Specifications for the identification of subdivisions in EU legislation using ELI

Table of contents

Part 1

Business Specifications

1 Introduction

The European Legislation Identifier (ELI) is a framework to make legislation metadata available online in a standardised format so that it can be accessed, exchanged and reused across borders. This initiative, taken jointly by EU countries and institutions, is enshrined in the Council Conclusions of 6 November 2017 on the European Legislation Identifier (2017/C 441/05).

According to these conclusions, "ELI gives the Member States and the European Union a flexible, self-documenting, consistent and unique way to reference legislation across different legal systems. ELI URIs are a stable means of uniquely identifying any legislative act throughout the European Union, while taking into account the specificities of national legal systems".

The implementation of ELI is part of a wider effort led by the Publications Office of the European Union in the area of standardisation of metadata and development of common vocabularies. The overall objective is to ensure that various information systems, both on the EU level and beyond, use the same 'language' to exchange information.

The present document complements the technical specifications defining ELI in the context of AKN4EU, the future machine-readable structured format for the exchange of legal documents in the EU decision-making process. Therefore, the specifications outlined on the following pages are primarily aimed at the EU institutions for their future exchanges of legal documents. However, it can serve as a basis for the implementation of references to subdivisions by any user interested in ELI such as members of the ELI Task Force, national administrations or even private individuals.

2 Objectives

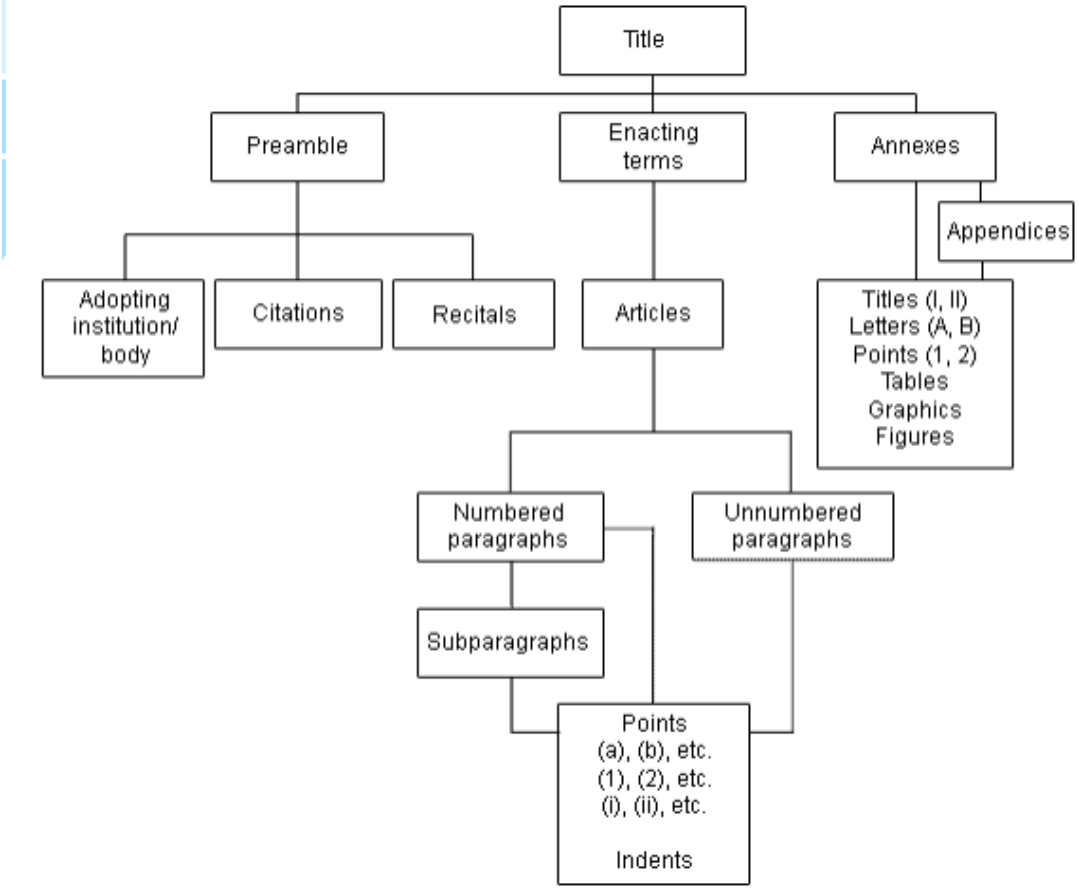
2.1 Structure of a legal act

This document specifies how subdivisions in legislation should be identified with an ELI.

It is based on the IMFC Common Vocabulary/terms used in legal citations and text references.

The Interinstitutional Style Guide describes the following subdivisions of a legal act:

'This diagram shows the basic elements of a legal act. Depending on the complexity of the text, elements such as parts, titles, chapters or sections may be used in the preamble, enacting terms and annexes.



Interested readers can also read this page.

2.2 Canonical identifier vs. User identifier

The question of the identification of the subdivision of a legal act can be understood from 2 different perspectives:

1. How to assign identifiers to subdivisions in an (XML) content (that is "elifying" a content); the subdivision identifiers in the XML content are referred to as ***canonical identifiers***.
2. How to write a reference to the subdivision of another legal act, typically in a corrigendum; these references are referred to as ***user references***. There can be differences between a *canonical identifier* and a *user reference* to the same subdivision;

The first part of this document deals with the generation of *canonical identifiers*. Rules governing *user references* are described in the second part of this document.

3 Scope of business specifications

3.1 Restricted to where ELI is applicable

Although accessing the subdivision of a document makes sense in a general way (in Eur-Lex), the scope of this analysis is restricted to the subdivisions of the documents that fall in the scope of ELI. The scope of ELI is:

1. Everything published in OJ-L.
2. Their corresponding publications in DD.
3. Their corresponding consolidations[1].
4. The treaties from OJ-C.

3.2 References to subdivision ranges are out of scope

The references to ranges of articles and paragraphs is out of scope of this specification. E.g. "article 28 to 36".

4 Use-cases for subdivisions identification

4.1 Table of Contents

A typical use-case relying on the identification of subdivision is the table of contents navigation in Eur-Lex: clicking on the element in the ToC brings the user down to the correct anchor within the content of the page.

4.2 Drafting amending acts in the XML-editor EdiT

Drafting amending acts in the XML-editor EdiT: the XML-editor relies on an 'elified' base act (where subdivisions have been identified with eIDs) and on the use of *user references* ELIs in the metadata describing the modification (pointing to the target subdivision of a modification without specifying a version).

Drafting of an amending act in EdiT will start from importing the base act (which can be a consolidated version). Each subdivision in the base act in

1 In phase one only the main part of the act will be 'elified', excluding annexes, even if they will be included in the same file, as is usually the case in consolidated versions.

AKN4EU format[2] should have an identifier (eID) which will facilitate targeting this subdivision with an ELI, in the metadata of the amendment in the amending act.

Simplified example:

1. Subdivision in the base act, with eID:

```
<paragraph eId="art_2.unp_1">
```

2. Amendment drafted in EdiT:

```
‘Regulation (EU) yyyy/nnnn is amended as follows:
In Article 2, first paragraph is replaced by the following: “…”’
```

3. Metadata of the amendment:

```
<textualMod type="substitution">
  <source href="..."/>
  <destination href="http://data.europa.eu/eli/reg/yyyy/nnnn/art_2/unp_1"/>
</textualMod>
```

Note on consolidation: The <activeModifications> meta contains, in a formalized way, the amendment information: operation, target location in the amended act, and, eventually, text of the amendment.

When drafting an amending act, the drafter works concretely on the version of the amended act that is valid at the drafting time (he works on a consolidated text). So, he can make an amendment on an “article 5”, even if this article did not exist in the OJ version and has been introduced by a previous amendment.

He expresses the target location in the amended act, in a dynamic way (virtual way, with no specific version information) as there is no guarantee that the consolidated text on which he has worked will be still valid when the drafted amendment will really enter into efficacy.

As the metadata in the <activeModifications> normalizes the amendment, the destination is also expressed in a dynamic way (virtual URI).

So, there is no information "/oj" or "/<date>". The only place where the information on version is specified is in the meta <activeRef>, for information purpose only.

In the passiveModifications (metadata on the consolidation) the information on the version is pre-sent as these metadata are build when "executing the amendment" and describes this operation of consolidation.

2 If the base act exists only in Formex, it will be converted into AKN4EU and ‘elified’ before the import into EdiT (the subdvisions will be given eIDs).

5 ELI URI pattern of a subdivision

5.1 URI pattern description

Subdivisions in ELI are identified with a slash-separated sequence of URI components of the form

```
subdivisions_code [ "_" + subdivision_ID ]
```

The subdivision code is taken from the Authority table 'Subdivision' (not all the codes from the table will be used, see below). Subdivision ID is a suffix inserted after the subdivision code and an underscore; it is optional.

Examples are:

- art_2
- art (case of sole articles, see section 5.3.6)
- pbl_1 (with a default "_1" SubdivisionID, see section 5.3.1)
- art_212a
- pnt_b

This sequence of URI components is inserted in the general ELI URI that identifies the whole act, at the end of the work-level ELI URI, that is after the {type}/{year}/{natural_number} sequence, and before the {version} component, if there is one, to indicate the precise version being referred to.

This effectively implies that each subdivision is considered to be a FRBR work.

Example

Given the ELI URI for a full act, referring to the OJ version (with final "/oj" to indicate the version) http://data.europa.eu/eli/dir/yyyy/nnnn/oj and consolidated version that is in force in 14-06-2021,

- Example ELI URI of a subdivision of this act http://data.europa.eu/eli/dir/yyyy/nnnn/art_2/unp_3/oj ("Article 2, third paragraph, in the OJ version of this act")
- Example ELI URI of the same subdivision in the version of the 2021-06-14 http://data.europa.eu/eli/dir/yyyy/nnnn/art_2/unp_3/2021-06-14
- Example ELI URI of the same subdivision in virtual way (dynamic reference that do not specify a version): http://data.europa.eu/eli/dir/yyyy/nnnn/art_2/unp_3

5.2 Subdivision Code in ELI URI

Business subdivisions	Example legal citation	Code from AT Subdivision
title of an act	'the title of Regulation (EC) No 1950/2006 is replaced by the following:'	**tit**
preamble	'as defined in the preamble'	**pbl**
citation	'in the first citation'	**cit**
recital	'in recital 1'	**rct**
enacting terms	'provided for in the enacting terms'	**enc**
part	'in Part I' 'in Part One'	**prt**
title	'in Title I'	**tis**
chapter	'in Chapter 1' 'in Chapter I'	**cpt**
section	'in Section 1'	**sct**
title of a higher subdivision (part, title, chapter, section)	'the title of Chapter II is replaced by the following:'	**tit**[3]
subsection	'provided for in Subsection 2 of Section 3'	**sbs**
article	'Article 15 is deleted'	**art**
title of an article	'the title of Article 11 is replaced by the following:'	**tit**
numbered paragraph	'in Article 19, paragraph 1 is replaced by the following:'	**par**
unnumbered paragraph	'in the first paragraph of Article 20'	**unp**
Subparagraph[4]	'in Article 2(1), the second subparagraph is amended as follows:'	**sub**

3 As per definition from the authority table "This element indicates the designation of an act or the designation of a subdivision (e.g. article, chapter, part) of an act. The complete title of an act comprises type of act, number of act, name of the author of the act, date of adoption and subject matter. In the case of subdivisions, the title contains the subject matter."

4 Subparagraphs can be found only under numbered paragraphs, points and indents

point	'the third indent of the first subparagraph of point (a) of Article 2 is replaced by the following:'	**pnt** (note pta or pti are not used to identify subdivisions in ELI)
indent	'the third indent of the first subparagraph of point (a) of Article 2 is replaced by the following:'	**idt**
introductory part	'in paragraph 4, the introductory part of the second subparagraph is replaced by the following'	**inp**
closing part	-	**wrp**
figure[5]	'in point (b), Figure 2 is replaced by the following'	**fgr**
image	'the drawing in Figure 1', 'the following picture'	**img**
table	'as laid down in Table 5'	**tab**
scientific formula	'calculated according to the following formula:'	**sfr**
legend	'the legend of the table is replaced by the following'	**lgd**
line of legend	-	**llg**
annex	'in Annex VII'	**anx**
title of an annex	'in Annex VII, the title is replaced by the following:'	**tit**
appendix	' the provisions of Appendix 7 of Annex 4a'	**app**
table of contents	-	**toc**

Note how the subdivision code "tit" is used for

- Title of an act
- Title of an article
- Title of a higher subdivision
- Title of an annex

5 Image together with accompanying text, i.e. heading, numbering, legend etc.

This is consistent with the definition from the authority table. We don't see any risk of identifier collision due to this, as these titles will always be under a different parent subdivision and cannot be referenced without specifying this parent subdivision, thus guaranteeing uniqueness.

Also note that "list" is not used as a subdivision for ELI, as it was determined that this level is not required to guarantee uniqueness of subdivisions (2 lists inside the same article shall be in 2 separate unnumbered paragraphs, 2 lists inside the same numbered paragraph shall be in 2 separate subparagraphs).

5.2.1 Rules for the subdivision code

- The code is always 3 letters;
- Taken from the codes in the Authority table: https://op.europa.eu/en/web/eu-vocabularies/at-dataset/-/resource/dataset/subdivision
- It is always in lowercase;

5.3 Subdivision_ID in ELI URI

5.3.1 Subdivision_ID always present

The `["_" +subdivision_ID]` is present even for subdivisions that are not numbered and cannot be repeated inside their context. For preamble for example, the identifier would be `pbl_1`.

The following subdivision codes use a default "_1" Subdivision_ID:

- Preamble: `pbl_1`
- Enacting Terms: `enc_1`
- Closing Part: `wrp_1`
- Introductory Part: `inp_1`
- Table Of Contents: `toc_1`
- Title of the act, title of higher subdivisions, title of article, title of the annex: `tit_1`

  This is inline with the choice that has been made to identify the subdivisions of treaties.

5.3.2 Subdivision_ID can be number or letters or symbols or mixed

The Subdivision_ID can be a number ("art_1"), letters ("pnt_a"), or mixed ("art_212a").

5.3.3 Subdivision_ID is case-sensitive (but can be searched case-insensitive)

Canonical identifiers are case-sensitive and follow the case used in the document. It may be possible to have both "point (a)" and "point A" within the same subdivision of a higher level, thus leading for example to `sct_1/pnt_a` and `sct_1/pnt_A`.

User references must be matched case-insensitive (see the section 8.2).

5.3.4 Subdivision ID uses English way of identifying the subdivision

The numbering used for subdivision ID is based on the way the subdivisions are numbered in the English version of a text (E.g., `art_1a` and not `art_1bis` (French)). The subdivisions in every XML linguistic variant of the same legal act will have the same canonical identifier based on the English variant.This applies also to cases, where English version is not available (for example corrigenda specific to one linguistic version that is not English). This requires a mapping between the way to number subdivisions in a specific language (Greek, French, etc.) and their corresponding ELI identifier based on English (e.g., map `1bis` to `1a`). This mapping between language-specific subdivision numbering is provided in annex.

5.3.5 Subdivision_ID is what is written in the text

The Subdivision_ID is the subdivision identifier as it is written in the (English variant) of the text, with no interpretation.

In particular some texts use Roman numbering for the identification of their titles, and no translation to Arabic numbering will be made.

5.3.6 Special cases

1.1.1.1 Unique articles / "Sole articles" and unique annexes

Some acts have a unique article that is not numbered.

In this case the canonical URI for the article would be simply "art", and not "art_1" (note that a single non-numbered article is not the same as "article 1").

The same situation applies for single annexes. The annex would be referred to with simply "anx", and not "anx_1".

1.1.1.2 Article -1

Also note the special case of "-n" as numbering (for example "Article -1": an Article inserted before the first one, for which the derived identifier would be `art_-1`.

5.4 Rules for subdivisions hierarchy

5.4.1 Identification of articles or subdivisions below articles starts at the article level

The article numbering is continuous across the whole enacting terms of an act ; as such a reference to an article or any subdivision below an article does not include higher subdivisions (chapters, sections, etc.), but starts directly at the article level.

Given the following legal act structure

- Chapter 1
    - Article 1
    - Article 2
        - Paragraph 1
        - Paragraph 2
        - Paragraph 3
            - Subparagraph 1
                - Introductory part
                - Point a
                - Point b
- Chapter 2
    - Article 3

« article 2(3) point b » is identified with `art_2/par_3/sub_1/pnt_b` and _not_ with `cpt_1/art_2/par_3/sub_1/pnt_b`.

5.4.2 Identification of higher subdivisions includes the full subdivision hierarchy

The identification of « higher subdivisions », above the article level, shall include all the ancestor subdivisions.

Given the following legal act structure

- Title A
    - Chapter 1
        - Article 1
        - Article 2
    - Chapter 2
        - Article 3

- Title B
    - Chapter 3
        - Article 4
        - Article 5

A reference to Chapter 3 is made with `tis_B/cpt_3`.

5.4.3 Numbered paragraphs, unnumbered paragraphs and subparagraphs

Subparagraphs (sub) are used inside numbered paragraphs (par), points or indents, and as such are present in the canonical identifiers of such subdivisions.

Unnumbered paragraphs (unp) always have a single subparagraph. So such a subparagraph is never identified in the canonical identifier (nor in a user reference, see below).

5.4.4 Text of amendments does not follow the hierarchy structure

The text of amendment can contain subdivision that breaks the current hierarchy, typically with articles below other articles, or higher subdivisions, like titles or annexes, below articles.

In this case the full subdivision hierarchy is used to identify the lower-level subdivision.

This topic can be the subject of further discussions.

Given the below example

- Article 1
    - Paragraph 1
    - Paragraph 2 "The content of article 23 is replaced with the following content"
        - Article 23
            - … new content of article 23

The ELI URI for Article 23 is art_1/par_2/art_23

Note: the correct interpretation of this is Article 1, paragraph 2, article 23 of the "text of the amendment" (CoV concept).

The "text of the amendment" can have a very complex structure (as a complete annex, or chapter, …) and this structure breaks the normal structure of the amending act, restarting a new hierarchy. But it is still part of the amending act even if the numbering of the internal structures is based on the location of the amended act.

The "consolidation" will take this "text of amendment" to creats a new "version" of a fragment of the amended act, according to the instructions given in the amending act.

Given the below example:

- Article 1
    - Paragraph 1
    - Paragraph 2
      The following modifications are done:
        - Point a: The annex of the act is replaced with the following:
            - Annex 1 …

The ELI URI for Annex 1 is art_1/par_2/sub_1/pnt_a/anx_1.

6 Relationship between subdivision eID in AKN XML and ELI canonical URI

In AKN XML, subdivisions will be identified with an eID that:

- Contains only the subdivision part, and not the full document URI, with domain name, type and reference of the document, etc.
- Uses a dot as a separator and not a slash, to comply with XML ID constraints.
- Do not contain a version indicator, either "/oj" to indicate the original OJ version or "/yyyyddmm" to indicate a consolidated version at a given date. This version indicator is part of the metadata of the complete act.
- Can contain a prefix (case of CONS-LEG acts with multiversioning) when a file contains all the evolution of an act during the time and so, can contain multiple identical subdivisions.
  For example if, in an act, an article 2 is replaced by a new text,
    - The eId of the first version of article 2 is "art_2"
    - The eId of the second version of article 2 is "prefix of the modifier (consisting of ELI components)\art_2

Thus, to reconstruct the complete ELI URI of a subdivision, it is necessary to

- Prepend the complete ELI URI of the document to the subdivision identifier.
- Remove the prefix
- Replace dots with slashes.
- Add the version indication at the end of the URI. The exact way how to determine this precise version indication is out of scope of this specification; it depends on additional information that is either in the document metadata (a metadata to indicate the version) or could be in the subdivision itself (in the case of the multiversioning documents described above).

Example

- Given an AKN XML document for the act identified with the ELI http://data.europa.eu/eli/dir/2020/1234 (fake example).
- Given the version indicator of this document indicated in the metadata "20210614".
- Given an XML element identified with the subdivision identifier "art_2.unp_1.pnt_a".
- The full ELI URI of the subdivision of that specific version of the act in an RDF graph would be http://data.europa.eu/eli/dir/2020/1234/art_2/unp_1/pnt_a/20210614

7 Working hypothesis

7.1 Corrigenda shall be considered as separate works for each linguistic variant

Today a corrigendum is considered a single work even though its linguistic variants do not have the same structure (a Spanish version of the corrigenda may correct article 1 only, while the Italian version corrects article 2, 24 and 36. Hence the subdivision structure of each of these expressions of the corrigenda is different).

Since we need to represent each subdivision as a Work, and we need to make the assumption a subdivision work has expressions in the same language as the languages of its containing document, the corrigenda will need to be split into different works, one for each linguistic variant.

Note: To be discussed with the OJ production unit if we still will have the situation also for the future that corrigenda with different content in different LVs are still published with a shared identifier and are hence a single work.

7.2 Homogeneous structure across linguistic variants

We assume that differences in structure between the linguistic variants (that can result from errors or be made on purpose) are out of scope of this analysis. It is assumed all the linguistic variants of the same text have the same content structure.

This is however not true for corrigenda, where the structure of the corrigenda is currently different from one linguistic variant to another; but, as stated above, we consider each corrigendum as a separate work.

Part 2

User references

8 Differences between user references and canonical identifiers

8.1 Abstract references

It shall be possible to create *user references* to a subdivision of a legal act at an abstract level (without indicating the precise version being referenced), or at a precise "non-abstract" version (non-abstract means that the specific version of the resource is indicated, for example, the version as published initially in OJ, the version in force on a specific date etc.).

In case of *abstract user references (also called dynamic references)*, additional information will be needed to resolve the reference correctly to one specific version. The technical specifications for the implementation of subdivision ELI will indicate what information is needed and how the reference will be resolved, depending on the use case.

Note that most legal citations pointing to a subdivision, in particular in the text of amending provisions, are abstract (no specific version of an act indicated).

Examples:

- An abstract user reference is made to `art_2.par_2` of the act identified by the ELI http://data.europa.eu/eli/dir/yyyy/nnnn
- In an additional metadata of a file or in the application context, we know the reference is actually made to the version "2000-07-21" of the act.
- The complete reference of the subdivision in this given consolidated version of the act will be actually http://data.europa.eu/eli/dir/yyyy/nnnn/art_2/par_2/20000721

Note on consolidation: The amendment is done on the text that is "valid" at the moment of the amendment

The metadata regarding the consolidation (passiveModifications) must refer a concrete version as it formalizes the concrete operation that has be done to create this consolidation (it is a metadata describing the operation of consolidation and all temporal consequences).

The metadata of the amendment (activeModifications) does not specify a concrete version as the drafter of the amending act cannot know which consolidated version of text is efficient when applying the amendment.

8.2 Case-insensitive match

When resolving a *user identifier* against canonical identifiers in the content, the expected strategy is that implementations would search subdivisions first

in a case-sensitive way, but if not found, they shall default to a case-insensitive matching.

For example, when trying to find subdivision `…sub_1/pnt_a`, if no subdivision with URI `…sub_1/pnt_a` is found (case-sensitive), a second search shall be attempted in a case-insensitive way to match a potential `…sub_1/pnt_A`.

8.3 It shall be possible to write user references to subdivisions without subparagraph level when there is only one subparagraph

Typical *user references* to subdivisions inside articles are made without specifying the unnumbered paragraph when there is only one, while the unnumbered paragraph is present and identified as such in the canonical identifier[6]. The same thing applies when there is only one subparagraph under a numbered paragraph, point or indent. In other words

- A user reference may omit the unnumbered paragraph level (unp) when there is only one unnumbered paragraph below an article. (The canonical identifier always mentions it.)
- A user reference may omit the subparagraph level (sub) when there is only one below a numbered paragraph, a point or an indent. (The canonical identifier always mentions it.)
- A user reference never uses the subparagraph level below the unnumbered paragraph. Subparagraph is never used in the context of unnumbered paragraphs. (The canonical identifier never mentions it.)

For example the modifier 32015L2302 modifies 32011L0083 this way

"Point (g) of Article 3(3) of Directive 2011/83/EU is replaced by the following" ("L'article 3, paragraphe 3, point g), de la directive 2011/83/UE est remplacé par le texte suivant")

The corresponding user reference would be `art_3/par_3/pnt_g`, without any subparagraph indication.

However the canonical identifier for that subdivision will be: `art_3.par_3.sub_1.pnt_g.` We see there is an extra "sub" level.

The *canonical identifier* may have "sub_1" in it even if there is a single subparagraph, and the *user references* cannot be forced to use an extra

[6] Note that unnumbered paragraphs (unp) always have a single subparagraph. So such a subparagraph is never identified in the canonical identifier or in a user reference. But subparagraphs are present in the canonical identifier when they are in a numbered paragraph.

"sub_1" if not present in the original citation. Thus, there needs to be a way to reconcile the user references without subparagraph to the canonical identifiers including subparagraph.

Samples of user references and their corresponding canonical ID are given I the table below

User reference	Canonical URI
art_1.pnt_a	art_1.unp_1.pnt_a
art_1.par_1.pnt_a	art_1.par_1.sub_1.pnt_a
art_1.pnt_a.pnt_c	art_1.unp_1.pnt_a.sub_1.pnt_c
art_1.par_2.pnt_a	art_1.par_2.sub_1.pnt_a
art_1.unp_2.pnt_a	art_1.unp_2.pnt_a
art_1.inp	art_1.unp_1.inp
art_1.par_2.inp	art_1.par_2.sub_1.inp
art_1.unp_3.pnt_a	art_1.unp_3.pnt_a (the canonical identifier is the same)
art_1.par_2.sub_3.pnt_a	art_1.par_2.sub_3.pnt_a (the canonical identifier is the same)

8.3.1 This is possible only for unnumbered paragraph 1 or subparagraph 1

The matching strategy described above to match references with an extra unnumbered paragraph 1 or subparagraph 1 works only in cases where the implicit missing subdivision is the first. It cannot be applied in situations that would require a parsing of the actual XML content to find the subdivision.

The strategies are limited to what can be done easily without parsing the XML.

Any references to something that is below an "unp_2" or "sub_2" must mention these subdivisions explicitly.

Consider the following example

Article 1

1. This is first sub of par 1.

This is intro starting the second sub:

a) xxx,

b) xxx.

2. This is par 2.

User reference "in Article 1(1) point (a)" (which is enough/unique in terms of legal citation, if there is only one point (a) in the paragraph 1 of article 1), must explicitly contain a "sub_2" level in its reference to actually point to the correct subdivision: `art_1.par_1.sub_2.pnt_a`

Note that a more precise legal citation would use an explicit subparagraph reference "in Article 1(1) subparagraph 2 point(a)".

8.4 Special cases

8.4.1 Sole / unique articles

A user_reference to "art_1" should first be searched for "art_1", but, if not found, the search should be extended to also match "art" (case of sole article). The same applies to annexes, where a reference to "anx_1", if not found, should searched for "anx".

9 No language specific references (art_1bis vs. art_1a)

It will *not* be possible to refer to a subdivision with a language-specific reference (typically "art_1bis" vs. "art_1a"). Only the English way of numbering subdivisions can be used to refer to them.

10 References to manifestation URI of a subdivision

It shall be possible to refer to a subdivision of a specific language variant of the text, and in a given format, e.g.

http://data.europa.eu/eli/dir/yyyy/nnnn/art_2/par_2/20000721/fra/xml

It shall also be possible to refer to a specific language and format in an "abstract reference" without specifying the version

http://data.europa.eu/eli/dir/yyyy/nnnn/art_2/par_2/fra/xml

References to AKN document are made with the corresponding format code "akn4eu" …/art_2/par_2/fra/akn4eu. The expected behavior is to retrieve the piece of AKN XML encoding this subdivision. This will not be, strictly speaking, an AKN-compliant XML document as it will not contain the metadata part, document identifier, etc. This could change later, depending on the implementation of these user references.

Note that references to AKN4EU Zip using "akn4eu_zip" format scope are not possible for subdivisions, as it would not make sense to retrieve the subdivision content in a zip format.

11 What happens when accessing a subdivision ELI URI?

11.1 Introduction

This section specifies what happens when the full URI of a subdivision is accessed; e.g., http://data.europa.eu/eli/reg/2005/2075/art_2/par_2/oj.

11.2 Formats

There is a need to resolve *user references* to subdivisions in XHTML, HTML or XML.

As for dereferencing a subdivision URI to a PDF, it is technically feasible to have anchors inside them, but

1. OJ PDFs don't contain these anchors;
2. Even if they did, we would not be able to link the ELI subdivision with the corresponding anchor;
3. It is not possible to regenerate the PDFs to add anchors;

11.3 Accessing the subdivision in context vs. the subdivision content fragment

The main use-case is the "Table of Content": displaying the whole text at a specific anchor corresponding to a subdivision.

Another use-case could be to return only the fragment of the corresponding subdivision (without the full content).

If these 2 ways of accessing a subdivision are implemented ("in context" or "fragment only"), the client must be able to state whether they want to see the subdivision in context or the fragment only in their request.

To allow this, we need:

- a separate URL to access the content fragment (so that it can be stored in cache if needed), using a new URI pattern.
- and an extra HTTP Accept header extension to indicate if the client wants the document fragment.

Typically:

1. The subdivision URI `/eli/dir/2011/83/art_2/oj/fra/html` is the identifier of the Manifestation of a subdivision, and uses content negotiation to determine what should be returned (full content at requested anchor or content fragment only);
    a. By default, it redirects to the full content with the corresponding anchor (see below), that is `/eli/dir/2011/83/oj/fra/html#art_2`
    b. If the client adds the extra parameter "?content=fragment" at the end of the URI[7], then the URI redirect to the partial content URL (see below), that is `/eli/dir/2011/83/oj/fra/html/art_2`
2. The full content URL is `/eli/dir/2011/83/oj/fra/html` and contains HTML anchors so that a client can display the full content at any given anchor, e.g. `/eli/dir/2011/83/oj/fra/html#art_2`.
3. Each content fragment can be accessed individually by adding the subdivisions at the **end** of the full content URL, e.g. `/eli/dir/2011/83/oj/fra/`**`html/art_2`** – this would give the same result as /`eli/dir/2011/83/art_2/oj/fra/html`?content=fragment

The same behavior is expected for XML content.

11.4 Add some contexts to responses of subdivision fragments?

While it is out of scope of this specification, it is believed that, when returning the fragment corresponding to a subdivision, some additional context should also be provided in the response, in particular:

7 An extra HTTP request header "Accept: text/html;content=fragment" was also proposed, but at that time the CELLAR team preferred the solution with the extra parameter in the URI

- To get an understanding of the ancestor subdivisions (If a request is made to "art_2/par_2", and in the answer I see only "paragraph 2", how can I be sure it was really the second paragraph of article 2, and not 1 and will I remember that this snipped corresponds to article 2)
- To get an understanding of the version of the act the subdivision comes from (if a request is made to an "abstract" version, without specifying the version, how can I understand which concrete version of the document was returned?)

11.5 What if the user refers to a subdivision that does not exists?

- In the case of displaying the whole document at a given anchor, since the anchor does not exist, the page is simply displayed at the top.
- In the case of extracting the subdivision content, an error should be returned (typically 404, resource does not exist).

Part 3

Technical implementation

12 Mapping to Formex

12.1 Scope: excluding annexes

The mapping of ELI subdivisions URI to FORMEX includes everything in the body of the act, but excludes the annexes.

References to annexes themselves are in scope (anx_1)

12.2 Scope: Formex 4

The technical implementation of subdivision ELI in Formex is restricted to contents available in FORMEX 4 (after May 2004).

12.3 FMX to subdivision mapping table

FMX v4	Subdivision code[8]	FORMEX eIDs
//Root/TITLE/TI	TIT	tit_1
//Root/PREAMBLE	PBL	pbl_1
//Root/ENACTING.TERMS	ENC	enc_1
//Root/FINAL	FNP	fnp_1
//Root/PREAMBLE//VISA	CIT	cit_X (Numbered in sequence from 1 to N (last citation)
//Root/PREAMBLE//CONSID	RCT	rct_X (Numbered taking into account the its own numbering)
//Root/ENACTING.TERMS//ARTICLE	ART	art_X
//Root/ENACTING.TERMS//DIVISION[contains(TITLE/TI/text(),'PART')]	PRT	prt_X (Numbered taking into account the its own numbering. Also applicable to Chapter, Title, Section and subsection. This elements can be nested with lower high subdivisions)
//Root/ENACTING.TERMS//DIVISION[contains(TITLE/TI/text(),'CHAPTER')]	CPT	cpt_X
//Root/ENACTING.TERMS//DIVISION[contains(TITLE/TI/text(),'TITLE')]	TIS	tis_X
//Root/ENACTING.TERMS//DIVISION[contains(TITLE/TI/text(),'SECTION')]	SCT	sct_X
//Root/ENACTING.TERMS//DIVISION[contains(TITLE/TI/text(),'SUBSECTION')]	SBS	sbs_X (Can be preceded by eID of the previous high subdivisión)
//Root/ENACTING.TERMS//DIVISION/TITLE/STI	TIT	prt_X.tit_1, cpt_X.tit_1, tis_X.tit_1, sct_X.tit_1,

[8] Based on Authority Table "Subdivision": https://op.europa.eu/en/web/eu-vocabularies/at-dataset/-/resource/dataset/subdivision

		sbs_X.tit_1 (Numbered taking into account the its own numbering. Also applicable to Chapter, Title, Section and subsection This elements can be nested with lower high subdivisions)
//Root/ENACTING.TERMS//ARTICLE/STI.ART	TIT	art_X.tit_1
//Root/ENACTING.TERMS//ARTICLE/PARAG	PAR	art_X.par_X (Numbered taking into account its own numbering)
//Root/ENACTING.TERMS//ARTICLE/ALINEA	UNP	art_X.unp_X (Numbering from 1 to N)
//Root/ENACTING.TERMS//ARTICLE/PARAG/ALINEA	SUB	art_X.par_X.sub_X (Numbered from 1 to N)
//P|TXT[ends-with(.,':')] Followed by a LIST	INP	art_X.par_X.sub_X.inp art_X.unp_X.inp art_X.par_X.sub_X.pnt_X.inp art_X.unp_X.pnt_X.inp art_X.par_X.sub_X.idt_X.inp art_X.unp_X.idt_X.inp par, pnt are numbered taking into account its own numbering sub and unp and idt are numbered from 1 to N
//P[start-with(.,[a-z])] (starts with a small letter) Preceded by a LIST	WRP	art_X.par_X.sub_X.wrp_1 art_X.unp_X.wrp_1 art_X.par_X.sub_X.pnt_X.wrp_1 art_X.unp_X.pnt_X.wrp_1 art_X.par_X.sub_X.idt_X. wrp_1 art_X.unp_X.idt_X. wrp_1 par, pnt are numbered taking into account its own numbering sub and unp are numbered from 1 to N
//LIST[@type="arab|roman|alpha"]/ITEM	PNT	art_X.par_X.sub_X.pnt_X art_X.unp_X.pnt_X art_X.par_X.sub_X.pnt_X.pnt_X

		art_X.unp_X.pnt_X.pnt_X
//LIST[not(@type="arab|roman|alpha")]/ITEM	IDT	art_X.par_X.sub_X.idt_X art_X.unp_X.idt_X art_X.par_X.sub_X.idt_X.idt_X art_X.unp_X.idt_X.idt_X par, pnt are numbered taking into account its own numbering sub and unp and idt are numbered from 1 to N
//NOTE	NOT	eID_element_contaning_note.not_X (Numbered taking into account its own numbering.)

13 Mapping to AKN4EU

13.1 Scope: excluding subdivisions inside annexes

The mapping of ELI subdivisions URI to AKN4EU includes everything in the body of the act, but excludes the annexes.

References to annexes themselves are in scope (anx_1)

13.2 AKN to subdivision mapping table

AKN4EU structure	ELI subdivision
<title>	TIS
<part>	PRT
<chapter>	CPT
<section>	SCT
<subsection>	SBS
<division> or <subdivision>	?[9]
<article>	ART
<paragraph>[num]	PAR
<paragraph> [not(num)]	UNP
<level>	?[10]
<subparagraph>	SUB
<list>	SUB
<subparagraph refersTo="~_INP">	INP
<point>	PNT
<indent>	IDT

[9] Unnamed section
[10] pnt ?

<subparagraph refersTo="~_WRP">	WRP
<crossheading> <block name="crossheading">	TIT
<heading>	TIT
<preamble>	PBL
<body>	ENC
<doc name="ANNEX">	ANX
<conclusions>	FNP
<table>	TAB
<tr>	ROW
<td>	?
<authorialNote>	FTN
<formula name="actingEntity"	?
<formula name="enactingFormula"	?
<citations>	?
<citation>	CIT
<recitals>	?
<recital>	RCT
<longTitle>/<p> <longTitle>/<p>/docTitle>	TIT
<img>	IMG

14 Annex: Language mappings for subdivision IDs

The Annex documents how subdivision numbering works in different languages. The logic of the numbering is discussed, and in some cases a technical implementation of the algorithmic numbering is provided.

14.1 English Documents

Documents in English follow a predictable pattern for numbering. This can be replicated and implemented as an algorithmic sequence of indefinite length.

14.1.1 Logic

The logic is described below.

We start with the basic alphabet – A, B, C, D … to Z. Each letter corresponds to a position number i.e. A = position 1, B = position 2, C = position 3 …. Z = position 26.

The logic works as follows using the 26 position letters of the English alphabet.

- for position 1 to 26 we just number it with the alphabet at that position; So:
  - position 1 = a,
  - position 2 = b,
  - position 3 = c,
  - position 26 = z.
- for position 27 to 52 we prefix the sequence with 'a' and suffix it alphabetically just like we did for 1 to 26. So:
  - position 27 = position 1 + position 1 = a + a = aa
  - position 28 = position 1 + position 2 = a + b = ab
  - position 29 = position 1 + position 3 = a + c = ac
  - position 52 = position 1 + position 26 = a + z = az
- for position 53 to 81 we prefix the sequence with 'b' and and suffix it alphabetically just like we did for 1 to 26. So:
  - position 53 = position 2 + position 1 = b + a = ba
  - position 54 = position 2 + position 2 = b + b = bb
  - position 55 = position 2 + position 3 = b + c = bc
  - position 81 = position 2 + position 26 = b + z = bz

14.1.2 Technical

An implementation of the algorithm logic in python with an explanation has been published as a Jupyter notebook:

https://datalore.jetbrains.com/view/notebook/YEtWfIH38QKMefJzxXTZ5w

14.2 French and Italian Documents

French and Italian documents use two variations on Latin based numbering. Each of the variants is described here.

14.2.1 Variant 1

Variant 1 is not implementable as an algorithm.

The full series from 1 to 100 is provided below as a table:

1	bis
2	ter
3	quater
4	quinquies
5	sexies
6	septies
7	octies
8	nonies
9	decies
10	undecies
11	duodecies
12	terdecies
13	quaterdecies
14	quindecies
15	sexdecies
16	septdecies
17	octodecies
18	novodecies
19	vicies
20	unvicies
21	duovicies
22	tervicies
23	quatervicies

24	quinvicies
25	sexvicies
26	septvicies
27	octovicies
28	novovicies
29	tricies
30	untricies
31	duotricies
32	tertricies
33	quatertricies
34	quintricies
35	sextricies
36	septtricies
37	octotricies
38	novotricies
39	quadragies
40	unquadragies
41	duoquadragies
42	terquadragies
43	quaterquadragies
44	quinquadragies
45	sexquadragies
46	septquadragies
47	octoquadragies
48	novquadragies
49	quinquagies
50	unquinquagies
51	duoquinquagies
52	terquinquagies

Full list at the end of the Annex.

14.2.2 Variant 2

This variant is also based on latin numbering, but the syntax can be implemented as an algorithm. This is used in both Italian and French documents.

Subdvision numbering in French (variant 2) and Italian follows a sequential Latin numbering pattern which can be algorithmically expressed. It looks like this, positions 1 to 26 are numbered like this (translates to 2 times, 3 times, 4 times, etc.):

- bis
- ter
- quater
- quinquies
- sexies
- ...
- quinvicies
- sexvicies
- septvicies

position 27-52 onwards they are numbered like this:

- bis bis
- bis ter
- bis quater
- ...
- bis septivices

The full 1 to 26 series is shown below:

Num	Series
1	bis
2	ter
3	quater
4	quinquies

5	sexies
6	septies
7	octies
8	nonies
9	decies
10	undecies
11	duodecies
12	terdecies
13	quaterdecies
14	quindecies
15	sexdecies
16	septdecies
17	octodecies
18	novodecies
19	vicies
20	unvicies
21	duovicies
22	tervicies
23	quatervicies
24	quinvicies
25	sexvicies
26	septvicies

14.2.3 Logic

The logic is described below.

The logic works as follows using the 26 letters of the English alphabet.

- for 1 to 26 we just number it with the alphabet at that position; So:
  - position 1 = bis,
  - postion 2 = ter,
  - position 3 = quater,
  - position 26 = septvicies.
- for 27 to 52 we prefix the sequence with 'bis' and suffix it with positions 1 to 26. So:

        - position 27 = position 1 + position 1 = bis bis
        - position 28 = position 1 + position 2 = bis ter
        - position 29 = position 1 + position 3 = bis quater
        - position 52 = = position 1 + position 26 = bis septivicies
    - for 53 to 81 we prefix the sequence with 'ter' and and suffix it alphabetically just like we did for 1 to 26. So:
        - position 53 = position 2 + position 1 = ter bis
        - position 54 = position 2 + position 2 = ter ter
        - position 55 = position 2 + position 3 = ter quater
        - position 81 = position 2 + position 26 = ter septivicies

This logic can be implemented algorithmically for a very large sequence. The algorithm is implemented below in python, and the version presented can generate up to 676 sequential numbers in alphabetical form. The logic can however be adapted to sequences of indefinite length.

14.2.4 Technical

An implementation of the algorithm logic in python with an explanation has been published as a Jupyter notebook:

https://datalore.jetbrains.com/view/notebook/HfPyjotO04DxkcyzpcVpwg

14.2.5 Usage Note

The recommended variant to use is variant 2.

The usage of these variants is not entirely consistent. We recommend using variant 2 going ahead, and variant 1 should be treated only as legacy numbering found in a few existing documents. New documents should not be created with variant 1.

Example 1 - REGULATION (EU) 2019/876 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL

For this regulation – the Spanish and the French versions of the same regulation use different variants to the numbering.

Spanish version - uses variant 2 (L_2019150ES.01000101.xml (europa.eu)**)**

Artículo 325 ***bis quatervicies***

Correlaciones respecto del riesgo de tipo de cambio

Se aplicará un parámetro de correlación γ_{bc} uniforme igual al 60 % a la agregación de sensibilidades a factores de riesgo de tipo de cambio.

Subsección 2

Ponderaciones y correlaciones relativas al riesgo vega y de curvatura

Artículo 325 ***bis quinvicies***

Ponderaciones por riesgo vega y de curvatura

1. Los factores de riesgo vega utilizarán los segmentos relativos al delta a que se refiere la subsección 1.
2. La ponderación de riesgo correspondiente a un determinado factor de riesgo vega k (RW_k) se determinará como proporción del valor actual de ese factor de riesgo k, que representa la volatilidad implícita de un subyacente, tal como se describe en la sección 3.
3. La proporción a que se refiere el apartado 2 se hará depender de la liquidez supuesta de cada tipo de factor de riesgo con arreglo a la siguiente fórmula:

French version – uses variant 1 (L_2019150FR.01000101.xml (europa.eu))

Artículo 325 ***bis quatervicies***

Correlaciones respecto del riesgo de tipo de cambio

Se aplicará un parámetro de correlación γ_{bc} uniforme igual al 60 % a la agregación de sensibilidades a factores de riesgo de tipo de cambio.

Subsección 2

Ponderaciones y correlaciones relativas al riesgo vega y de curvatura

Artículo 325 ***bis quinvicies***

Ponderaciones por riesgo vega y de curvatura

1. Los factores de riesgo vega utilizarán los segmentos relativos al delta a que se refiere la subsección 1.
2. La ponderación de riesgo correspondiente a un determinado factor de riesgo vega k (RW_k) se determinará como proporción del valor actual de ese factor de riesgo k, que representa la volatilidad implícita de un subyacente, tal como se describe en la sección 3.
3. La proporción a que se refiere el apartado 2 se hará depender de la liquidez supuesta de cada tipo de factor de riesgo con arreglo a la siguiente fórmula:

Italian version – uses variant 1 (L_2019150IT.01000101.xml (europa.eu))

Articolo 325 quinquagies

Correlazioni per il rischio di cambio

All'aggregazione di sensibilità al fattore di rischio di cambio è applicato un parametro di correlazione uniforme γ_{bc} pari al 60 %.

Sottosezione 2

Fattori di ponderazione del rischio vega e di curvatura e correlazioni

Articolo 325 unquinquagies

Fattori di ponderazione del rischio vega e di curvatura

1. I fattori di ponderazione del rischio vega utilizzano le categorie delta di cui alla sottosezione 1.
2. Il fattore di ponderazione del rischio per un dato fattore di rischio vega k è determinato in percentuale del valore corrente di tale fattore di rischio k, che rappresenta la volatilità implicita di un sottostante, come descritta nella sezione 3.
3. La percentuale di cui al paragrafo 2 è subordinata alla presunta liquidità di ciascun tipo di fattore di rischio conformemente alla formula seguente:

Example 2 – RÈGLEMENT (CE) N o 1234/2007 DU CONSEIL

Older abrogated parts of this regulation make use variant 1, screenshot below.

Règlement (CE) nº 1234/2007	Présent règlement	Règlement (UE) nº 1306/2013
Article 118 *novovicies*	—	—
Article 119	—	—
Article 120	—	—
Article 120 *bis*	Article 81	—
Article 120 *ter*	—	—
Article 120 *quater*	Article 80	—
Article 120 *quinquies*, premier alinéa	Article 83, paragraphe 2	—
Article 120 *quinquies*, deuxième alinéa	[Article 223]	—

See L_2007299FR.01000101.xml (europa.eu) Règlement (UE) n ° 1308/2013 du Parlement européen et du Conseil du 17 décembre 2013 portant organisation commune des marchés des produits agricoles et abrogeant les règlements (CEE) n ° 922/72, (CEE) n ° 234/79, (CE) n ° 1037/2001 et (CE) n ° 1234/2007 du Conseil (europa.eu) (page 178)

The above appear to be the only cases where variant 1 (Latin based numbering) is used. In all other instances variant 2 has been used. There are cases of French legislation which use variant 1 numbering being referred to in European Parliament directives.

For example –

National transposition measures communicated by the Member States concerning: Directive (EU) 2018/844 of the European Parliament and of the Council of 30 May 2018 amending Directive 2010/31/EU on the energy performance of buildings and Directive 2012/27/EU on energy efficiency (Text with EEA relevance)

EUR-Lex - 32018L0844 - EN - EUR-Lex (europa.eu)

14.3 Portuguese Documents

Portuguese documents use a numbering scheme based on the English alphabet with an additional prefix in the front.

14.3.1 Logic

We start with the basic alphabet – A, B, C, D … to Z (in Capital letters). Each letter corresponds to a position number i.e. A = position 1, B = position 2, C = position 3 …. Z = position 26.

The logic works as follows using the 26 position letters of the English alphabet.

- for position 1 to 26 we just number it with the alphabet at that position; So:
    - position 1 = .º-A,
    - position 2 = .º-B,
    - position 3 = .º-C,
    - position 26 = .º-Z.
- for position 27 to 52 we prefix the sequence with '.º-A' and suffix it alphabetically just like we did for 1 to 26. So:
    - position 27 = position 1 + position 1 = .º-A + A = .º-AA
    - position 28 = position 1 + position 2 = .º-A + B = .º-AB
    - position 29 = position 1 + position 3 = .º-A + C = .º-AC
    - position 52 = position 1 + position 26 = .º-A + Z = .º-AZ
- for position 53 to 81 we prefix the sequence with 'b' and and suffix it alphabetically just like we did for 1 to 26. So:
    - position 53 = position 2 + position 1 = .º-B + A = .º-BA
    - position 54 = position 2 + position 2 = .º-B + B = .º-BB
    - position 55 = position 2 + position 3 = .º-B + C = .º-BC
    - position 81 = position 2 + position 26 = .º-B + Z = .º-BZ

14.3.2 Technical

See link to the Jupyter notebook:

https://datalore.jetbrains.com/view/notebook/BAg9zJZ5YJJnQys0CY7gkK

14.4 Greek Documents

Greek documents use a subdivision numbering logic which is distinct from English and Latin subdivision numbering. The logic is explained below.

14.4.1 Logic

The following are the Greek numerals from 1 to 10

Greek 1 to 9 [sequence 1]
α
β
γ
δ
ε
στ
ζ
η

θ

The following are Greek numbers for 10, 20, 30…100:

Greek 10 to 100 [sequence 2]
ι
κ
λ
μ
ν
ξ
ο
π
ϙ
σ

Greek Subdivision numbering uses:

- [sequence 1] for positions 1 to 9
- For position 10 uses the first item in [sequence 2], the Greek number for the number 10
- For position 11-19 suffixes each of the numerals from positions 1 to 9 to the Greek number 10
- For position 20 uses the second item in [sequence 2], the Greek number for the number 20
- For position 21-29 suffixes each of the numerals from positions 1 to 9 to the Greek number 20
- And so on

The entire 1 to 100 sequence of numbers is provided at the end of the Annex.

14.4.2 Technical

See link to the Jupyter notebook of the implementation:

https://datalore.jetbrains.com/view/notebook/PWOKxbcVGjez0xerM6s8YL

14.5 Bulgarian Documents

Bulgarian documents use Cyrillic numerals for subdivision numbering.

14.5.1 Logic

The logic for the Bulgarian subdivision numbering is similar to English.

Below are presented the Bulgarian (Cyrillic) numerals from 1 to 26

Bulgarian numerals
а
б
в
г
д
е
ж
з
и
й
к
л
м
н
о
п
р
с
т
у
ф
х
ц
ч
ш

Щ

- for position 1 to 26 we just number it with the alphabet at that position; So:
    - position 1 = а,
    - position 2 = б,
    - position 3 = в,
    - position 26 = Щ.
- for position 27 to 52 we prefix the sequence with 'а' and suffix it alphabetically just like we did for 1 to 26. So:
    - position 27 = position 1 + position 1 = а + а = аа
    - position 28 = position 1 + position 2 = а + б = аб
    - position 29 = position 1 + position 3 = а + в = ав
    - position 52 = position 1 + position 26 = а + Щ = аЩ
- for position 53 to 81 we prefix the sequence with ' б ' and and suffix it alphabetically just like we did for 1 to 26, and so on...

14.5.2 Technical

See link to the Jupyter notebook below:

https://datalore.jetbrains.com/view/notebook/oTwXv7A8qBDNENRZor0Kmx

14.6 Subdivision Numbering Table

The full table with all the subdivision numbers from 1 to 100 is presented below:

No.	EN	FR, IT, ES *	FR, IT, ES	PT	EL	BG
1	a	bis	bis	.º-A	α	а
2	b	ter	ter	.º-B	β	б
3	c	quater	quater	.º-C	γ	в
4	d	quinquies	quinquies	.º-D	δ	г
5	e	sexies	sexies	.º-E	ε	д
6	f	septies	septies	.º-F	στ	е
7	g	octies	octies	.º-G	ζ	ж
8	h	nonies	nonies	.º-H	η	з
9	i	decies	decies	.º-I	θ	и
10	j	undecies	undecies	.º-J	ι	й
11	k	duodecies	duodecies	.º-K	ια	к
12	l	terdecies	terdecies	.º-L	ιβ	л
13	m	quaterdecies	quaterdecies	.º-M	ιγ	м
14	n	quindecies	quindecies	.º-N	ιδ	н
15	o	sexdecies	sexdecies	.º-O	ιε	о
16	p	septdecies	septdecies	.º-P	ιστ	п
17	q	octodecies	octodecies	.º-Q	ιζ	р
18	r	novodecies	novodecies	.º-R	ιη	с
19	s	vicies	vicies	.º-S	ιθ	т
20	t	unvicies	unvicies	.º-T	κ	у
21	u	duovicies	duovicies	.º-U	κα	ф
22	v	tervicies	tervicies	.º-V	κβ	х
23	w	quatervicies	quatervicies	.º-W	κγ	ц
24	x	quinvicies	quinvicies	.º-X	κδ	ч
25	y	sexvicies	sexvicies	.º-Y	κε	ш

26	z	septvicies	septvicies	.°-Z	κστ	щ
27	aa	octovicies	bis bis	.°-AA	κζ	аа
28	ab	novovicies	bis ter	.°-AB	κη	аб
29	ac	tricies	bis quater	.°-AC	κθ	ав
30	ad	untricies	bis quinquies	.°-AD	λ	аг
31	ae	duotricies	bis sexies	.°-AE	λα	ад
32	af	tertricies	bis septies	.°-AF	λβ	ае
33	ag	quatertricies	bis octies	.°-AG	λγ	аж
34	ah	quintricies	bis nonies	.°-AH	λδ	аз
35	ai	sextricies	bis decies	.°-AI	λε	аи
36	aj	septtricies	bis undecies	.°-AJ	λστ	ай
37	ak	octotricies	bis duodecies	.°-AK	λζ	ак
38	al	novotricies	bis terdecies	.°-AL	λη	ал
39	am	quadragies	bis quaterdecies	.°-AM	λθ	ам
40	an	unquadragies	bis quindecies	.°-AN	μ	ан
41	ao	duoquadragies	bis sexdecies	.°-AO	μα	ао
42	ap	terquadragies	bis septdecies	.°-AP	μβ	ап
43	aq	quaterquadragies	bis octodecies	.°-AQ	μγ	ар
44	ar	quinquadragies	bis novodecies	.°-AR	μδ	ас
45	as	sexquadragies	bis vicies	.°-AS	με	ат
46	at	septquadragies	bis unvicies	.°-AT	μστ	ау
47	au	octoquadragies	bis duovicies	.°-AU	μζ	аф
48	av	novquadragies	bis tervicies	.°-AV	μη	ах
49	aw	quinquagies	bis quatervicies	.°-AW	μθ	ац
50	ax	unquinquagies	bis quinvicies	.°-AX	ν	ач
51	ay	duoquinquagies	bis sexvicies	.°-AY	να	аш
52	az	terquinquagies	bis septvicies	.°-AZ	νβ	ащ
53	ba	quaterquinquagies	ter bis	.°-BA	νγ	ба
54	bb	quinquinquagies	ter ter	.°-BB	νδ	бб
55	bc	sexquinquagies	ter quater	.°-BC	νε	бв

56	bd	septquinquagies	ter quinquies	.°-BD	νστ	бг
57	be	octoquinquagies	ter sexies	.°-BE	νζ	бд
58	bf	novoquinquagies	ter septies	.°-BF	νη	бе
59	bg	sexagies	ter octies	.°-BG	νθ	бж
60	bh	unsexagies	ter nonies	.°-BH	ξ	бз
61	bi	duosexagies	ter decies	.°-BI	ξα	би
62	bj	tersexagies	ter undecies	.°-BJ	ξβ	бй
63	bk	quatersexagies	ter duodecies	.°-BK	ξγ	бк
64	bl	quinsexagies	ter terdecies	.°-BL	ξδ	бл
65	bm	sexsexagies	ter quaterdecies	.°-BM	ξε	бм
66	bn	septsexagies	ter quindecies	.°-BN	ξστ	бн
67	bo	octosexagies	ter sexdecies	.°-BO	ξζ	бо
68	bp	novosexagies	ter septdecies	.°-BP	ξη	бп
69	bq	septuagies	ter octodecies	.°-BQ	ξθ	бр
70	br	unseptuagies	ter novodecies	.°-BR	ο	бс
71	bs	duoseptuagies	ter vicies	.°-BS	οα	бт
72	bt	terseptuagies	ter unvicies	.°-BT	οβ	бу
73	bu	quaterseptuagies	ter duovicies	.°-BU	ογ	бф
74	bv	quinseptuagies	ter tervicies	.°-BV	οδ	бх
75	bw	sexseptuagies	ter quatervicies	.°-BW	οε	бц
76	bx	septseptuagies	ter quinvicies	.°-BX	οστ	бч
77	by	octoseptuagies	ter sexvicies	.°-BY	οζ	бш
78	bz	novoseptuagies	ter septvicies	.°-BZ	οη	бщ
79	ca	octogies	quater bis	.°-CA	οθ	ва
80	cb	unoctogies	quater ter	.°-CB	π	вб
81	cc	duooctogies	quater quater	.°-CC	πα	вв
82	cd	teroctogies	quater quinquies	.°-CD	πβ	вг
83	ce	quateroctogies	quater sexies	.°-CE	πγ	вд
84	cf	quinoctogies	quater septies	.°-CF	πδ	ве
85	cg	sexoctogies	quater octies	.°-CG	πε	вж

86	ch	septoctogies	quater nonies	.º-CH	πστ	вз
87	ci	octooctogies	quater decies	.º-CI	πζ	ви
88	cj	novooctogies	quater undecies	.º-CJ	πη	вй
89	ck	nonagies	quater duodecies	.º-CK	πθ	вк
90	cl	unnonagies	quater terdecies	.º-CL	ϙ	вл
91	cm	duononagies	quater quaterdecies	.º-CM	ϙα	вм
92	cn	ternonagies	quater quindecies	.º-CN	ϙβ	вн
93	co	quaternonagies	quater sexdecies	.º-CO	ϙγ	во
94	cp	quinnonagies	quater septdecies	.º-CP	ϙδ	вп
95	cq	sexnonagies	quater octodecies	.º-CQ	ϙε	вр
96	cr	septnonagies	quater novodecies	.º-CR	ϙστ	вс
97	cs	octononagies	quater vicies	.º-CS	ϙζ	вт
98	ct	novononagies	quater unvicies	.º-CT	ϙη	ву
99	cu	centies	quater duovicies	.º-CU	ϙθ	вф
100	cv	uncenties	quater tervicies	.º-CV	σ	вх

*- used in in very few documents