ASSERTION
- ..

- SIMPLE (DIRECT)
- DIMENSIONAL
- CONCEPT

Each validation rule defined in FMD is composed of pattern applied to the given addressing schemas.

e.g.

The most common: LOGICAL RELATIONSHIP in SIMPLE ADDRESSING SCHEME

Validation patterns

Existence assertion

– (the simplest V.P.) for expressing that a certain field (cell)
is either required or forbidden

Logical relation

- (the most common) for examining left-hand and right-hand expression
against given logical operator
a + b >= x + y

Conditional assertion

- (very common) examining logical expression
only in case of another expression (precondition) beeing true
if (A > 0) then check (a + b >= x + y)

Complex assertion

- logical expression
isNotNull(A) and (B >= C) and (D > 0)

FMD addressing schemes

Simple

formula operates on single data points

Dimensional

validation expression relates to dimensions

Concept

validation expression relates to concepts

Tuple addressing scheme – (unwanted kid) relates to tuples

Example

Assets	[A]
Liabilities	[L]
Equities	[E]

```
A = L + E
```

		[EUR]	[USD]	[T]	
Assets	**[A]**			A,T =	A,EUR + A,USD
Liabilities	**[L]**			L,T =	L,EUR + L,USD
Equities	**[E]**			E,T =	E,EUR + E,USD

A,T = L,T + E,T

A,USD = L,USD + E,USD

A,EUR = L,EUR + E,EUR

The common pattern is LOGICAL RELATION
and addressing scheme is SIMPLE ADDRESSING SCHEME

A,EUR represents a single data point – in this case Asset, Currency: EUR

```
A,T = A,EUR + A,USD
L,T = L,EUR + L,USD
E,T = E,EUR + E,USD
```

These three validation rules differ only in one component – concept.

Dimensional addressing scheme:
- Logical relations are defined in a context of dimension
- Validatoin rule can be applied either to ALL or
  to selected concepts (filter)

```
*,T = *,EUR  + *,USD  applied to [A,L,E]
```

```
A,T   = L,T   + E,T
A,USD = L,USD + E,USD
A,EUR = L,EUR + E,EUR
```

These three validation rules differ only in one component - dimension.

Concept addressing scheme:

- Logical relations are defined with concepts
  (to some extent, similar to calculation linkbase)
- Validatoin rule can be applied either to ALL
  or to selected dimension/s (filter)

```
A,* = L,* + E,*  applied to [T,EUR,USD]
```

Example

		[EUR]	[USD]	[T]	
Assets	[A]				
Liabilities	[L]				
Equities	[E]				

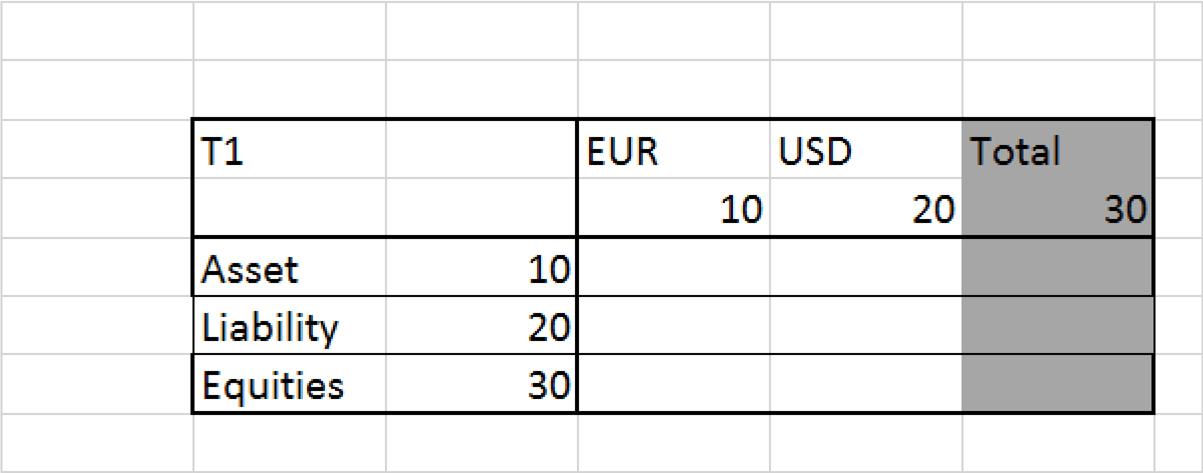
```
*,T = *,EUR  + *,USD  applied to [A,L,E]

A,* = L,* + E,*  applied to [T,EUR,USD]
```

IMPORTANT OBSERVATIONS:

1. FMD can reduce the numer of validation rules !!
2. FMD (as XF) will do its work regardless of the presentation layer

Data addressing vs table addressing



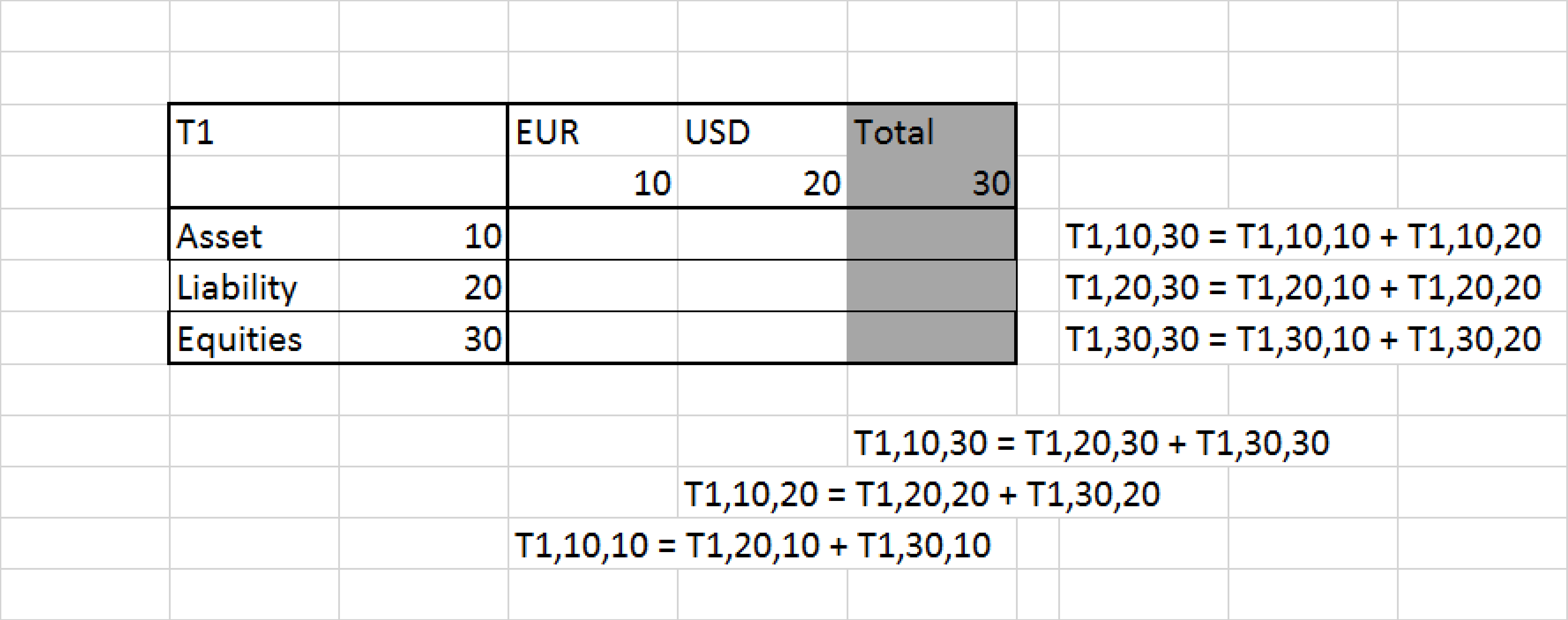
T1		EUR	USD	Total
		10	20	30
Asset	10			
Liability	20			
Equities	30			

Table addressing scheme is based on cell coordinates.
e.g.

T1.10.20

This is the most common addressing scheme.
Business users prefer to use excell like coordinates instead of data concepts ID’s.

Data addressing vs table addressing



T1		EUR	USD	Total
		10	20	30
Asset	10			
Liability	20			
Equities	30			

T1,10,30 = T1,10,10 + T1,10,20

T1,20,30 = T1,20,10 + T1,20,20

T1,30,30 = T1,30,10 + T1,30,20

T1,10,30 = T1,20,30 + T1,30,30

T1,10,20 = T1,20,20 + T1,30,20

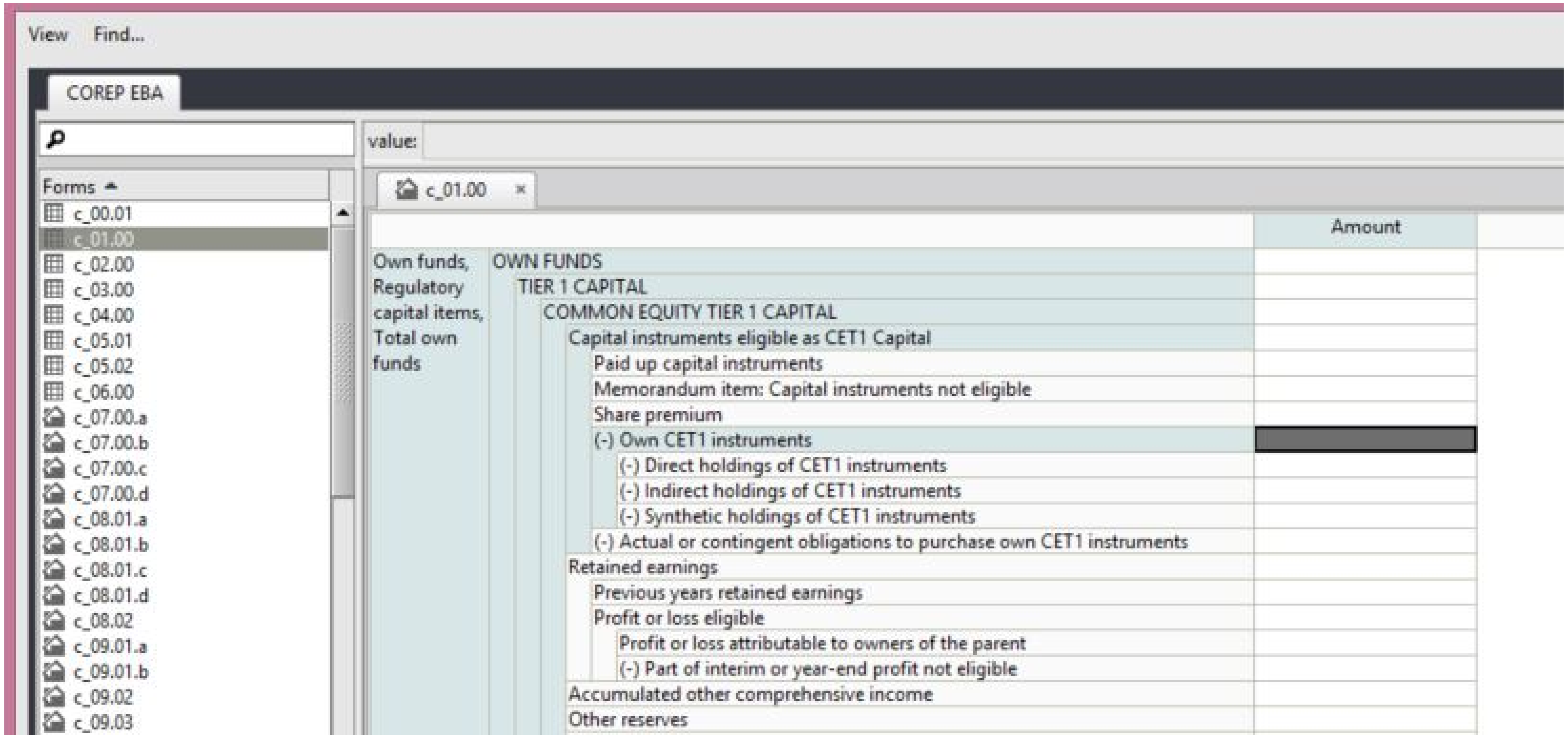
T1,10,10 = T1,20,10 + T1,30,10

IMPORTANT OBSERVATION

In case of 'table addressing’ (Excel-like way, i.e. TABLE,ROW,COLUMN) the notation does not help in revealing logical relations among data.

Visual validation rule editor

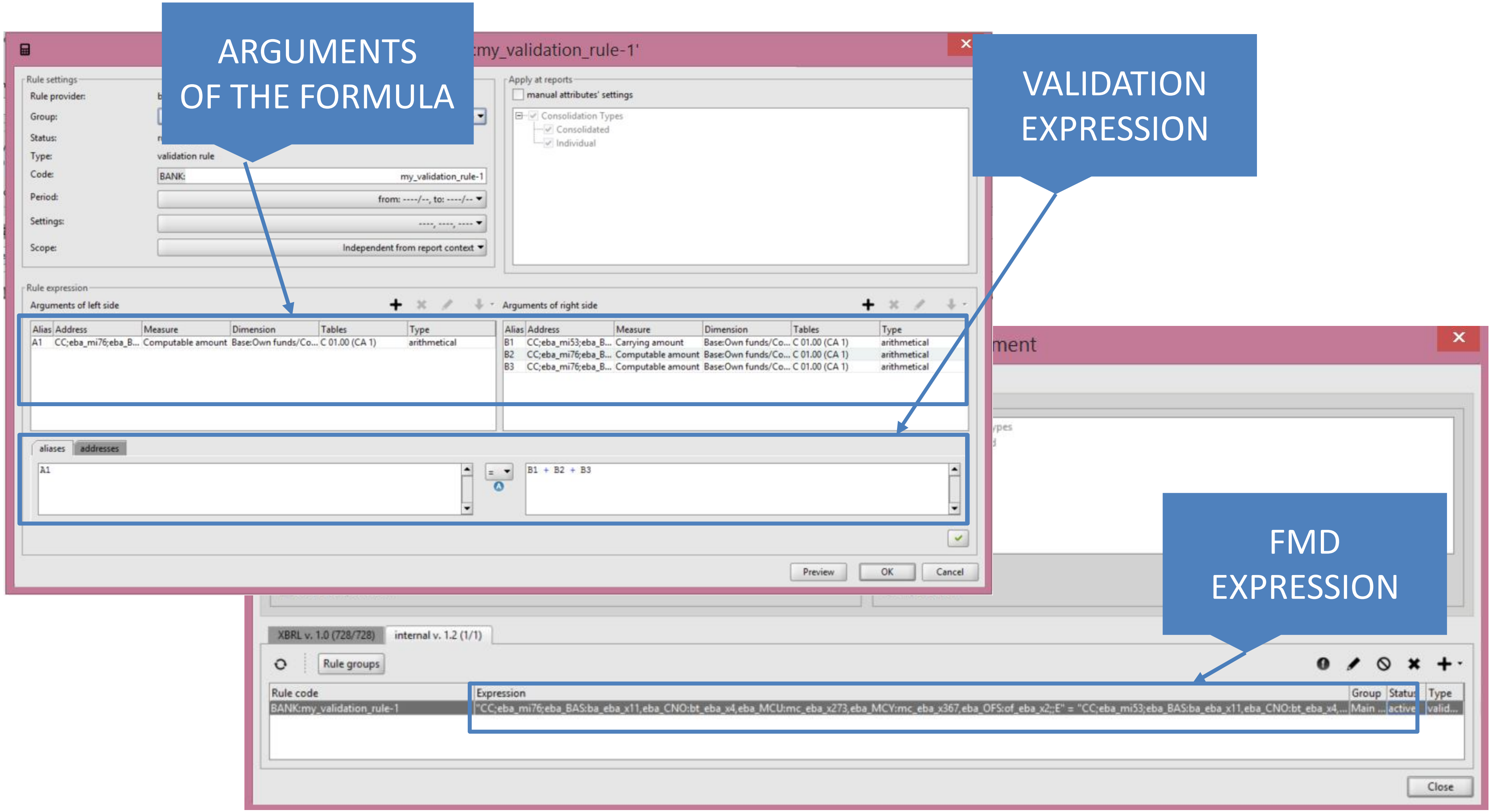
Business users need to work with visual form of reports.
Visual editor should facilitate it by presenting rendered tables and allowing to select fileds which will be arguments of validation rules.



- c_00.01
- c_01.00
- c_02.00
- c_03.00
- c_04.00
- c_05.01
- c_05.02
- c_06.00
- c_07.00.a
- c_07.00.b
- c_07.00.c
- c_07.00.d
- c_08.01.a
- c_08.01.b
- c_08.01.c
- c_08.01.d
- c_08.02
- c_09.01.a
- c_09.01.b
- c_09.02
- c_09.03

		Amount
Own funds, Regulatory capital items, Total own funds	OWN FUNDS	
	TIER 1 CAPITAL	
	COMMON EQUITY TIER 1 CAPITAL	
	Capital instruments eligible as CET1 Capital	
	Paid up capital instruments	
	Memorandum item: Capital instruments not eligible	
	Share premium	
	(-) Own CET1 instruments	
	(-) Direct holdings of CET1 instruments	
	(-) Indirect holdings of CET1 instruments	
	(-) Synthetic holdings of CET1 instruments	
	(-) Actual or contingent obligations to purchase own CET1 instruments	
	Retained earnings	
	Previous years retained earnings	
	Profit or loss eligible	
	Profit or loss attributable to owners of the parent	
	(-) Part of interim or year-end profit not eligible	
	Accumulated other comprehensive income	
	Other reserves	

Visual editor should produce FMD



ARGUMENTS OF THE FORMULA

VALIDATION EXPRESSION

FMD EXPRESSION

:my_validation_rule-1'

Rule settings

Rule provider:	b
Group:	
Status:	r
Type:	validation rule
Code:	BANK: my_validation_rule-1
Period:	from: ----/--, to: ----/--
Settings:	----, ----, ----
Scope:	Independent from report context

Apply at reports

□ manual attributes' settings

- ☑ Consolidation Types
  - ☑ Consolidated
  - ☑ Individual

Rule expression

Arguments of left side

Alias	Address	Measure	Dimension	Tables	Type
A1	CC;eba_mi76;eba_B...	Computable amount	Base:Own funds/Co...	C 01.00 (CA 1)	arithmetical

Arguments of right side

Alias	Address	Measure	Dimension	Tables	Type
B1	CC;eba_mi53;eba_B...	Carrying amount	Base:Own funds/Co...	C 01.00 (CA 1)	arithmetical
B2	CC;eba_mi76;eba_B...	Computable amount	Base:Own funds/Co...	C 01.00 (CA 1)	arithmetical
B3	CC;eba_mi76;eba_B...	Computable amount	Base:Own funds/Co...	C 01.00 (CA 1)	arithmetical

aliases | addresses

```
A1
```

=

```
B1 + B2 + B3
```

Preview | OK | Cancel

ment

XBRL v. 1.0 (728/728) | internal v. 1.2 (1/1)

Rule groups

Rule code	Expression	Group	Status	Type
BANK:my_validation_rule-1	"CC;eba_mi76;eba_BAS:ba_eba_x11,eba_CNO:bt_eba_x4,eba_MCU:mc_eba_x273,eba_MCY:mc_eba_x367,eba_OFS:of_eba_x2;;E" = "CC;eba_mi53;eba_BAS:ba_eba_x11,eba_CNO:bt_eba_x4,...	Main ...	active	valid...

Close

Conclusions

- FMD produces more stable validation rules

  – it is because FMD is based on DATA instead of FORMS

- FMD is not contrary to XBRL Formula

  FMD supports XBRL Formula, because:

  - XF can be produced by FMD2XBRL translator/generator
  - FMD and XF have a lot in common (filters, multidimensiona approach)

- The intermediate layer between natural language and XF is beneffcial for the XBRL project