EBA/XBRL/2015/06

11. August 2015

Filing Rules version 4.1

# EBA XBRL Filing Rules

# Contents


**Change History** 3

**Abbreviations** 5

**Normative references** 5

**Bibliography** 5

**Terms and definitions** 6

**Introduction** 8

Scope of Application 8

Basis in harmonized “European Filing Rules” guidance 8

Target Audience 9

Relationship to Other Work 9

Use of Language 10

**1. Filing syntax rules** 11

**2. Instance syntax rules** 17

Context related rules 19

Fact related rules 22

Unit related rules 27

**3. Additional Guidance** 30

Streaming 32

**Examples** 33

# Change History

| Version | Date | Changes |
|---|---|---|
| 1 | | Baseline |
| 2 | March 2014 | Included missing bibliographic references<br>Reordered auxiliary sections<br>Slight expansion of rules around filing indicators, and inclusion of illustrative examples<br>Further elaboration of the scope of applicability of these rules, highlighting discretion of the competent regulatory authorities as to format and mechanism of reporting (i.e. EBA XBRL not compulsory at first level reporting).<br>Added requirement for pre-registered LEI code to be used as entity identifier in 2nd level remittance, and recommendation of scheme URI to use for LEI (and other) entity codes<br>Emphasize that @xml:lang is not generally required by EBA |
| 3 | February 2015 | Rule 1.6 altered to indicate requirement to include negative filing indicators to indicate non-reporting ("nil" report) for templates that are "expected" to be reported (i.e. for which there is a high likelihood a reporter will need to submit the report, most banks of that kind are expected to have events reported in these templates), in accordance with new business instructions.<br>Minor tweaks to other text referring to filing indicators to clarify where "positive" indicators are being discussed.<br>Annotated instructions regarding monetary values to highlight possibility of explicitly being requested to report monetary values as decimals (without currency units), and resultant effects.<br>*Note that it is considered somewhat likely that rule 3.1 (requiring only a single explicit currency to be reported per instance) may need to be relaxed in future (i.e. if required by future EBA reporting requirements).*<br>Improved layout and phrasing in table at rule 2.19.<br>Wording improvement and removal of comment regarding @decimals and @precision being used on the same fact (which is anyway contrary to XBRL 2.1 spec and so invalid XBRL).<br>Remove reference to MFI ID, or specific national IDs from 3.6 |
| 3.1 | April 2015 | Correction of the scheme identifier for LEIs and pre-LEIs |
| 4 | June 2015 | Rules 1.13 to 1.15 were added. Rules 1.6, 2.16.1 and 2.18 have been updated with significant changes. More guidance is provided by adding rules 2.25 and 3.7 to 3.10. Minor changes have been done for clarification and better understanding.<br>In the section Terms and definitions a new term "byte order mark" was added and the definitions for "fact" and "filing indicators" were improved. |

| Version | Date | Changes |
|---|---|---|
| | | With the establishment of the SSM the wording in the section "Scope of application" has undergone some changes. For the "Filing syntax rules" there were changes applied to rule 1.6. A new sublevel rule 1.6 (d) was incorporated to add a constraint to the declaration of filing indicators in the instance document. Moreover a table was added to clarify the use of the @find:filed attribute for filing indicators for remittance to the EBA. Rule 1.6.3 was reworded to clarify that only valid filing indicators may be used respective to the reported instance. The EBA note for rule 1.7.1 was rewritten for clarification. The new text for rule 1.12 emphasizes that not only resubmissions but also the first submission of a reporter must be complete.<br>In the section "instance syntax rules" a new rule 2.25 has been added to include information about the use of XBRL footnotes in instances. The rephrasing of rule 2.16 together with the improved definition of facts compared to business facts improves the explanation of the occurrence of duplicates. Furthermore the captions in this section have been reworded to be more self-describing. Rule 2.16.1 has been rewritten for clarification. A new sublevel rule 2.18 (c) was added to emphasize that the @decimals attribute used should be realistic. Also a new row was included in the table provided with this rule to indicate the accuracy of millions allowed for the module Funding Plans. The rules 3.1 to 3.3 were moved to the section "instance syntax rules". The section "additional guidance" was extended by four additional rules that should be noted. The filing rule examples at the end of the document were adjusted with more concrete examples and clearer formatting. Moreover examples for the new guidance on namespace prefix declaration were included. Furthermore the file naming structure for remittance to the EBA was added.<br>Section highlighting the impact of "streamable" instance preparation on the application of guidance rules was added.<br>Reference to the CEN (European Normalization Centre) publication about European Filing Rules was updated. |
| 4.1 | August 2015 | Adapted to allow multicurrency reporting as per EBA Single Rulebook QA #1042 - change to rule 3.1 (pp27-28)<br>Correction to LEI URI ("correction" in v3.1 sadly changed the one of the two variations used that was in fact correct into the incorrect form) |

# Abbreviations

| | |
|---|---|
| UML | Unified Modeling Language |
| W3C | World Wide Web Consortium |
| XBRL | eXtensible Business Reporting Language |
| XML | eXtensible Markup Language |

# Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

XBRL 2.1
XBRL Dimensions 1.0
XBRL Registry specification 1.0
XBRL Formula specification 1.0
CWA European Filing Rules

# Bibliography

[CWA] CEN Workshop Agreement 16744-4:2014 Improving transparency in financial and business reporting - Harmonisation topics - Part 4: European Filing Rules (ftp://ftp.cen.eu/CWA/CEN/XBRL/CWA_16744-4_2014.pdf)

[EBA14] Representation in XBRL of the Data Point Model (http://www.eba.europa.eu/documents/10180/632822/EBA+Architecture+for+XBRL+representation+of+DPM.pdf)

[EFM13] EDGAR Filer Manual. U.S. Securities and Exchange Commission (http://www.sec.gov/info/edgar/formdxmltechspec.htm)

[FRIS04] Financial Reporting Instance Standards 1.0 (http://www.xbrl.org/technical/guidance/FRIS-PWD-2004-11-14.htm)

[GFM11] Global Filing Manual (Interoperable Taxonomy Architecture Project) (http://www.ifrs.org/XBRL/Resources/Documents/GlobalFilingManual20110419.pdf)

[IEEE754] IEEE Standard for Floating Point Arithmetic, IEEE Std 754-2008

(http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=4610933)

[RFC 2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
(http://www.ietf.org/rfc/rfc2119.txt)

[SBR13] SBR FRIS rules 2013
(http://www.sbr-nl.nl/fileadmin/SBR/documenten/NT_2013/definitief_03122012/NL-FRIS_NT2013_20121210.pdf)

# Terms and definitions

For the purposes of this document, the following terms and definitions apply.

NOTE XBRL specific terms like context, unit, period, entity, s-equal, v-equal see XBRL 2.1 (http://www.xbrl.org/Specification/XBRL-2.1/REC-2003-12-31/XBRL-2.1-REC-2003-12-31+corrected-errata-2013-02-20.html)

applicable taxonomy
: an XBRL taxonomy recognised to use as a base for filings in a given filing system

byte order mark
: In UTF-8 documents, a sequence of characters (0xEF, 0xBB, 0xEF) that may be used to signal that the characters' are encoded using UTF-8 but, in this particular case, its use is neither required nor recommended by the Unicode consortium

competent authority (CA)
: legally responsible authority

data point
: a Data Point is an information component that is defined by a supervisory authority to be sent in an instance document

  Note: In XBRL a data point is represented by a fact and related dimensional combinations

dimension
: a Dimension is an xs:element in the substitutionGroup of xbrldt:dimensionItem; it relates to the ability to express multidimensional information

entry point
: a schema or linkbase in the applicable taxonomy that represents the filing requirements and gets mentioned in the instance by the filer

fact

a fact is an occurrence in an instance document of an element with a mandatory contextRef attribute and optional attributes like unitRef, decimals, xml:lang or xsi:nil

A business fact is a fact that conveys a business value. Filing indicators facts are not business facts

filer

an entity responsible for submission of a filing

filing

a filing is the fundamental unit of information that is transmitted to a filing system for receipt, validation and acceptance

Note: a filing is conveyed in an XBRL instance document or series of XBRL instance documents

filing indicators

indicate the reporting units (typically templates) reported in the instance

Note: Filing indicators are facts, according to XBRL definitions, but they have special characteristics and are not subjects to the rules defined in this document which cover all other type of facts, called business facts

filing system

a system in which XBRL instance documents are filed, received, checked, stored, analysed and redistributed

reporter

a reporting entity – described by instance(s)

reporting unit

set of facts in a filing which are conceptually either reported or not reported together as a unit

template

a (usually tabular) visible representation of a set of facts, typically identified with/as a single reporting unit

# Introduction

The eXtensible Business Reporting Language (XBRL) specification provides a high degree of flexibility in the creation of XBRL instance documents. Part of this flexibility stems from the nature of the syntax: XML, and part stems from the XBRL specification itself.

## Scope of Application

The European supervisory reporting process is conceptually a multi stage process, first institutions prepare, validate and remit supervisory data to their relevant national authorities ("first level reporting"), where applicable, some data are sent to a supranational authority, and subsequently those authorities remit data to the European Banking Authority ("second level reporting").

These filing rules represent a collection of additional rules and guidance specifically **applicable to the remittance** of XBRL instances for reporting entities in scope of relevant EBA regulations (e.g. banks) regulatory filings **by relevant national and supranational authorities to the European Banking Authority**.

Focussed on the preparation of XBRL instance files, rather than details of the mechanics of report submission/data collection these rules constrain the full flexibility of XBRL, to enable effective interaction between transmitter and recipient/consumer of regulatory reports.

The listed filing rules are influenced by the EBA Taxonomy Architecture in cases where the instance creation is affected.

This document was reviewed by a group of national experts in order to clarify any misleading formulation of rules and contribute to the pan-European harmonisation of the filing rules. The rules as stated in this document are those enforced on the second level of reporting (to EBA). In the case of supervisory authorities adopting these rules but with adaptations, for example changing preferences or guidance expressed by the EBA instead into obligations on the first level of reporting such variations will be communicated to the reporter by the respective supervisory authority.

**N.B these rules are not necessarily those that are applicable at the level of reporting by individual institutions or groups of institutions, nor do they address the entire scope of the reporting process. Guidance should be sought from the reporter's competent authority as to their reporting format and requirements for that reporting.**

## Basis in harmonized "European Filing Rules" guidance

In order to promote and enhance interoperability, these rules are largely drawn from the document CWA 16744-4:2014 European Filing Rules, promulgated by the European Normalization Centre (CEN), which *"represent a collection of recommendations to be seen as guidance to be implemented in the European*

*supervisory reporting process"*. This document should be read in conjunction/comparison with that CEN document.

### Numbering of rules

Please note that the rules are not necessarily numbered in sequential order. For ease of comparison, rules were originally numbered as per their numbering in the CEN document hence some numbers were omitted where the corresponding CEN rule was not applicable/not included. To aid identification and comparison between revisions of this document where possible the initial numbering of specific rules is retained, hence rules may be out of order, or in different sections from that implied by their numbering.

## Target Audience

Although primarily addressed to those (mostly technical staff) within the national and supranational authorities responsible for preparation or submission of XBRL instance files directly to the European Banking Authority, these filing rules will also be of value to individual reporters (i.e. financial institutions or groups of institutions) reporting to those authorities which may utilise the EBA filing rules or XBRL format, or derivatives of them.

This document is intended for a technical audience and assumes that the reader has a working knowledge of the XBRL 2.1 and the XBRL Dimensions 1.0 Specifications and has a basic understanding of XML, Namespaces, and XML Schema.

To readers with XML knowledge, many of the guidelines in this document will be familiar however, others originate from features that are XBRL-specific and therefore the reasoning behind them may be less obvious.

## Relationship to Other Work

This document should be read in conjunction with the EBA Taxonomy Architecture. [EBA14]

The guidelines in this document pertain to XBRL filings. Parts of this document reiterate for expository clarity certain syntactic and semantic restrictions imposed by XBRL, but this document does not modify XBRL. In the event of any conflicts between this document and XBRL, XBRL prevails. This document does place additional restrictions beyond those prescribed by XBRL.

The rules are based closely on the recommendations of the CEN Workshop Agreement on European filing rules developed by the CEN WS/XBRL project (http://cen.eurofiling.info/).

To ease the understanding by software developers implementing these guidelines in their reporting system, an UML model is included to show the relationships between the different XBRL objects mentioned in this document.

Some of the filing rules are accompanied by constraints defined in the Object Constraint Language (OCL). OCL is part of UML and allows description of constraints based on the UML objects of the class model. OCL is not a

programming language; it just supports the definition of technical specifications. OCL eases the understanding of the rules by using a formal language to provide an unambiguous and consistent description.

For harmonization and explanatory purposes, where similar filing rules are used in other jurisdictions, references are indicated.

## Use of Language

The use of language in this document follows that specified in [RFC 2119], in summary:

The use of “MUST” implies an obligation, and the preparation of instance files not following these rules will generally result in rejection of the instance file.

The use of “SHOULD” implies an indication of preference or best practice, but also a degree of tolerance, following the principle of “comply or explain”). The rule must be respected unless there are good reasons not to do so. Failure to follow the rule will not result in rejection of an instance file by EBA.

The use of “MAY” implies permission, and describes actions that can be taken or constructs that can be used, but that are not required. Utilising these options will not result in rejection of an instance file.

XML attribute names are preceded by the "@" character in this document, as in XPath syntax.

# 1. Filing syntax rules

### 1.1 — Filing naming

Common practice is to use the extension .xbrl for instance documents. Detailed file naming requirements should be confirmed with the intended recipient of an instance file. Credit institutions should confirm with their relevant supervisory authority for reporting. The file naming convention to be used by CAs for remittance to the EBA can be found in the appendix.

### 1.4 — Character encoding of XBRL instance documents

The XML and XBRL specifications place no restrictions on the character encodings that may be used in instance documents. In order to avoid using a character encoding that is not supported by a receiving processor, all instances must use the UTF-8 character encoding (regardless of with or without BOM).

**XBRL instance documents MUST use "UTF-8" encoding. [GFM11, p. 11]**

```
context xmlDocument inv: self.encoding = 'UTF-8'
```

### 1.5 — Taxonomy entry point selection

A taxonomy is loaded through a reference to one or more URLs. Although technically a user can reference any file in the taxonomy, a taxonomy publisher will typically nominate specific URLs which are intended to be referenced by users of the taxonomy. These URLs are called entry points, and allow users to import the correct modules from the taxonomy, with different modules including different templates and different associated validation rules.

The EBA taxonomy defines multiple specific entry points ("modules"), suitable for different reports. The taxonomy also contains other XML schemas, these are not to be treated as entry points. Through the 'filing indicators' it is communicated which tables are reported in an instance.

**(a) Reporting entities MUST reference only one entry point schema ("module", link:schemaRef element), as specified in the applicable taxonomy, per XBRL instance. [SBR13, p. 6]**

**(b) The schemaRef element MUST refer to a URL appropriate to the module and the reference date of an instance, drawn from the list of entry points published by the EBA[1].[EBA14]**

### 1.6 — Filing indicators

Each reported fact in a filing is assigned to one or more reporting units (typically "templates") of the specific domain of reporting.

[1] or competent authority for first level reporting.

A filing indicator element (filingIndicator), grouped (potentially with other such elements) within a containing element (fIndicators), containing a code associated with a particular reporting unit, is used to indicate the intention of a reporter to report that reporting unit, or to indicate the intention *not* to report that reporting unit (see example under the heading "Filing indicator examples" for illustration). Filing indicators also trigger the appropriate taxonomy formulae checks. Missing filing indicators can lead to inconsistencies because facts for unindicated reporting units might not be validated.

**(a) Reported XBRL instances MUST include appropriate positive (i.e. either with @find:filed="true" or without @find:filed attribute) filing indicator elements to express which reporting units ("templates") ARE intended to be reported in the instance.**

**(b) Instances MAY include appropriate negative (i.e. with @find:filed="false") filing indicator elements indicating reporting units which are intended NOT to be reported in the instance.**

**(c) Negative filing indicators MUST be included when a reporting unit is deliberately not reported[2] which is potentially expected by the EBA to be contained in that instance (e.g. due to the reporter having no relevant transactions or positions to report, or on that occasion falling outside a relevant threshold for the reporting of the unit), in order to express the intention of the reporter not to report definite values for said template.[3]**

**(d) The context referenced by the filing indicator elements MUST NOT contain xbrli:segment or xbrli:scenario elements.**

Selected example scenarios:

| **Scenario** | **@find:filed attribute of filing indicator for template** | **Causes rejection** |
|---|---|---|
| A template is included in the reported instance with facts | true / absent | No |
| A template is included in the reported instance, but no associated facts are *explicitly* reported (i.e. included in the XBRL instance). | true | No (all facts for template may be assumed to be zero, see 1.7) |
| A template is explicitly *not* reported in the instance due to<br>a. reporter having no relevant transactions or positions to report<br>b. on that occasion falling outside a relevant threshold for the reporting of the unit | false | No |
| Fact values for a template are reported, at least some of which are not also part of another template which has a positive filing indicator | false | Yes (violation of rule 1.7.1) |

[2] For which it is actually legitimate to not report definite values (this is not the case for all reporting units).

[3] CAs (and filers) should aim to comply with this requirement immediately; however the EBA does not intend to strictly enforce this rule earlier than the reporting cycle for reports with reference date Dec 2015 (i.e. no earlier than early 2016).

| Scenario | @find:filed attribute of filing indicator for template | Causes rejection |
|---|---|---|
| A template is not reported, but facts "appearing on that template" *are* reported, they are all contained in other template(s) which *are* indicated as reported in the instance | false | No (see EBA advice to 1.7.1) |
| A template is reported. Multiple filing indicators with the same code are included in the instance. | n/a | Yes (violation of rule 1.6.1) |

### 1.6.1 — Multiple filing indicators for the same reporting unit

There is no benefit in filing several filing indicators for the same reporting unit. Inconsistent occurrences might occur (different values of @find:filed attribute).

**Reported XBRL instances MUST contain only one filing indicator element for a given reporting unit ("template").**

### 1.6.2 —Filing indicators in several tuples

Reporting filing indicator elements spread across several separate fIndicators tuples is a more complex approach than using a single containing element, and is likely to be more complex to handle by receivers.

However this construction may be useful for generating large instances (generation in a single pass or *streaming*), by allowing e.g. a tuple containing a single filing indicator to immediately precede (or follow) the data items for each reporting unit.

**For flexibility, reported XBRL instances MAY include different filing indicators in several separate fIndicators tuple elements, for simplicity this SHOULD in general be avoided where not necessary.**

### 1.6.3 – Filing indicator codes

As stated in the EBA Taxonomy Architecture the values of filing indicators to be used are indicated by label resources associated with the tables in the XBRL taxonomy. The value used should be exactly as indicated.

**The values of filing indicators MUST only be those given by the label resources with the role http://www.eurofiling.info/xbrl/role/filing-indicator-code applied to the relevant tables in the XBRL taxonomy[4] for that reporting module (entry point). Filing indicator values must be formatted correctly (for example including any underscore characters).**

[4] N.B. equivalent information is available in the EBA DPM Database.

### 1.7 — Implication of no facts for an indicated template

If a positive filing indicator is given in the XBRL instance, appropriate consistency checks may be processed by the recipients' reporting system. If no facts appear for an indicated template, the filing may well be rejected because the system requires an appropriate, coherent set of fact values for the checks.

If there are no facts reported that match a template indicated with a positive filing indicator, this conveys that the template is intended to be explicitly reported and every cell on that template may be considered (i.e. when applying validation checks) as equivalent to zero (for numeric value) or blank (for non-numeric), not that the template as a whole is intended to be unreported[5]. In practice, this is unlikely to be the intent of a filer, and may indicate an error in instance preparation.

**(a) Reported XBRL instances MUST include appropriate positive filing indicator elements to express which reporting units ("templates") are intended to be reported in the instance**

**(b) Reported XBRL instances MUST NOT include positive filing indicator elements indicating a reporting unit is filed (i.e. @find:filed=true, or no @find:filed attribute) for reporting units which are NOT intended to be reported in the instance.**

#### 1.7.1 — No facts for non-indicated templates

**Reported XBRL instances MUST NOT include business facts which are not contained in any of the reporting units ("templates") indicated by filing indicators as reported.**

*EBA Advice: Note that a single fact may notionally appear in several reporting units ("templates") - i.e. cells from several templates may represent the same data item, which would be transmitted as just a single fact. It may be the case that only some of these templates are reported in an instance, and others are not. In these situations the presence of such a fact which is part of a reported template but which would also be part of an unreported template is NOT a breach of these rules – i.e. they do not require that all templates containing an reported fact are indicated as reported, just that all reported facts appear in at least one template which is indicated as reported.*

### 1. 9 — Valid XML-XBRL

In order to increase the likelihood that instance documents pass validation, filers must validate their compliance with the XBRL 2.1 and Dimensional 1.0 specification prior to submission.

**Instance documents MUST be XBRL 2.1 and XBRL Dimensions 1.0 valid. [EFM11, p. 6-8]**

[5] Which would be indicated with a *negative* filing indicator – and would indicate that any facts associated to the reporting unit (which are not anyway reported in the instance as part of another reporting unit with an associated positive filing indicator) are to be considered "unknown"

### 1.10 — Valid according to the defined business rules

XBRL allows the definition of business validation rules which can be discovered by XBRL software when opening the respective module referenced in the instance document. These business validation rules are applied on the content of the instance document to check the data quality.

(a) **Instance documents MUST be valid with regards to the validation rules as defined in the taxonomy (using XBRL formula), and discoverable from the referenced entry point, with the exception of any validation rules indicated as either deactivated or not mandatory to comply with in material published by the EBA.**

(b) **Instance documents MUST also be valid with regards to validation rules published in the applicable ITS, including those not implemented by the validation rules as defined in the taxonomy (using XBRL formula), again with the exception of any validation rules marked as deactivated or non-mandatory in material published by the EBA.**

```
context Instance::isValidationValid() : Boolean post: result = true
```

### 1.11 — Taxonomy extensions by reporters

XBRL Taxonomies can be extended by anybody with the proper technical knowledge. Filings to European Banking Authority are 'closed form' i.e. all data points allowed by the regulator are in the taxonomy. There can be no extension of the taxonomy by reporters to report more (or less) data points to the supervisor. However some CA's may extend European taxonomies. For reporters the combination of base and extension taxonomies is regarded as a single taxonomy. (also see 1.5)

**Instances MUST reference only the taxonomy entry points specified by the relevant authority (i.e. reporters MUST NOT reference their own extension taxonomies).**

### 1.12 — Completeness of the instance

In case corrections are needed on filings that already have been sent, it is required to resubmit the complete filing, rather than partial data with just the corrected facts. Non-complete submissions could lead to invalid instance documents (according to either XBRL 2.1, XDT 1.0 or appropriate Formulae), might raise conflicts with already processed data in the reporting system of the receiver, and may lead to significant errors if sender and receiver disagree as to the list and sequence of historical submissions.

**Instances MUST contain the full report, even in the case of resubmission of an amendment – no content/values from previous instances may be assumed.**

### 1.13 — Standalone Document Declaration

The standalone document declaration in the XML declaration (e.g.: <?xml version="1.0" encoding="UTF-8" standalone="yes" ?> or <?xml version="1.0" encoding="UTF-8" standalone="no" ?>) is only relevant for XML documents using a DTD. This information has no meaning for XBRL instances. and may cause problems to some software component.

**XBRL instance documents SHOULD NOT use the XML standalone declaration.**

### 1.14 — @xsd:schemaLocation and @xsd:noNamespaceSchemaLocation

@xsd:schemaLocation and @xsd:noNamespaceSchemaLocation are attributes defined in the XML Schema specification that are used to indicate where the schema to be applied to the XML document may be found. Since the XML Schema to be used in XBRL instances is defined by the link:schemaRef element, this attribute may introduce ambiguity.

**@xsd:schemaLocation or @xsd:noNamespaceSchemaLocation MUST NOT be used.**

### 1.15 — XInclude

The XInclude specification provides a way to embed an XML document in another one, by using xi:include elements. This possibility is rarely supported by XBRL processors.

**XBRL instance documents MUST NOT use the XInclude specification (xi:include element).**

# 2. Instance syntax rules

### 2.1 — The existence of xml:base is not permitted

XBRL processors interpret this attribute differently, and there is no semantic need for this attribute.

*XML-XBRL:* The attribute xml:base may be inserted in XML documents to specify a base URI other than the base URI of the document or external entity.

**The attribute @xml:base MUST NOT appear in any instance document. [EFM13, p. 6-7]**

```
context xmlDocument inv:
self.element->select(xml:base)->isEmpty()
```

### 2.2 — The absolute URL has to be stated for the link:schemaRef element

The taxonomy which is used for an XBRL report is identified by the URL(s) referenced by link:schemaRef elements. Although it is often convenient to work with local copies of the relevant taxonomies, it is important that link:schemaRef elements resolve to the published entry point locations. XBRL software typically provides functionality to "remap" references to URLs of published entry points to local copies of the taxonomy.

**The link:schemaRef element in submitted instances MUST resolve to the full published entry point URL (absolute URL).**

### 2.3 — Only one link:schemaRef element is allowed per instance document

Under the XBRL standard, the element link:schemaRef can occur several times in an instance. In the EBA taxonomy however only a single entry point schema must be referred to in any instance. This entry point will specify all required data points, and is used to reference a particular report type.

**Any reported XBRL instance document MUST contain only one xbrli:xbrl/link:schemaRef element.**

```
context Instance inv: self.SchemaReference->size() = 1
```

### 2.4 —The use of link:linkbaseRef elements is not permitted

Entry points will be defined by means of a schema. There is no use for link:linkbaseRef elements.

**Reference from an instance to the taxonomy MUST only be by means of the link:schemaRef element. The element link:linkbaseRef MUST NOT be used in any instance document.**

### 2.5 — XML comments and documentation are ignored by EBA

Comments inside the instance that do not get reported as a fact will be ignored by the EBA.

**Relevant business data MUST only be contained in contexts, units, schemaRef and facts.**

**A comment MUST not have any impact on the content of a report.**

Comments may be present in instances sent to EBA but their content will be ignored.

### 2.25 — XBRL footnotes are ignored by EBA

Footnotes may be supported and used by some CAs, but might not be accepted by other CAs. Footnotes within an instance will be ignored by the EBA.

**Relevant business data MUST only be contained in contexts, units, schemaRef and facts.**

**A footnote MUST not have any impact on the regulatory content of a report.**

Footnotes may be present in instances sent to EBA but their content will be ignored.

## Context related rules

### 2.6 — The length of the @id attribute should be limited to the necessary characters

The @id attribute is meant as a unique technical key within a XML document. Conveying semantics in the @id attribute will likely be lost when the XML content is processed, e.g. stored in a database (which generally works with database specific surrogate keys), any semantics are unlikely to be available to a (human) consumer of the instance data. Even though there is no limitation on the length of an id attribute it is recommended to keep it as short as possible.

**Semantics SHOULD NOT be expressed in the xbrli:context/@id attribute. The values of each @id attribute SHOULD be as short as possible.**

### 2.7 — No unused or duplicated xbrli:context nodes

Unused contexts (contexts which are not referred to by facts) clutter the instance and add no value to either regulator or reporter [GFM11, p. 12].

**(a) Unused xbrli:context nodes SHOULD NOT be present in the instance. [FRIS04]**

**(b) An instance document SHOULD NOT contain duplicated context, unless required for technical reasons, e.g. to support XBRL streaming.**

```
context Context inv: self.Fact.allInstances()->notEmpty()
```

### 2.8 — Identification of the reporting entity

The xbrli:identifier element combined with the @scheme attribute allows the identification of the reporting entity by the receiver. The @scheme provides a URI which uniquely identifies the type of identifier used in the xbrli:identifier node (see section 3.6 LEI and other entity codes).

**(a) Instances MUST use a @scheme attribute that is prescribed by the receiving supervisor. [GFM11, p. 11]**

**(b) Instances MUST use an identifier acceptable to the receiving regulator (likely to be one recognized in their reporting system), and that corresponds to the @scheme attribute used. [GFM11, p. 11]**

**(c) For remittance of data by CA's to the EBA, the entity identifier used should be a Legal Entity Identifier code, and must have been registered with the EBA by the CA prior to remittance.**

### 2.9 — Single reporter per instance

There can only be one reporter of an instance. Even if the content of the instance deals with a group of companies, there is only one entity reporting the instance to the regulator.

**All xbrli:identifier content and @scheme attributes in an instance MUST be identical. [EFM13, p. 6-8]**

```
context Context inv: self.Identifier.allInstances()->forAll(i1, i2|
i1 = i2 implies i1.value = i2.value)
```

### 2.10 — The xbrli:period date elements reported must be valid

The xbrli:startDate, xbrli:endDate and xbrli:instant elements all have data type which is a union of the xs:date and xs:dateTime types. EBA will only allow periods to be identified using whole days, specified without a timezone.

**All xbrli:period date elements MUST be valid against the xs:date data type, and reported without a timezone. [GFM11, p. 16]**

### 2.11 —The existence of xbrli:forever is not permitted

The extreme version of duration is 'forever'. The XBRL specification has created this to solve problems with dates starting 'at the beginning' and ending 'never'. E.g. the name of the founder of a company has in general no end date. The EBA is only interested in data for the reported time segment, that has a defined starting and ending date.

**The element 'xbrli:forever' MUST NOT be used. [GFM11, p. 19]**

```
context Context inv: self.Period.forever->isEmpty()
```

### 2.13 — XBRL period consistency

XBRL requires all facts to be associated with a "period" (either a duration or instant of time). Where there are multiple relevant date/period like concepts related to a fact (as is often the case), it may be unclear which of these concepts is expressed by the XBRL period.

A common approach is to associate the XBRL period with some variation of a "real-world date of the event" for a fact. Use of varying "event" dates for facts in a regulatory reporting instance may however lead to complexity, confusion, and practical difficulties (e.g. for selecting facts for table linkbase axes, validating dates, identifying related facts etc.), particularly where the relationship between reporting periods and current and prior conceptual dates (e.g. accounting periods) is unclear, complex, and/or time-varying, such as in jurisdictions allowing non-calendar financial periods.

For simplicity therefore, the European Banking Authority has instead chosen to associate the "reference date" of an instance with the XBRL period concept.

Logical distinctions between other date-like aspects of a fact, such as the "event date, "applicable period", "date offset from reporting date" are conveyed via dimensional attributes of a fact.

**All xbrl periods in a report instance MUST refer to the (same) reference date instant. All xbrl periods MUST be instants.**

```
context Context inv: self.Period.allInstances()->forAll(p1, p2| p1 =
   p2 implies
p1.instant = p2.instant)
```

### 2.14 — The existence of xbrli:segment is not permitted

The XBRL Dimensions specification allows taxonomies to specify dimensions for use within either the segment or the scenario of the context. For consistency reasons and simplification of processing, EBA only uses the xbrli:scenario element.

**xbrli:segment elements MUST NOT be used.**

### 2.15 — Restrictions on the use of the xbrli:scenario element

The xbrli:scenario element MUST NOT be used for anything other than for explicit or typed members. Custom reporter XML schema content may create problems with the regulatory system.

*XML-XBRL:* The XBRL specification allows xs:any content. This means that all XML schema content can be stored (not just XBRL Dimensions).

**If an xbrli:scenario element appears in a xbrli:context, then its children MUST only be one or more xbrldi:explicitMember and/or xbrldi:typedMember elements, and MUST NOT contain any other content. [EFM13, p. 6-8]**

# Fact related rules

### 2.16 — Duplicate (Redundant/Inconsistent) facts

Facts are business duplicates of each other in the reporting sense if they notionally convey answers to precisely the same question. At best such duplicates are simply redundant (where they are truly semantically equivalent), at worst they are inconsistent or contradictory.

An instance document must not have duplicated business fact items. Item X and item Y are "duplicate facts" if and only if all the following conditions apply:
1. X is not identical to Y (not exactly the same XML node[6]), and
2. The element local name of X is S-Equal to the element local name of Y, and
3. X and Y are defined in the same namespace[7], and
4. X is P-Equal to Y[8], and
5. X is C-Equal to Y, and
6. X is U-Equal to Y, and
7. X and Y are dimensionally equivalent (d-equal in all dimensions of each of X and Y)[9], and
8. If X and Y are string items, they also have S-Equal xml:lang attributes[10].

Inconsistent facts are duplicates that are not V equal.

*XML-XBRL:* Duplicate facts are XML-XBRL syntax valid. However (whether or not their values are different) the semantic meaning may be unclear.

**Instances MUST NOT contain duplicate business facts. [FRIS04],[EFM13, p. 6-10]**

### 2.16.1 — No multi-unit fact sets

Two facts which differ only by unit are not technically duplicates. Indeed there might be situations in which, for example, the natural answer to a question is a bundle of set of values in several currencies (e.g. £4, $3, €3). However there is clearly a significant potential for confusion with such reporting - e.g. are the different facts supposed to be alternatives ($4 or £3), equivalents ($4 = £3), to be taken as a set ($4 and £3), or just a mistake.

In order to avoid any such doubt or confusion, reporting of "the same fact"[11] in more than one unit is not allowed in EBA reporting.

**Instances MUST NOT contain business facts which would be duplicates were their units not different.**

[6] This apparently trivial condition is stated here since it is sometimes relevant, e.g. when X and Y are the result of different XPath conditions
[7] 2&3 may loosely be considered to mean "refer to the same primary item"
[8] Somewhat irrelevant in the EBA context, since all data fact items should be reported in a single root element, and no tuples are used to report data facts.
[9] 1-7 effectively mean "refer to the same data point". Note that this definition is very similar to, but not the same as the definition of a "duplicate item", notably it does not require that facts be U-equal to be considered "duplicate facts".
[10] Multiple string facts that would otherwise be duplicates are in principle acceptable in the EBA reporting context if each has a distinct effective xml:lang attribute (i.e. if they are translations of each other). Note that the following elements do NOT make two facts non-duplicate if they differ (or if they are the same!): **value**, decimals, xml:lang for non-strings.
[11] i.e. facts which meet all the conditions in rule 2.16 except point 6.

### 2.17 — The use of the @precision attribute is not permitted

The XBRL standard provides two methods of communicating the precision of a numeric fact: @precision and @decimals attributes. Humans seem to have an easier time reading a document that uses the decimals attribute, probably because in most uses the decimals value is likely to be one of a limited set e.g. 2, 0, -3, -6, -9 or INF (and often the same for all/many facts). Moreover, given a decimals value the precision can always be computed, but this is not symmetric.

**@decimals MUST be used as the only means for expressing precision on a fact. [FRIS 2.8.1.1, EFM13, p. 6-12]**

### 2.18 — Interpretation of the @decimals attribute

The @decimals attribute indicates the accuracy of the reported fact value. If a numeric fact has an @decimals attribute with the value n then it is considered to be "correct to n decimal places". Leading zeros and trailing digits should be compact and appropriate to the reported value.

The EBA will interpret the @decimals attribute on reported data as specifying that the absolute difference between the true value of the number as known to the reporter and its reported lexical representation (known as the "absolute error" of the representation - $e_{abs}$) is less than or equal to $0.5 \times 10^{-n}$. Reporters must prepare submitted reports consistently with this interpretation[12].

The EBA XBRL validation rules use interval arithmetic for validation. To best enable XBRL Formula calculations to be performed on instance values for validation purposes, preferably no truncations or rounding or any other kind of change should be applied to the reported lexical representation of the numeric facts in the instance. See the explanatory RFC at http://www.xbrl.org/RFC/PDU/PWD-2008-10-09/PDU-RFC-PWD-2008-10-09.html. Note however that if numbers are for any reason rounded, they MUST be rounded as per the XBRL 2.1 specification (i.e. [IEEE-754] 4.3.1 Rounding-direction attributes to nearest, roundTiesToEven), and as above the @decimal attribute must accurately represent the relationship between the reported and unrounded values.

**(a) The accuracy of a numeric fact MUST be expressed using @decimals**

**(b) There SHOULD be no truncation, rounding or change to the original fact value, which should be reported as known.**

**(c) The reported accuracy (@decimal attribute) of a numeric fact SHOULD be a realistic indication of the accuracy to which the lexical representation represents the true value. In particular it SHOULD NOT be excessively high.[13]**

*EBA Note: In particular, if numbers are truncated or rounded for reporting, they should not be "adjusted" so that they "appear" to be visually consistent (i.e. so that they "foot" or "cast"), but should instead be simply reported with the appropriate @decimals value – the validation checks will take into account the declared accuracy to determine if reported values are (could be) valid.*

[12] See also the explanation of "Correct to n decimal places" given in the (now superseded) 2008-07-02 Errata version of the XBRL 2.1 specification at http://www.xbrl.org/Specification/XBRL-RECOMMENDATION-2003-12-31+Corrected-Errata-2008-07-02.htm#_4.6.7.2

[13] E.g. @decimal values of greater than 2 would generally be inappropriate for calculated "monetary" values resulting from e.g. multiplications or divisions, "INF" is often unlikely to be appropriate for calculated values etc.

| Accuracy Requirements | | | |
|---|---|---|---|
| **Data Type** | **Decimals attribute** | **Note** | **Representation** |
| Monetary[14] | >= -3,<br>>= -6 for the module Funding Plans only | | 42563.26 |
| Percentage | >= 4 | Must be expressed as a ratio in instances – i.e. typical values between 0 and 1 | 0.1234 (=12.34%) |
| Integer | 0 | Must of course be reported without any decimal part | 126 |

N.B. INF (meaning exact as written) is of course acceptable for the decimal attribute of all numeric types.

*EBA Note: This, combined with the definition of the @decimals property, means that in general monetary values must not be* truncated *to thousands (since the reported value might then be up to 1000 from the true value, which is more than the 500 implied by @decimals=-3, requiring instead decimals=-4 to be consistent), but may be* rounded *(i.e. to nearest value) to thousands[15].*

The decimals attribute is not a scale factor. The decimals attribute is not a formatting code; it does not indicate that the digits in the instance must subsequently be presented to a user in any particular way.

The @decimals attribute influences how numbers are interpreted. Use the following table to select the correct value of the @decimals attribute for a fact so that it corresponds to the accuracy to which the value is known.

| Accuracy of the amount | Value of decimals attribute |
|---|---|
| Absolutely exact monetary, percentage or other amount | INF |
| Accurate to millions | -6 |
| Accurate to thousands | -3 |
| Accurate to hundreds | -2 |
| Accurate to units | 0 |
| Accurate to cents | 2 |
| Accurate to a hundredth of a percentage point (i.e. a *basis point*) | 4 |

[14] N.B. Also applies to facts representing monetary values that are specified (via their primary item) to be reported as currency-less decimal values.

[15] For the funding plans module the equivalent observation regarding truncating vs rounding to millions applies.

Examples: The table below illustrates correct use.

| Data | Reported Value | Value of @decimals attribute | Range of value considered in interval arithmetic |
|---|---|---|---|
| A percentage (ratio) of (exactly) 46% | 0.46 | INF[16] | 0.46 |
| A profit margin of 9.3% (minimum accuracy) | 0.093 | 4 | 0.09295 to 0.09305 |
| Monetary amount “in millions” | 1534512 | -6 | 1034512 to 2034512 |
| Monetary amount “in thousands” | 117822 | -3 | 117322 to 118322 |
| Monetary amount “in hundreds” | 124265 | -2 | 124215 to 125215 |
| Monetary amount, accuracy of “units” | 100205.23 | 0 | 100204.73 to 100205.73 |

[EFM13, p. 6-28], [GFM11, p. 45f.]

*EBA NOTE: For clarification - this guidance applies only to the representation of the values in the transmission XBRL instance file, it of course places no constraints on the display of information to any user or preparer of the data. Tools may choose to display values however they (and their user's) desire, so long as when instance files are produced the canonical representation is used.*

### 2.19 —Guidance on use of zeros and non-reported data

Data could be reported with a non-zero value, as zero or unreported.

**The @xsi:nil attribute MUST NOT be used in the instance.**

The table below shows the different possible scenarios:

[16] N.B. it is only appropriate to use “INF” where the true value is known to be absolutely precisely the value reported, as written. E.g. monetary balances in cents, or chosen rather than calculated percentages.

| | | | |
|---|---|---|---|
| **Reported Zero or Non-zero value** | e.g. <eba_met:mi53 unitRef="uEUR" decimals="2" contextRef="c2">1025.25</eba_met:mi53> | | The value of the fact is known. |
| **Reported nil value** | e.g. <eba_met:mi53 unitRef="uEUR" contextRef="c2" @xsi:nil="true" /> | | MUST NOT be used |
| **Missing fact** | The fact doesn't appear in the instance. | Template including this fact is reported | The value is treatable as equivalent to zero (if numeric fact) or empty (if non-numeric) by the recipient. |
| | | No template including this fact is reported | The value is "unknown" to the recipient. |

Inapplicable information need not be included in an instance, i.e. inapplicable facts MAY be left out.

*EBA Note: For validation purposes, unreported numeric facts belonging to a template indicated as "reported" by an instance (using filing indicators) will be treated as equivalent to zero in the evaluation of certain rules – see the details of individual rules.*

*EBA Note: Zero values SHOULD, preferably, be explicitly reported where they are interesting supervisory reporting information. "Uninteresting zeros" (i.e. large swathes/permutations of trivially zero or simply inapplicable information, for example the large bulk of countries, currencies, lines of activity etc. in which a reporter has nothing relevant to report) SHOULD NOT be reported for obvious practical reasons.*

### 2.20 — Information on the use of the xml:lang attribute

The language used on string based facts may need to be identified. This can be done by declaring the @xml:lang on the xbrli:xbrl element just once, or on every string based fact individually. No restrictions are placed on language used in reporting string facts (such as entity names), however some strings are required to have specific values by the ITS which are not language specific, and should be the same whatever language is marked.

In practice, the @xml:lang attribute is in general not required in instances remitted to the EBA and may be omitted. It is compulsory to use the attribute in the specific case of distinguishing otherwise duplicate string facts, where an individual fact is reported in more than one language (i.e. with translation). This is expected to be a relatively rare situation as there is no requirement to submit translations of string facts.

## Unit related rules

### 2.21 — Duplicates of xbrli:xbrl/xbrli:unit

Units are equivalent if they have equivalent measures or equivalent numerator and denominator. Measures are equivalent if their contents are equivalent QNames. Numerators and Denominators are equivalent if they have a set of equivalent measures. Duplicated units do not express extra semantics and potentially disturb comparison of facts that point to any of the duplicated occurrences [EFM13, p. 6-10].

> **An XBRL instance SHOULD NOT, in general, contain duplicated units, unless required for technical reasons, e.g. to support XBRL streaming.**

### 2.22 — Unused xbrli:xbrl/xbrli:unit

Unused units (units which are not referred to by facts) clutter the instance and add no value to either supervisor or reporter.

> **An XBRL instance SHOULD NOT contain unused xbrli:unit nodes. [FRIS04]**

### 2.23 — Reference xbrli:unit to XBRL International Unit Type Registry (UTR)

XII has released a standard numeric data type registry: it has a schema with numeric type declarations, and each numeric data type is associated with consistent unit declaration measures, numerators and denominators. Use of this registry that contains all the usual units eases implementation in software and simplifies validation (http://www.xbrl.org/utr/utr.xml ).

> **xbrli:unit children MUST refer to the XBRL International Unit Type Registry (UTR). [EFM13, p. 6-17]**

### 2.24 —Report of the actual physical value of monetary items (see also 3.3)

Facts that represent amounts in any currency will be of an item that is derived from xbrli:monetaryItemType, which must follow the restriction in XBRL 2.1, section 4.8.2, regarding monetaryItemType (i.e., unit measure is an ISO 4217 currency designation). Such facts must not have unit measures that express any scaling (which would interfere with the expression of accuracy by the @decimals attribute).

> **Units representing currencies MUST represent the actual physical value of these currencies, i.e. in basic units, not including any scaling factor in the unit.**

### 3.1 – Choice of Currency for Monetary facts

In general monetary values in an instance must all be expressed in the same (“reporting”) currency, i.e. values should be converted to that currency.

For some specific data items however it may be indicated (in the taxonomy/DPM) that the values reported should be expressed in their "currency of denomination" (i.e. intrinsic currency), and not converted to the reporting currency[17].

This is indicated by such facts having the "Expressed in currency of denomination (not converted to reporting currency)" member of the "Currency Conversion Approach" (CCA) dimension in their context.

Such a marker will often be used in tables that e.g. report a breakdown of figures with a different currency on each sheet. Such facts should have a currency that is consistent with the currency breakdown they intend to express.

One "Reporting" Currency:

**(a) An instance MUST express all monetary facts[18] which do not fall under point (b) using a single currency[19]:**

"Currency of denomination" facts:

**(b) Monetary facts whose associated context contains the eba_CA:x1 member for the CCA dimension MUST be expressed in units of their currency of denomination.**

Currency dimension consistency:

**(c) For facts falling under point (b), whose context also includes the dimension "Currency with significant liabilities" (CUS), the currency of the fact (i.e. unit) MUST be consistent with the value given for this dimension.**

### 3.2 - Non-monetary numeric units

**(a) An instance MUST express its non-monetary numeric values using the "pure" unit, a unit element with a single measure element as its only child. The local part of the measure MUST be "pure" and the namespace prefix MUST resolve to the namespace: http://www.xbrl.org/2003/instance .**

**(b) Rates, percentages and ratios MUST be reported using decimal notation rather than in percentages where the value has been multiplied by 100 (e.g. 9.31% must be reported as 0.0931).**

[17] Note that this currency of denomination might of course actually be the same as the reporting currency for some facts.

[18] i.e. items of monetaryItemType. N.B. this rule does NOT apply to facts representing monetary positions that are explicitly indicated by the data type of the primary item as being required to be reported as "currency-less" decimal values (the value for which may be required to be based on a currency that is not the main currency of the report). (These are likely to be encountered only in the 1.0.1 version of Benchmarking reports)

[19] For clarity – currently, where providing a breakdown by currency where the relevant data points are NOT marked as being reported in their intrinsic currency/currency of denomination, the value of an item in the non-reporting currency should be converted to the equivalent value in the reporting currency (e.g. 2USD -> 1.44 EUR) for submission (the data item being identified as corresponding to an exposure in the breakdown currency by its dimensional attributes). Again, this rule does not apply to facts representing monetary positions which are to be reported using metrics of a decimal data type – for these the specific instructions for the report should be followed as to whether conversion to the reporting currency is required. Stakeholders should be aware that such tables may potentially be subject to change in future.

### 3.3 - Decimal representation

**The value of numeric facts must be expressed in the specified units, without any change of scale and should be expressed without rounding or truncation.**

The content of a numeric fact must therefore not include any scale factors like "%". Specifically, Monetary values[20] must be expressed in units, not in thousands or millions.

i.e. the value €2,560,561.43 may be transmitted as, amongst others, any of

| Acceptable representations of €2,560,561.43 | | |
|---|---|---|
| **Value** | **Value of decimals** | **Implies** |
| 2560561.43 | INF | Exact |
| 2560561.43 | 2 | +/- 0.005 |
| 2560561.43 | 0 | +/- 0.5 |
| 2560561.43 | -3 | +/- 500 |
| *2560561* | *0* | *+/- 0.5* |
| *2561000* | *-3* | *+/- 500* |

Note that although the last two representations (rounding the transmitted value) are acceptable, EBA would prefer that they are avoided where a better estimate for the value is known, and this is transmitted without rounding or truncation as in the first four examples.

But, for example, €2,560,561.43 MUST NOT be transmitted as "2561"

| Wrong representation of an amount of 2,560,561.43 | |
|---|---|
| **Value** | **Value of decimals** |
| 2561 | -3 |

As this represents €2,561 (+/-500), rather than the intended €2,561,000.00 (+/-500)

[20] Whether using monetaryItemType metrics or decimal.

# 3. Additional Guidance

### 3.4 Unused namespace prefixes

Declaring unused namespaces is uncalled for and clutters the instance document.

> **Namespace prefixes that are not used SHOULD not be declared in the instance document. [FRIS04]**

### 3.5 Re-use of canonical namespace prefixes

Most schema authors provide a namespace prefix for their targetNamespace. It is common practice to re-use these prefixes in other XML documents when needed. It may lead to confusion to human readers to see commonly understood prefixes used on a different namespace, or novel prefixes used for a common namespace. E.g. the prefix 'xs' used for the http://xbrl.org/2003/xbrl-instance-2033-12-31 namespace (which would normally be associated with the prefix ‘xbrli’, ‘xs’ in contrast usually being associated with http://www.w3.org/2001/XMLSchema). Note that this does not affect the use of a default namespace attribute on an element to avoid the need for the use of a namespace prefix on the element and its children altogether.

> **Namespace prefixes, where used in instance documents, SHOULD mirror the namespace prefixes as defined by their schema author(s). [FRIS04]**

### 3.6 LEI and other entity codes

#### Practical Considerations

For second level remittance to the EBA, the entity code used must be pre-registered with the EBA by the appropriate CA.

#### Guidance on representation of codes as entity identifier

**LEIs**

In accordance with Eurofiling suggestions the EBA requires the use of “http://standards.iso.org/iso/17442” as the scheme identifier for LEIs and pre-LEIs, i.e.

```
<xbrli:entity>
  <xbrli:identifier
scheme="http://standards.iso.org/iso/17442">LEIIDENTIFIERABCDEFG</xbrli:identifier>
</xbrli:entity>
```

where LEIIDENTIFIERABCDEFG is replaced with the appropriate pre-LEI code for the entity.

Historic errors - acceptance of variations

Please note that previous editions of these filing rules have sadly erroneously specified a scheme URI of "http://standard.iso.org/iso/17442" (note the missing final s of "standards"), in some versions solely using this form and in others the text had a mix both with and without the final s. RFC5141[21] specifies the plural form.

Given this unfortunate history of error

(a) producers of instance documents are encouraged to switch as quickly as possible to producing the correct form "http://standard**s**.iso.org/iso/17442"

(b) The EBA will, and consumers of instance documents are strongly encouraged to, in practice accept either form from submitters.

***Other Identifiers***

*In general, i.e. for first level remittance, or for specific data collections, the scheme URI (and entity code) to be used in an instance should be determined by the relevant competent authority.*

### 3.7 — Unused @id attribute on facts

Unused @id attributes on facts add no value to the supervisor and should not be included in the instance unless they are valuable to the reporter.

**The instance SHOULD NOT include unused @id attributes on facts.**

### 3.8 — Length of strings in instance

Even though there is no limitation on the length of a string reported in an instance, excessively long strings are likely to cause issues in systems involved in the reporting process, many of which will have some practical constraints on the length of string they are able to handle. For this reason it is recommended to limit reported string to only the necessary characters.

**The values of each string SHOULD be as short as possible.**

### 3.9 Namespace prefix declarations restricted to the document element

**Namespace prefixes declarations SHOULD be restricted to the document element.**

Namespace prefixes should be avoided in other elements of the instance document.
This helps to improve the readability of the document and reduces its size. (See examples on pages 35f. .)

### 3.10 Avoid multiple prefix declarations for the same namespace

**Namespaces used in the document SHOULD be associated to a single namespace prefix.**

Two namespace prefixes declarations SHOULD NOT correspond to the same namespace
This helps to improve the readability of the document. (See examples on pages 35f. .)

[21] https://tools.ietf.org/html/rfc5141

## Streaming

There is an XBRL specification called the “XBRL Streaming Extensions Module” which is under development that aims to facilitate the processing of very large XBRL instances. A “Streamable Instance Document” is an XBRL v2.1 instance document that obeys the serialisation constraints defined by that specification.

Several of the filing rules in this document provide guidance on the production of “nice” XBRL instances, i.e. instances that are compact, clear and less prone to errors in creation or usage. However when producing instances focussing on the efficient creation and processing of very large files it may be necessary to adapt or ignore some of these normal best practices. In general, the creation of a “streamable instance document” is a legitimate reason not to follow “SHOULD” rules where they conflict with or inhibit the usage of the Streaming Extensions Module specification.

Rules that are noted as being particularly relevant in this context (i.e. for which it is acknowledged that streamable instance documents may need not to comply) include:

- 1.6.2 —Filing indicators in several tuples
- 2.7 — No unused or duplicated xbrli:context nodes
- 2.21 — Duplicates of xbrli:xbrl/xbrli:unit

# Examples

### Filing indicator examples

#### Conventions:

Positive examples are given a solid border, with crucial sections highlighted with green text and shading:

```
Sample text of example, sample text of example,
Sample text of example, crucial section of example,
Sample text of example, sample text of example
```

Key sections of counterexamples (examples of poor, discouraged or disallowed usage) are highlighted with red text and shading, and the counterexamples are given a dashed border and red background:

```
Sample text of counterexample, sample text of counterexample,
Sample text of counterexample, crucial section of counterexample,
Sample text of counterexample, sample text of counterexample
```

Consider a report containing information for tables C 00.01 (mandatory template), and C 01.00 (mandatory template), but not C 05.01 (required based on activity). The typical approach to indicating this with filing indicator elements would be:

```
<find:fIndicators>
  <find:filingIndicator contextRef="c1">C_00.01</find:filingIndicator>
  <find:filingIndicator contextRef="c1">C_01.00</find:filingIndicator>
</find:fIndicators>
```

Here there is a single “fIndicators” element grouping two filing indicator elements, which indicate the intention to report the tables associated with the codes “C_00.01” and “C_01.00”.

Some **acceptable variations** of this include using the @find:filed attribute:

```
<find:fIndicators>
  <find:filingIndicator contextRef="c2">C_00.01</find:filingIndicator>
  <find:filingIndicator contextRef="c2" find:filed="true">C 01.00</find:filingIndicator>
</find:fIndicators>
```

Or utilising more than one containing “fIndicators” element:

```
<find:fIndicators>
  <find:filingIndicator contextRef="A" find:filed="true">C_00.01</find:filingIndicator>
</find:fIndicators>
…
<find:fIndicators>
  <find:filingIndicator contextRef="A">C_01.00</find:filingIndicator>
</find:fIndicators>
```

It is also acceptable (and in some cases required) to explicitly indicate that a template is NOT reported, e.g.

```
<find:fIndicators>
  <find:filingIndicator contextRef="c1">C_00.01</find:filingIndicator>
  <find:filingIndicator contextRef="c1">C_01.00</find:filingIndicator>
  <find:filingIndicator contextRef="c1" find:filed="false">C_05.01</find:filingIndicator>
</find:fIndicators>
```

**Unacceptable variations** include, for example:

Not indicating that a reported template is reported (C_01.00 is missing):

```
<find:fIndicators>
  <find:filingIndicator contextRef="c1">C_00.01</find:filingIndicator>
</find:fIndicators>
```

Indicating that an unreported template is reported (C_05.01 is not reported):

```
<find:fIndicators>
  <find:filingIndicator contextRef="c1">C_00.01</find:filingIndicator>
  <find:filingIndicator contextRef="c1">C_01.00</find:filingIndicator>
  <find:filingIndicator contextRef="c1">C_05.01</find:filingIndicator>
</find:fIndicators>
```

Duplicating a filing indicator. Here both C_00.01 and C_01.00 appear twice, either repetition is an error, i.e. it does not matter that the two C_01.00 filing indicators are in different tuples:

```
<find:fIndicators>
  <find:filingIndicator contextRef="A" find:filed="true">C_00.01</find:filingIndicator>
  <find:filingIndicator contextRef="A">C_00.01</find:filingIndicator>
  <find:filingIndicator contextRef="A">C_01.00</find:filingIndicator>
</find:fIndicators>
…
<find:fIndicators>
  <find:filingIndicator contextRef="A">C_01.00</find:filingIndicator>
</find:fIndicators>
```

Consider also a template “C_09.02”, known to be expected/anticipated by the recipient (i.e. EBA) to be reported in this instance by this filer. In this case, if this template is not in fact reported, it is it is not acceptable to omit to indicate this explicitly:

```
<find:fIndicators>
  <find:filingIndicator contextRef="c1">C_09.02</find:filingIndicator>
</find:fIndicators>
```

To convey that the template is not reported, it should instead be:

```
<find:fIndicators>
  <find:filingIndicator contextRef="c1" find:filed="false">C_09.02</find:filingIndicator>
</find:fIndicators>
```

### Namespace prefix declaration examples

As shown in the example below, namespace prefix declarations should only be in the document element.

```
<?xml version="1.0" encoding="UTF-8"?>
<xbrli:xbrl xmlns:xsi=”http://www.w3.org/2001/XMLSchema-instance”
    xmlns:link="http://www.xbrl.org/2003/linkbase"
    xmlns:xlink="http://www.w3.org/1999/xlink"
    xmlns:xbrli="http://www.xbrl.org/2003/instance"
    xmlns:eba_dim="http://www.eba.europa.eu/xbrl/crr/dict/dim"
    xmlns:eba_BA="http://www.eba.europa.eu/xbrl/crr/dict/dom/BA"
    xmlns:eba_MC="http://www.eba.europa.eu/xbrl/crr/dict/dom/MC"
    xmlns:eba_OF=”http://www.eba.europa.eu/xbrl/crr/dict/dom/OF”
    … >
    <link:schemaRef xlink:type="simple"
xlink:href="http://www.eba.europa.eu/eu/fr/xbrl/crr/fws/corep/its-2013-02/2013-12-
01/mod/corep_con.xsd"/>
    <xbrli:context id="i10416092">
        <xbrli:period>
            <xbrli:instant>2014-03-31</xbrli:instant>
        </xbrli:period>
        <xbrli:scenario>
            <xbrldi:explicitMember dimension="eba_dim:BAS">eba_BA:x9</xbrldi:explicitMember>
            <xbrldi:explicitMember dimension="eba_dim:EXC">eba_EC:x15</xbrldi:explicitMember>
            <xbrldi:explicitMember dimension="eba_dim:MCY">eba_MC:x195</xbrldi:explicitMember>
            <xbrldi:explicitMember dimension="eba_dim:PRP">eba_PL:x11</xbrldi:explicitMember>
            <xbrldi:explicitMember dimension="eba_dim:TCP">eba_CP:x27</xbrldi:explicitMember>
            <xbrldi:explicitMember dimension="eba_dim:TRI">eba_TR:x4</xbrldi:explicitMember>
                …
    </xbrli:context>
```

No namespaces should be declared on another level than the document level. The following example shows bad practice with the declaration of eba_dim at context level.

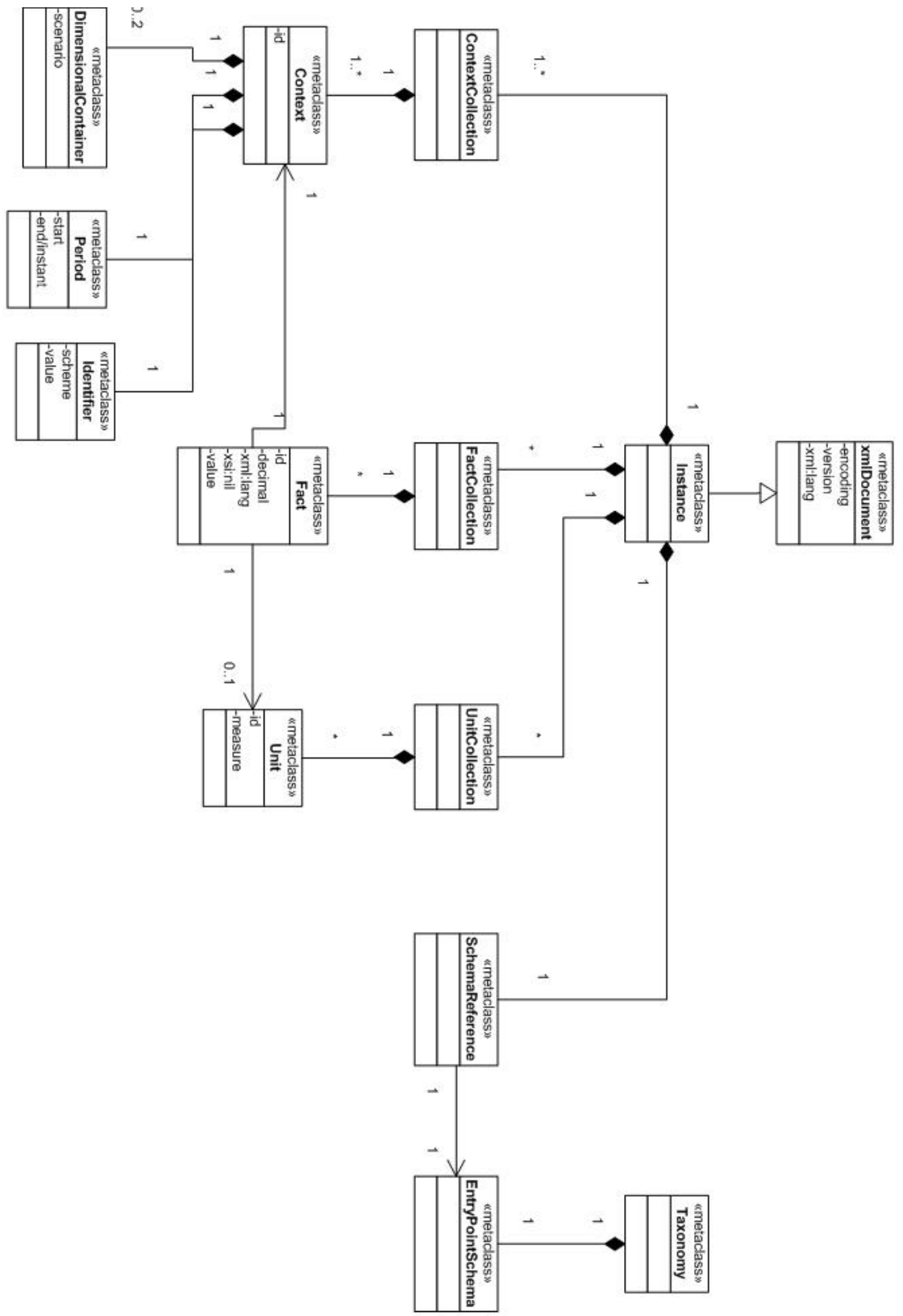
```
<?xml version="1.0" encoding="UTF-8"?>
<xbrli:xbrl xmlns:xsi=”http://www.w3.org/2001/XMLSchema-instance”
    xmlns:xbrli="http://www.xbrl.org/2003/instance"
    xmlns:link="http://www.xbrl.org/2003/linkbase"
    xmlns:xlink="http://www.w3.org/1999/xlink"
    xmlns:eba_BA="http://www.eba.europa.eu/xbrl/crr/dict/dom/BA"
    xmlns:eba_MC="http://www.eba.europa.eu/xbrl/crr/dict/dom/MC"
    xmlns:eba_OF=”http://www.eba.europa.eu/xbrl/crr/dict/dom/OF”
    … >
    <link:schemaRef xlink:type="simple"
        xlink:href="http://www.eba.europa.eu/eu/fr/xbrl/crr/fws/corep/its-2013-02/2013-12-
01/mod/corep_con.xsd"/>
    <xbrli:context xmlns:eba_dim="http://www.eba.europa.eu/xbrl/crr/dict/dim" id="i10416092">
        <xbrli:period>
            <xbrli:instant>2014-03-31</xbrli:instant>
        </xbrli:period>
        <xbrli:scenario>
            <xbrldi:explicitMember dimension="eba_dim:BAS">eba_BA:x9</xbrldi:explicitMember>
            <xbrldi:explicitMember dimension="eba_dim:EXC">eba_EC:x15</xbrldi:explicitMember>
            <xbrldi:explicitMember dimension="eba_dim:MCY">eba_MC:x195</xbrldi:explicitMember>
            <xbrldi:explicitMember dimension="eba_dim:PRP">eba_PL:x11</xbrldi:explicitMember>
            <xbrldi:explicitMember dimension="eba_dim:TCP">eba_CP:x27</xbrldi:explicitMember>
            <xbrldi:explicitMember dimension="eba_dim:TRI">eba_TR:x4</xbrldi:explicitMember>
                …
    </xbrli:context>
```

In this second wrong example the default prefix is redefined in the schemaRef element.

```
<?xml version="1.0" encoding="UTF-8"?>
<xbrl xmlns ="http://www.xbrl.org/2003/instance"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xlink="http://www.w3.org/1999/xlink"
    xmlns:eba_dim="http://www.eba.europa.eu/xbrl/crr/dict/dim"
    xmlns:eba_BA="http://www.eba.europa.eu/xbrl/crr/dict/dom/BA"
    xmlns:eba_MC="http://www.eba.europa.eu/xbrl/crr/dict/dom/MC"
    xmlns:eba_OF="”http://www.eba.europa.eu/xbrl/crr/dict/dom/OF”
    … >
    <schemaRef xmlns="http://www.xbrl.org/2003/linkbase"
        xlink:type="simple" xlink:href="http://www.eba.europa.eu/eu/fr/xbrl/crr/fws/corep/its-
2013-02/2013-12-01/mod/corep_con.xsd"/>
    <context id="i10416092">
        <period>
            <instant>2014-03-31</instant>
        </period>
        <scenario>
            <explicitMember dimension="eba_dim:BAS">eba_BA:x9</explicitMember>
            <explicitMember dimension="eba_dim:EXC">eba_EC:x15</explicitMember>
            <explicitMember dimension="eba_dim:MCY">eba_MC:x195</explicitMember>
            <explicitMember dimension="eba_dim:PRP">eba_PL:x11</explicitMember>
            <explicitMember dimension="eba_dim:TCP">eba_CP:x27</explicitMember>
            <explicitMember dimension="eba_dim:TRI">eba_TR:x4</explicitMember>
                …
    </context>
```

There should be no multiple prefix declarations for the same namespace.
In the wrong example below the xbrl instance namespace is declared by the default prefix and the xbrli prefix.

```
    <?xml version="1.0" encoding="UTF-8"?>
    <xbrli:xbrl xmlns:xsi="”http://www.w3.org/2001/XMLSchema-instance”
        xmlns:link="http://www.xbrl.org/2003/linkbase"
        xmlns:xlink="http://www.w3.org/1999/xlink"
        xmlns:xbrli="http://www.xbrl.org/2003/instance"
        xmlns="http://www.xbrl.org/2003/instance"
        xmlns:eba_dim="http://www.eba.europa.eu/xbrl/crr/dict/dim"
        xmlns:eba_BA="http://www.eba.europa.eu/xbrl/crr/dict/dom/BA"
        xmlns:eba_MC="http://www.eba.europa.eu/xbrl/crr/dict/dom/MC"
        xmlns:eba_OF="”http://www.eba.europa.eu/xbrl/crr/dict/dom/OF”
        … >
```


European Filing Rules: UML model
«metaclass» xmlDocument
-encoding
-version
-xml:lang
«metaclass» Instance
«metaclass» ContextCollection
«metaclass» Context
-id
«metaclass» DimensionalContainer
-scenario
«metaclass» Period
-start
-end/instant
«metaclass» Identifier
-scheme
-value
«metaclass» FactCollection
«metaclass» Fact
-id
-decimal
-xml:lang
-xsi:nil
-value
«metaclass» UnitCollection
«metaclass» Unit
-id
-measure
«metaclass» SchemaReference
«metaclass» EntryPointSchema
«metaclass» Taxonomy

**File naming structure for remittance to the EBA**

The file naming structure for remittance to the EBA is as follows:

LEI_Country_TaxonomyVersion_Module_ReferenceDate_CreationTimestamp.xbrl

| | |
|---|---|
| LEI | Legal Entity Identifiers. For example 549300I84DXMIK4UUL30 for Catalunya Banc |
| Country | ISO Country Code. For example DE for Germany |
| TaxonomyVersion | Framework name defined by the DPM/XBRL taxonomy in uppercase followed by the taxonomy version in 6 digits FRAMEWORKNAMEXXYYZZ. For example for the COREP reporting taxonomy 2.0.1: XX=02,YY=00 and ZZ=01 → COREP020001 |
| Module | Module name as defined by the taxonomy without underscore and in upper-case. For example for the module corep_lcr_con defined by the taxonomy →COREPLCRCON |
| Reference Date | YYYY-MM-DD. For example: 2012-03-31 |
| Creation Timestamp | YYYYMMDDhhmmssfff. For example, 20140602581112463 |

A typical XBRL instance file created by a CA and conforming to the above file naming structure will be named as follows:

635400PNXCHKON18BK07_AT_COREP020102_COREPCON_2014-12-31_20140604181132453.xbrl