

CEN eBusiness Coordination Group

Best practices on the development and registration of

EN 16931-1 compliant registrable Core Invoice Usage Specifications

This document, developed and approved by the CEN eBusiness Coordination Group, aims to give guidance on the creation and implementation of a CIUS with a quality control objective.

It provides guidance for the creation and implementation of Core Invoice Usage Specifications (CIUS) as defined in EN 16931-1.

This guidance document has been developed under the sole responsibility of the eBusiness Coordination Group, and has not the status of a CEN Technical Report (TR), Technical Specification (TS) or European Standard (EN).

Contents

European foreword	2
Introduction	3
Scope	4
Normative references	4
Terms and definitions	5
Compliance	6
Premises	6
Issues that should be considered to avoid unnecessary proliferation	6
Steps for the issuer of the CIUS	8
Guidance on the creation and implementation of CIUS, with a quality control objective	9
Machine readable format	11
Possible future steps	14
Annex A RELAX NG Schema for CIUS configuration	15
Annex B W3C XML Schema for CIUS configuration (non-normative)	18
Annex C Example of CIUS configuration (non-normative)	19

Introduction

The European Commission estimates that “The mass adoption of e-invoicing within the EU would lead to significant economic benefits and it is estimated that moving from paper to e-invoices will generate savings of around EUR 240 billion over a six-year period”. Based on this recognition “The Commission wants to see e-invoicing become the predominant method of invoicing by 2020 in Europe.”

To achieve this goal, Directive 2014/55/EU [1] on electronic invoicing in public procurement aims at facilitating the use of electronic invoices by economic operators when supplying goods, works and services to the public administration. The Directive sets out the legal framework for the establishment and use of a European Standard (EN) for the semantic data model of the core elements of an electronic invoice.

The semantic data model of the core elements of an electronic invoice – the core invoice model – as described in EN 16931-1 is based on the proposition that a quite limited, but sufficient set of information elements can be defined that supports generally applicable invoice-related functionalities. The core invoice model contains information elements that are commonly used and accepted including those that are legally required.

A “Core Invoice Usage Specification” (CIUS) is a specification that provides a seller with detailed guidance, explanations and examples, as well as rules (business rules) related to the actual implementation and use of structured information elements present in the core invoice model in a specific trading situation. An instance document created following a given CIUS is always compliant with the European Standard.

A receiving party may only claim compliance to the core invoice model if he accepts invoices that comply with the core invoice model in general, or with a CIUS, that is itself compliant with the core invoice model. A sending party may claim compliance if he sends invoices that comply to the core invoice model, including those issued in accordance with a compliant CIUS.

This document aims to give guidance on the creation and implementation of a CIUS with a quality control objective. Therefore it is necessary to define a clear set of criteria a CIUS has to comply with, before the CIUS can be registered in the appropriate Registry. Some of these criteria will be validated automatically while others are not.

To hinder excessive proliferation and to guide implementation, publication of CIUS's in a Registry is mandatory and the use of a machine processable format is recommended.

This document applies in case a CIUS is produced as a technical specification with the objective of registering it in the appropriate Registry. This document also establishes requirements for the steps to be taken in the process of creating Core Invoice Usage Specifications (CIUS) as defined in EN 16931-1. Furthermore this document provides guidance for the creation and implementation of Core Invoice Usage Specifications as defined in EN 16931-1.

The following points are in focus here.

- Steps that need to be taken in consideration to avoid unnecessary proliferation and fragmentation in the use of CIUS's;
- Guidance on the creation and implementation of CIUS's, with a quality control objective;

It should be noted that it is planned to apply the same principles and processes to Extensions to be documented in a separate document.

References

EN 16931-1:2017 *Electronic invoicing - Part 1: Semantic data model of the core elements of an electronic invoice*

EN 16931-1:2017 *Electronic invoicing - Part 5: Guidelines on the use of sector or country extensions in conjunction with EN 16931-1, methodology to be applied in the real environment*

[XML-REC] *Extensible Markup Language (XML) 1.0 (Fifth Edition)*, W3C Recommendation 26 November 2008, <https://www.w3.org/TR/REC-xml/>

[XSD-2] *XML Schema Part 2: Datatypes (Second Edition)*, W3C Recommendation 28 October 2004, <https://www.w3.org/TR/xmlschema-2/>

Directive 2014/55/EU of the European Parliament and of the Council of 16 April 2014 on electronic invoicing in public procurement, <https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A32014L0055>

Terms and definitions

CIUS

specification that provides a seller with detailed guidance, explanations and examples, as well as rules (business rules) related to the actual implementation and use of structured information elements present in the core invoice model in a specific trading situation

Note 1 to entry: Source: EN 16931-1:2017

Extension Specification

specification that describes the use of additional information elements, i.e. information elements not defined in the core invoice model, or alterations that add functionality

Note 1 to entry: Source: EN 16931-1:2017 and CEN/TR 16931-5:2017

Compliant invoice instance

invoice instance that respects all rules defined for the core invoice model, which may include the specification contained in a conformant CIUS

Note 1 to entry: Source: EN 16931-1:2017

Registrable CIUS

a CIUS that can be registered in the appropriate Registry and therefore complies with the specifications in this document

Appropriate Registry

registry defined as appropriate by CEN/Technical Committee 434 on Electronic invoicing

Premises

The CIUS has already been subject to extensive discussions, comments, decisions and actual implementations, all of which must be taken into consideration and contribute to the premises for this work:

- CIUS as a concept is described in the EN 16931-1, together with the types of specifications and/or restrictions that are allowed in a CIUS in order to ensure its compliance with the European Standard EN 16931-1;
- other reports generated within CEN/TC 434 and in the European Multi-Stakeholder Forum on e-Invoicing have already made important recommendations on implementing CIUS's e.g. EMSFEI's "Recommendation on the use of 'Core Invoice Usage Specifications' (CIUS)";
- the decision on continuing the work on creating an appropriate Registry (Decision paper from the Copenhagen meeting and CEN/TS 16931-X Electronic invoicing – Functional specification and guidance for registry services);
- the decision in CEN/TC 434/WG 1 to create a machine-readable format.

1 Issues that should be considered to avoid unnecessary proliferation

1.1 General

Various factors will influence the quantity and quality of CIUS's, and each of these should be discussed.

1.2 When should a CIUS be created?

Aside from considerations as to who can issue a CIUS, there is a need to consider when its issuance can be justified.

The concept of CIUS is mainly to enable buyers to establish restriction specifications needed for correctly processing the invoice and progressing towards payment. The buyers communicate to invoice issuers which information they need and it is in the interest of suppliers to provide the information as required by their clients.

In discussions on CIUS's estimates vary as to the expected number of CIUS's likely to be created from a few to many thousands. In this context the question has been raised as to whether simple specifications and/or restrictions on the use of information elements composing the EN should be treated as a CIUS or whether these are simply part of the contract between trading parties. The discussions indicate that there are different views on when it is necessary to create a CIUS, but there seems to be a general consensus to minimize the number of CIUS's and to recommend that a CIUS should only be considered if:

- a) there is a strong business or legal requirement to further restrict or clarify the specifications in the EN;
- b) there is no existing CIUS that covers the requirement (e.g. a sector CIUS or a CIUS made available through an infrastructure such as PEPPOL or a national platform);
- c) the CIUS is likely to apply to a large number of trading parties within a grouping such as a public sector, sub-sector of the public sector or a private sector community with common requirements;
- d) the CIUS doesn't create cross border trade barriers
- e) ...

1.3 Who can issue a CIUS?

To restrict the number and type of CIUS's it is recommended that the parties who may issue a CIUS should be "limited" to the following list.

- a) CEN/TC 434 itself;
- b) CEN National Standards Bodies (NSBs);
- c) CEN/TC 434 Liaison Organizations;
- d) member states;
- e) bodies representing a public sector or business community;
- f) national or European stakeholder fora;
- g) national or European e-invoicing fora;
- h) others (e.g. single buyer and single supplier).

Before mandating a CIUS, single buyers or suppliers should carefully consider whether there is a sufficient business case and whether they can exercise sufficient impact or influence on their trading partners to make the CIUS operative. This aspect is discussed below in the section 'Steps for the issuer of the CIUS'.

1.4 How to create a CIUS?

The tools and other help like templates and support that are made available to the issuer of the CIUS will have impact on both the quantity and quality of the CIUS's.

An example of such a capability is an appropriate Registry being taken forward by CEN/TC 434 where search facilities will provide extensive information individual instances of CIUS's, their associated rules, practices and minimum requirements.

1.5 Formalization

The degree of formalization will also strongly influence the degree of proliferation and render more uniform the specification and the publication of a CIUS. This will make it easier for both issuers and implementers of a CIUS to search, identify, compare, create and implement a CIUS. The list below specifies the minimum degree of formalization.

- a) Complete as a minimum a machine processable representation according to this specification with the necessary information to identify and describe the CIUS;
- b) publish the CIUS in the appropriate Registry;
- c) define and list any specific business rules with a unique identifier and, when appropriate, create validation artefacts;
- d) provide simple instructions how to apply the CIUS to facilitate the invoice preparation and to avoid errors;
- e) provide sample files;
- f) uniquely identify both for referencing and for identification in processing the specification and, when relevant, its version.

1.6 Machine processable representation

The issuer of a CIUS creates the machine processable representation as specified in the Annex A. This includes the creation and publication of related business rules to ensure that all information required for

processing the CIUS is available. The completion of the machine processable representation may also help the issuer to reflect on the need for a CIUS.

1.7 Registry

A main benefit of an appropriate Registry will be to make it easier for a user to search and sort the stored artefacts, gain an understanding of the content, and then compare the component elements e.g. to see the difference between existing CIUS's and/or a planned CIUS and establish whether an existing CIUS could be completely or partially re-used. The more transparent the information the easier it will be to avoid duplicates.

2 Steps for the issuer of the CIUS

In order to avoid proliferation a number of specific steps are specified for prospective CIUS issuer consideration prior to creating a new CIUS.

- a) Elements to be considered before creating a CIUS :
 1. a non-registrable CIUS increases confusion and complexity in the marketplace and potentially additional costs for all parties involved;
 2. economic and other types of impact to mandate the use of CIUS to suppliers, such as responsibility for a sector, sub-sector or trading community;
 3. check if the publication of the CIUS is in line with competition or other laws and regulations;
 4. mandating a CIUS requires extra effort to create, maintain and publish it;
 5. whether the added value in processing invoice instances is sufficient to justify the effort;
 6. the rules applied could potentially create confusion instead of solving issues;
 7. mandating a CIUS to make sure to comply with (local) legislation or policy is like creating rules that one has to comply with the rules. Such a CIUS runs the risk appearing to create new or different rules and even create confusion.
- b) Consider if the intended CIUS is a threat for interoperability
- c) Verify if a CIUS covering the needs already exists
 1. check the available resources to see if an existing CIUS may cover your needs;
 2. contact the responsible persons or entity for an existing CIUS that actually or substantially covers the needs to check if collaboration can be achieved.
- d) Before creating a new CIUS consider what is the maximum reach of the CIUS
 1. consider what the largest reach and number of users is for the CIUS;
 2. contact the authority responsible for the sector, country or other group of users of this largest reach;
 3. check if collaboration or support is available at the largest reach possible;
 4. if a large reach is not feasible, only then consider your own CIUS.
- e) Assess if the necessary resources (human, technical, budget) available to support the creation and maintenance
 1. bear in mind the effort to create, maintain and publish the CIUS;
 2. bear in mind to clearly inform customers and/or suppliers;
 3. check if the back-office systems or processes require enhancement or adaption;
 4. consider a backup plan in case customers and/or suppliers do not respect the CIUS request;
 5. bear in mind the various types of requests from different types of user;

6. define what type of support to give free of charge to the market as a CIUS publisher (e.g. FAQ) and the type of support belonging to the competitive space (e.g. consultancy).
- f) Make sure to publish the CIUS
1. publish it on the appropriate Registry;
 2. publish it on the national authority website if available;
 3. inform customers and/or suppliers so they have access;
 4. publishing it will allow reuse of your CIUS.

3 Guidance on the creation and implementation of CIUS, with a quality control objective

The purpose of a CIUS in a broad sense is to clarify how certain information elements in an invoice instance are used, for example by further restricting, mandating or specifying how elements are used. It is obvious that the documentation describing the CIUS must contain all information for the stakeholders involved. The nature of the stakeholders may require various levels and types of information. Some information will be more textual while other may be more technical.

At the same time avoiding proliferation and considering ease of use and reuse of the CIUS is as important as the specific quality of a CIUS itself.

To achieve the various goals described above, it is necessary that:

- the documentation is presented in a structured and easy to use way;
- the documentation is partly human and partly machine readable, as appropriate;
- the documentation meets quality criteria, for example specified in the appropriate Registry;
- the CIUS's are listed in the appropriate Registry.

The necessary tools and procedures available facilitate the creation/maintenance of a CIUS. These tools and procedures will evolve over time to address the necessary requirements at the given time. Below you find an overview of tools and procedures.

- a) Make sure you use the correct notation for the CIUS specification identifier

A CIUS specification identifier is structured as follows:

SourceSpec#compliant#TargetSpec

- SourceSpec is the core invoice model.
- Conformance states how the changes relate to the SourceSpec, using TOGAF terminology.
- TargetSpec are the identifiers for the extension specification itself and the extension specification or core invoice usage specification that it builds on.

The TargetSpec and the SourceSpec is identified by giving a uniform resource name (urn) [6]. The identifier for the European standard is to include its EN number without the part number indicator (EN 16931:2017)

For clarity, the main parts of the identifier is separated with a hash mark. Hash marks are only used before and after the conformance type.

Following examples show how the identifier is used in different situations.

A core invoice instance document is identified by the specification identifier in the referenced version of the EN16931-1:

Below you find an example

urn:cen.eu:en16931:2017

b) A template helping the issuer to create/maintain the CIUS.

The template contains the information required to define the CIUS. The minimum information required is listed below.

- name and version number;
- date of last update;
- responsible organization;
- contact information;
- underlying syntax;
- brief description of the purpose;
- addressed context (like national, cross border, B2B, sector);
- beginning and end date or 'until further notice' the CIUS is valid;
- hyperlink to the full documentation;
- hyperlink to the validation artefacts (e.g. schematron in CEN/TC 16931-2-2017 syntaxes);
- involved business terms and rules.

The XML template structures the data entered and could be further validated (e.g. XSD or schematron)

The XML template could be the machine readable output that stakeholders need and can with a simple stylesheet be represented in a human readable format.

The template is an XML file that is sent to the Registration Authority for maintaining the information in the registry.

c) Validation artefacts of CIUS requirements

The validation artefacts is expressed in a machine readable format (e.g. schematron) and comply with CEN/TS 16931-2:2017 listed syntaxes, namely UBL 2.1 and UN/CEFACT CII 16B.

The validation artefacts include all the restrictions listed in the CIUS and do not duplicate the EN 16931-1:2017 validation artefacts. The EN 16931-1:2017 validation artefacts can be found via the hyperlink below.

<https://github.com/CenPC434/validation/releases/tag/validation-1.0.0>

Each validation artefact set the identifier, the context, the affected business terms and/or business rules, the severity level (fatal, warning), the message related to validation results.

d) A Registry (website) listing a subset of information of each CIUS.

An example of information subset entered in the template is shown in the following community driven registry:

<https://ec.europa.eu/cefdigital/wiki/display/EINVCOMMUNITY/Community-driven+Registry+of+CIUS+%28Core+Invoice+Usage+Specifications%29+and+Extensions>

e) The maintenance of the registry.

Each modification of the subset of the information in the registry is communicated to the Responsible Authority of the registry.

The Registration Authority validates the information.

The Registration Authority updates the registry uniquely identifying each version.

f) Steps c) and d) will be elaborated with more functionalities by a full Registry Service.

Any requirement related to how and where CIUS register is published the CIUS's will await the Registry WG, and also they will probably look further into the governance of the CIUS' based on the work already started in the study group report on the Registry service.

4 Machine readable format

4.1 General

Every *CIUS publisher* publishes a CIUS using XML based format defined in this section. Each CIUS is described by instance of this format that is called *CIUS definition*. Basic features of CIUS definition are:

- CIUS definition contains all metadata describing CIUS;
- CIUS definition contains human readable description of CIUS targeted for CIUS users;
- CIUS definition might contain machine readable artefacts allowing automatic validation of invoice instances against CIUS;
- CIUS definitions can be machine processed and queried.

4.2 CIUS definition

4.2.1 General

CIUS definition must be well-formed XML document [XML-REC] that is valid against RELAX NG schema in Annex A.

Elements in CIUS configuration are using namespace `urn:cen.eu:en16931:cius-extension:2018`

The following text assumes that prefix `c:` is bound to this namespace.

Individual elements are described in the following clauses.

4.2.2 `<c:cius>`

Root element of CIUS definition.

4.2.3 `<c:meta>`

Container element for all metadata related elements.

4.2.4 `<c:shortName>`

Short identifier of CIUS. For example "CIUS-AT-NAT".

4.2.5 `<c:name>`

Full name of the CIUS definition.

4.2.6 <c:id>

Identifier of CIUS as defined in 7.6 Identification of core invoice usage specifications in [EN 16931-1:2017].

4.2.7 <c:extends>

Optional identifier of the specification that is being furthermore restricted by CIUS definition. For example if one CIUS (e.g. sector specific) is developed on top of other CIUS (e.g. country specific) the original CIUS can be referenced in <c:extends>.

4.2.8 <c:specification>

Optional identification of the specification that serves as a basis for the CIUS definition. Currently it is not expected that other values than urn:ce.eu:en16931:2017 would be used.

4.2.9 <c:version>

Version number of the CIUS definition.

4.2.10 <c:status>

Status of the CIUS definition. Allowed status values are planned, development, active and revoked.

4.2.11 <c:date>

Date of publication of the CIUS definition.

4.2.12 <c:publisher>

Author of the CIUS definition. Any HTML markup can be used inside element for example to format full address or to provide email address.

4.2.13 <c:governor>

Governor of the CIUS definition.

4.2.14 <c:country>

Country where CIUS is used. Element is optional and should be used only when the CIUS definition is country specific. Country is specified by using ISO 3166-1 country codes. Optional.

4.2.15 <c:sector>

Sector to which CIUS applies. Optional.

4.2.16 <c:contact>

Contact email address for any questions and bug reports against the CIUS definition.

4.2.17 <c:abstract>

Short description of the CIUS definition. Any HTML markup can be used in this element. Optional.

4.2.18 <c:link>

URL address pointing to an additional documentation about the CIUS definition. Optional.

4.2.19 <c:description>

Optional element containing human readable description of CIUS. This element has mixed content and can contain arbitrary elements in XHTML namespace (<http://www.w3.org/1999/xhtml>). This allow inclusion of large formatted pieces of text, including tables, images, etc.

4.2.20 <c:rules>

Container element for all business rules.

4.2.21 <c:rule>

Element describing single business rule. It must have id attribute which is unique within CIUS definition. It is recommended that id attribute uses naming convention *BR-abbrev-NN*, where *abbrev* is abbreviation of CIUS publisher and *NN* is an ordinal number of the rule.

The <c:rule> element must contain <c:description> element with a human readable description of the rule. Other elements <c:restrictions> and <c:implementation> are optional.

4.2.22 <c:restrictions>

Element contains one or more restrictions that CIUS definition introduces in the corresponding business rule. Restrictions are categorized slightly differently than in [EN 16931-1:2017] in order to provide more logical approach for defining individual CIUS restrictions.

Elements that describe individual restrictions are: <c:cardinality>, <c:technical>, <c:codelist>, <c:value>, <c:other> and <c:synonyms>. Individual business terms or groups affected by restriction are referenced by using <c:term> element.

4.2.23 <c:term>

Element contains reference to a business term or group from [EN 16931-1:2017]. Terms and groups are identified by strings in the same way as in [EN 16931-1:2017], e.g. BT-103 or BG-17.

4.2.24 <c:cardinality>

This element restrict cardinality of a business term or group. It can contain two attributes minOccurs and maxOccurs that restrict minimal and maximal number of occurrences. Attributes must contain non negative integer numbers. Additionally maxOccurs attribute can also contain value "unbounded".

4.2.25 <c:value>

Element defines restriction on a value domain. Restrictions are declared by using one or more following attributes: pattern, length, maxLength, fractionDigits, totalDigits, minInclusive, maxInclusive. Semantics of these attributes corresponds to semantics of identically named facets in [XSD-2].

4.2.26 <c:codelist>

Element defines allowed values for a codelist. Allowed values are enumerated within <c:value> elements and optional name of codelist can be specified by using <c:name> element.

4.2.27 <c:technical>

This element indicates that constraint is of technical nature and no further info is provided. Typically this can be used when restriction on total size of XML document is being set in CIUS definition.

4.2.28 <c:synonyms>

Definition of synonyms for term. Each synonym is declared using <c:synonym> element.

4.2.29 <c:synonym>

Element contains text of synonym. Attribute xml:lang must be used to identify language if synonyms are provided in several languages.

4.2.30 <c:other>

This element is used for marking terms that are constrained in some way but such restriction can not be described by using other elements.

4.2.31 <c:implementation>

This element must contain syntax attribute with value "UBL" or "CII". Content of element can contain implementation of restrictions using schema languages like Schematron or W3C XML Schema. Different implementation can be provided for each transport syntax.

Note: From restrictions defined using <c:cardinality>, <c:value> and <c:codelist> such implementation can be automatically generated.

Annex A

RELAX NG Schema for CIUS configuration

Digital copy of this artefact is available at <https://github.com/CenPC434/ce-config/blob/master/schema/ce-config.rnc> here

```
default namespace = "http://www.w3.org/1999/xhtml"
namespace c = "urn:cen.eu:en16931:cius-extension:2018"
namespace html = "http://www.w3.org/1999/xhtml"

start =
  # CIUS definition
  element c:cius {

    # Metadata
    element c:meta {
      # Short identifier
      element c:shortName { text },
      # Name of CUIS/extension
      element c:name { text },
      # Identifier as defined in 7.6 Identification of core invoice usage specifications
      element c:id { text },
      # Identifier of specification that is being extended
      element c:extends { text }?,
      # Underlying specification
      element c:specification { text }?,
      # Version number
      element c:version { text },
      # Status of CUIS/extension
      element c:status { "planned" | "development" | "active" | "revoked" },
      # Date of publication
      element c:date { xsd:date },
      # Publisher
      element c:publisher { html-rtf },
      # Governonr
      element c:governor { html-rtf },
      # Country to which CIUS/extension applies
      element c:country { text }?,
      # Sector to which CIUS/extension applies
      element c:sector { text }?,
      # Contact email for any questions and bug reports against CUIS/extension
      element c:contact { text },
      # Short description of CIUS/extensions
      element c:abstract { attribute xml:lang { text }?, html-rtf },
      # Link for more information
      element c:link { xsd:anyURI }?
    },
    # Text description
    description*,
    # Business rules imposed by CIUS/extension
    element c:rules {
      # Business rule
      element c:rule {
        # Unique ID for business rule (BR-{country}-{id} pattern is recommended)
        attribute id { text },
        description+,

```

```

        restrictions?,
        implementation*
    }+
}
}

description =
    # Human readable description can contain mix of text and HTML markup if formatting is
    # needed
    element c:description { attribute xml:lang { text }?, html-rtf }

restrictions =
    # Restrictions that are further applied
    element c:restrictions { (cardinality | technical | codelist | value | other |
    synonyms)+ }

cardinality =
    # Constaint on cardinality
    element c:cardinality {
        attribute maxOccurs { xsd:integer | "unbounded" }?,
        attribute minOccurs { xsd:integer }?,
        term+
    }

technical =
    # Technical constaint
    element c:technical { empty }

codelist =
    # Codelist
    element c:codelist {
        name?,
        term+,
        codelist-value+
    }

value =
    # Restricted value domain
    # Restriction can be made using attributes similar to W3C XML Schema facets
    element c:value {
        # Regular expression describing allowed values
        attribute pattern { text }?,
        # Required length of value
        attribute length { xsd:nonNegativeInteger }?,
        # Maximum length of value
        attribute maxLength { xsd:nonNegativeInteger }?,
        # Maximal number of fraction digits
        attribute fractionDigits { xsd:nonNegativeInteger }?,
        # Maximal number of fraction digits
        attribute totalDigits { xsd:nonNegativeInteger }?,
        # Minimum allowed value
        attribute minInclusive { xsd:decimal }?,
        # Maximum allowed value
        attribute maxInclusive { xsd:decimal }?,
        term+
    }

other =

```

```

# FIXME: what should be content model here
element c:other {
  term+
}

name =
# Name
element c:name { text }

codelist-value =
# Allowed value
element c:value { text }

term =
# Reference to business term (i.e. BT-NN)
element c:term { text }

synonyms =
# Synonyms of term
element c:synonyms { term+, synonym+ }

synonym =
# Synonym of term
element c:synonym { attribute xml:lang { text }?, text }

html-rtf = text & any-html*

any-html = element html:* { any-attribute*, (any-html* & text) }
any-attribute = attribute * { text }

implementation =
# Technical implementation of constraint in any schema language, usually in Schematron
element c:implementation {
  # To which syntax implementation applies?
  attribute syntax { "UBL" | "CII" },
  any
}

any = element * { any-attribute*, (any* & text) }

```

Annex B

W3C XML Schema for CIUS configuration (non-normative)

XSD's are under construction.

Annex C

Example of CIUS configuration (non-normative)

```
<?xml version="1.0" encoding="UTF-8"?>
<c:cius xmlns:c="urn:cen.eu:en16931:cius-extension:2018"
        xmlns:sch="http://purl.oclc.org/dsdl/schematron"
        xmlns="http://www.w3.org/1999/xhtml">

  <!-- meta element contains basic metainformation about CIUS/Extension -->
  <c:meta>
    <!-- Common abbreviation used when referred to the CIUS/Extension -->
    <c:shortName>CIUS-AT-NAT</c:shortName>

    <!-- Human readable name of CIUS/Extension -->
    <c:name>Austrian National Core Invoice Usage Specification</c:name>

    <!-- Identifier as described in "7.6 Identification of core invoice usage
specifications" -->
    <c:id>urn:cen.eu:en16931:2017#compliant#urn:erechnung.gv.at:CIUS-AT-NAT:1.0.0</c:id>

    <!-- Version of CIUS/Extension -->
    <c:version>1.0.0 DRAFT</c:version>

    <!-- Status -->
    <c:status>active</c:status>

    <!-- Date of publication of CIUS/Extension -->
    <c:date>2018-06-18</c:date>

    <!-- Author of extension, any HTML markup can be used inside element -->
    <c:publisher> Philip Helger, Austrian Federal Computing Centre (BRZ), <a
      href="mailto:philip.helger@brz.gv.at">philip.helger@brz.gv.at</a>
    </c:publisher>

    <!-- Governor of extension -->
    <c:governor> Austrian Ministry of Finance (BMF) </c:governor>

    <!-- Country where CIUS/Extension is used -->
    <c:country>AT</c:country>

    <!-- Sector where CIUS/Extension is used -->
    <c:sector>Any</c:sector>

    <!-- Contact email for sending inquiries -->
    <c:contact>philip.helger@brz.gv.at</c:contact>

    <!-- Short description/purpose of spec -->
    <c:abstract>This CIUS-AT-NAT builds on top of the European Standard EN 16931-1:2017.
All rules of
    the underlying specification apply with the addition of the rules stated in this
    document.</c:abstract>
  </c:meta>

  <!-- desc can contain arbitrary HTML with human prose describing CIUS/Extension -->
  <c:description>
```

```

    <!-- Empty in this example. -->
</c:description>

<!-- In this place document contains allowed specification of CIUS as described in
      "7.2.3. Allowed specifications of CIUS" -->
<c:rules>

    <!--
    CIUS is specified as a set of business rules in c:rule element
    Each rule can put several types of constraints on any number of business terms

    The constraints are as follows:

    <c:cardinality minOccurs="M" maxOccurs="N">
        <c:term>BT-XYZ</c:term>
    </c:cardinality>
        - allows changing cardinality of term occurrence, term can be made mandatory, can
be forbidden
            or maximum occurrences can be changed from infinity to smaller number, typically
1

    <c:codelist>
        <c:term>BT-XYZ</c:term>
        <c:value>A</c:value>
        <c:value>B</c:value>
    </c:codelist>
        - allows to specify codelist for values allowed inside business term

    <c:value pattern="[A-Z]{2}[0-9]{10}" maxLength="20">
        <c:term>BT-XYZ</c:term>
    </c:format>
        - allows to specify format (i.e. maximal length, pattern, etc.) for term
        - concrete validation implementatin can be provided in c:implementation element
        - c:value only identifies which terms should have restricted format

    <c:technical>
        - defines that there is some technical constraint like maximal size of document
        - it's not necessary to link such constraint to term

    <c:other>
        - some other restriction, for example co-occurrence, semantic definition

    <c:synonyms>
        - synonyms of term

    -->

<c:rule id="BR-AT-1">
    <c:description>Postal address line 1 becomes mandatory</c:description>
    <c:restrictions>
        <c:cardinality minOccurs="1">
            <c:term>BT-35</c:term>
            <c:term>BT-50</c:term>
            <c:term>BT-64</c:term>
            <c:term>BT-75</c:term>
        </c:cardinality>
    </c:restrictions>
    <!--

```

Concrete implementation of this rule in Schematron or XSD can be specified inside c:implementation element

```
<c:implementation syntax="UBL">
  <sch:rule context="...">
    <sch:assert test="...">Postal address line 1 must be specified</sch:assert>
  </sch:rule>
</c:implementation>
<c:implementation syntax="CII">
  <sch:rule context="...">
    <sch:assert test="...">Postal address line 1 must be specified</sch:assert>
  </sch:rule>
</c:implementation>
-->
</c:rule>
```

```
<c:rule id="BR-AT-2">
  <c:description>Postal address city becomes mandatory</c:description>
  <c:restrictions>
    <c:cardinality minOccurs="1">
      <c:term>BT-37</c:term>
      <c:term>BT-52</c:term>
      <c:term>BT-66</c:term>
      <c:term>BT-77</c:term>
    </c:cardinality>
  </c:restrictions>
</c:rule>
```

```
<c:rule id="BR-AT-3">
  <c:description>Postal address post code becomes mandatory</c:description>
  <c:restrictions>
    <c:cardinality minOccurs="1">
      <c:term>BT-38</c:term>
      <c:term>BT-53</c:term>
      <c:term>BT-67</c:term>
      <c:term>BT-78</c:term>
    </c:cardinality>
  </c:restrictions>
</c:rule>
```

```
<c:rule id="BR-AT-4">
  <c:description>Actual delivery date or invoicing period must be
provided</c:description>
  <c:restrictions>
    <c:cardinality minOccurs="1">
      <c:term>BT-72</c:term>
      <c:term>BT-73</c:term>
      <c:term>BT-74</c:term>
    </c:cardinality>
  </c:restrictions>
</c:rule>
```

```
<c:rule id="BR-AT-5">
  <c:description>Document level allowance VAT rate becomes mandatory</c:description>
  <c:restrictions>
    <c:cardinality minOccurs="1">
      <c:term>BT-96</c:term>
    </c:cardinality>
  </c:restrictions>
```

```

</c:rule>

<c:rule id="BR-AT-6">
  <c:description>Document level charge VAT rate becomes mandatory</c:description>
  <c:restrictions>
    <c:cardinality minOccurs="1">
      <c:term>BT-103</c:term>
    </c:cardinality>
  </c:restrictions>
</c:rule>

<c:rule id="BR-AT-7">
  <c:description>External document location is not supported</c:description>
  <c:restrictions>
    <c:cardinality maxOccurs="0">
      <c:term>BT-124</c:term>
    </c:cardinality>
  </c:restrictions>
</c:rule>

<!--
  I think that section "7.3 What may be specified in a CIUS" does not allow such
type of constraint
  But such constraint is sensible so perhaps we should send defect report against EN
16931-1:2017
-->
<c:rule id="BR-AT-8">
  <c:description>The size of all attached documents together may not exceed
15MB</c:description>
  <c:restrictions>
    <c:technical/>
  </c:restrictions>
</c:rule>

<c:rule id="BR-AT-9">
  <c:description>Invoiced item VAT rate becomes mandatory</c:description>
  <c:restrictions>
    <c:cardinality minOccurs="1">
      <c:term>BT-152</c:term>
    </c:cardinality>
  </c:restrictions>
</c:rule>

</c:rules>

</c:cius>

```