OASIS

Akoma Ntoso Naming Convention Version 1.0

Committee Specification Draft ~~01~~02 /
Public Review Draft ~~01~~02

~~14 January 2015~~

04 May 2016

Specification URIs
This version:
> http://docs.oasis-open.org/legaldocml/akn-nc/v1.0/csprd02/akn-nc-v1.0-csprd02.html (Authoritative)
> http://docs.oasis-open.org/legaldocml/akn-nc/v1.0/csprd02/akn-nc-v1.0-csprd02.doc
> http://docs.oasis-open.org/legaldocml/akn-nc/v1.0/csprd02/akn-nc-v1.0-csprd02.pdf

Previous version:
> http://docs.oasis-open.org/legaldocml/akn-nc/v1.0/csprd01/akn-nc-v1.0-csprd01.html (Authoritative)
> http://docs.oasis-open.org/legaldocml/akn-nc/v1.0/csprd01/akn-nc-v1.0-csprd01.doc
> http://docs.oasis-open.org/legaldocml/akn-nc/v1.0/csprd01/akn-nc-v1.0-csprd01.pdf

~~Previous version:~~
> ~~N/A~~

Latest version:
> http://docs.oasis-open.org/legaldocml/akn-nc/v1.0/akn-nc-v1.0.html (Authoritative)
> http://docs.oasis-open.org/legaldocml/akn-nc/v1.0/akn-nc-v1.0.doc
> http://docs.oasis-open.org/legaldocml/akn-nc/v1.0/akn-nc-v1.0.pdf

Technical Committee:
> OASIS LegalDocumentML (LegalDocML) TC

Chairs:
> Fabio Vitali (fabio@cs.unibo.it), University of Bologna-CIRSFID
> Monica Palmirani (monica.palmirani@unibo.it), University of Bologna-CIRSFID

Editors:
> Fabio Vitali (fabio@cs.unibo.it), University of Bologna-CIRSFID
> Monica Palmirani (monica.palmirani@unibo.it), University of Bologna-CIRSFID
> Véronique Parisse (V.PARISSE@aubay.lu), Aubay S.A.

Related work:
> This specification is related to:

- *Akoma Ntoso Version 1.0 Part 1: XML Vocabulary*. http://docs.oasis-open.org/legaldocml/akn-core/v1.0/csprd02/part1-vocabulary/akn-core-v1.0-csprd02-part1-vocabulary.html.
- *Akoma Ntoso Version 1.0 Part 2: Specifications*. http://docs.oasis-open.org/legaldocml/akn-core/v1.0/csprd02/part2-specs/akn-core-v1.0-csprd02-part2-specs.html.
- Akoma Ntoso Version 1.0 XML schemas. http://docs.oasis-open.org/legaldocml/akn-core/v1.0/csprd02/part2-specs/schemas/.
- Akoma Ntoso Version 1.0 XML examples. http://docs.oasis-open.org/legaldocml/akn-core/v1.0/csprd02/part2-specs/examples/.
- *Akomo Ntoso: XML for parliamentary, legislative & judiciary documents*. http://www.akomantoso.org.

Abstract:

This document provides the naming convention for defining IRIs and ids related to the Akoma Ntoso XML standard. Within the schema of Akoma Ntoso, id attributes are declared as optional, but whenever attributes `eId` and `wId` are actually used the specifications in this document are mandatory.

Status:

This document was last revised or approved by the OASIS LegalDocumentML TC on the above date. The level of approval is also listed above. Check the "Latest version" location noted above for possible later revisions of this document. Any other numbered Versions and other technical work produced by the Technical Committee (TC) are listed at https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=legaldocml#technical.

TC members should send comments on this specification to the TC's email list. Others should send comments to the TC's public comment list, after subscribing to it by following the instructions at the "Send A Comment" button on the TC's web page at https://www.oasis-open.org/committees/legaldocml/.

For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the TC's web page (https://www.oasis-open.org/committees/legaldocml/ipr.php).

Citation format:

When referencing this specification the following citation format should be used:

[AkomaNtosoNaming-v1.0]

Akoma Ntoso Naming Convention Version 1.0. Edited by Fabio Vitali, Monica Palmirani, and Véronique Parisse. 14 January 2015.04 May 2016. OASIS Committee Specification Draft 0102 / Public Review Draft 0102. http://docs.oasis-open.org/legaldocml/akn-nc/v1.0/csprd02/akn-nc-v1.0-csprd02.html. Latest version: http://docs.oasis-open.org/legaldocml/akn-nc/v1.0/akn-nc-v1.0.html.

Notices

Table of Contents

1. Introduction

1.1 Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in **[RFC2119]**.

1.2 Normative References

[RFC2119]   Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997. http://www.ietf.org/rfc/rfc2119.txt.

[IRI]   International Resource Identifiers as per RFC 3987 (http://tools.ietf.org/html/rfc3987).

[ISO3166]   ISO 3166. (http://www.iso.org/iso/home/standards/country_codes/iso-3166-1_decoding_table.htm).

[ISO639-2]   ISO 639-2 alpha-3. (http://www.loc.gov/standards/iso639-2/).

1.3 Non-Normative References

[FRBR]   Functional requirements for bibliographic records: final report / IFLA Study Group on the Functional Requirements for Bibliographic Records. — München: K.G. Saur, 1998. — viii, 136 p. — (UBCIM publications; new series, vol. 19). — ISBN 978-3-598-11382-6. http://www.ifla.org/files/assets/cataloguing/frbr/frbr_2008.pdf.

[AkomaNtosoNaming-v1.0]   *Akoma Ntoso Naming Convention Version 1.0*. Edited by Véronique Parisse, Monica Palmirani, Fabio Vitali. OASIS Committee Specification Draft 01. http://docs.oasis-open.org/legaldocml/akn-nc/v1.0/csd01/akn-nc-v1.0-csd01.html. Latest version: http://docs.oasis-open.org/legaldocml/akn-nc/v1.0/akn-nc-v1.0.html.

1.4 Status

The present specification defines the naming convention that needs to be implemented in order to conform to the second level of compliance with the Akoma Ntoso schema.

In this specification, when MUST is used in the text, it MUST be understood as "in order to conform to level 2 of compliance with the Akoma Ntoso schema."".

2. Context

2.1 The Importance of Text Identification in Legislation

In HTML, the primary link type is ~~the~~ *anchor-to-document*, while ~~the~~ *anchor-to-anchor* ~~link is~~links are a minor addition for uncharacteristic cases ~~(and highly criticized by usability experts).~~. For this reason ~~is why~~ identifiers are never required~~.:~~ ~~Aa~~uthors are expected to provide identifiers only for those structures that are likely destinations of anchor-to-anchor links – ~~such as~~often just a few section headings.

In legislation, on the other hand, ALL references are to a precise substructure of a highly hierarchical document flow, and any substructure may become a destination. This is the reason why identifiers are ~~required~~useful for most ~~elements~~structures of a legislative document.

Additionally, in ~~the second level of compliance.~~
~~In~~ HTML~~,~~ the reference is usually meant for navigation by human users. It is only necessary to come close enough to the intended destination that a human eye can scan the surrounding text or elements and find the exact destination in the vicinity.

In legislation, ~~we have~~there is an additional type of reference, ~~that of "modifications~~"modification". Modifications require that a specific substructure be precisely identified and modified by a modification instruction. In this case, one cannot be satisfied with the fact that the intended destination is somewhat near the arrived destination -- they must coincide.

By using the layering provided by the Functional Requirements for Bibliographic Records (FRBR) ~~layering~~ (See Section ~~5~~4.1), ~~we are~~Akoma Ntoso strongly differenti~~at~~inges between the legislative context and the markup used to represent it. References are legislative concepts, and exist regardless of whether they exist in the markup. The same content, for instance, could be represented in a number of different XML files created by various authors. They would all be distinct manifestations of the same Expression, each of which may have the same body, but different markup choices, metadata, commentary, etc. References would need to work regardless of the specific Manifestation chosen as the destination, and, indeed, it is important that all manifestations use the same identifiers for the same structures, standardized by the LegalDocML Technical Committee. This is ~~imp~~affected by the fact that ~~a user~~an author may not even have the ~~XML~~document of the destination, or that it may not even exist yet (time-based alchemies frequently occur in legislation, one might need to create links to documents that have yet to be converted into Akoma Ntoso, etc.). Thus, providing a forced and precise syntax for identifiers is the best guarantee that all different manifestations of the same content have the same identifiers and that one does not need to read an XML file to divine the values of its identifiers.

Legal references have peculiar traits regarding time. For instance, in the case of an evolving document (e.g., a piece of legislation receiving references and being actively modified by the legislator), the actual destination of the reference is neither the original version, nor the current version, but in many cases the version of the document that was valid at the moment in time when the case took place. References are dynamic rather than static, because the destination moves in time and jurisdiction according to the needs rather than being fixed to a specific sentence or fragment. This means that point-in-time consolidation is an important affair, and that determining the destination of a dynamic link requires, at the very least, that structures existing in multiple versions are named consistently. ~~It~~For example, it must be clear that, if section 35 of the initial version of a title of a U.S. code had some identifier ~~Y~~y, then ALL subsequent versions of that same section 35 (even after a renumbering action) have the same identifier ~~Y~~y, so that once you determine the needed version, arriving at the right structure is easy and straightforward.

The syntax of identifiers is defined to ensure that identifiers can be used regardless of the versions of the same document, regardless of the author of individual XML markups, regardless of usage of navigational or modification references, and knowing full well that point-to-point references are the norm rather than the exception.

The Akoma Ntoso naming convention does not assume that the document is stored in Akoma Ntoso XML, but only that there is a mapping between the FRBR IRI reference and the URL of a file stored somewhere on the Internet, and to which our ~~URI~~reference can be resolved into.

In the case of identifiers, the Akoma Ntoso naming convention does not assume, yet again, that the document is stored in Akoma Ntoso XML, but only that identifiers work in whatever format has been used.

This means that any XML-based language, including Akoma Ntoso, HTML, TEI, DocBook, ePub, kf8, or Mobi are acceptable~~, while PDF or Microsoft Word are not so acceptable~~ and can make use of the Naming Convention presented here.

The Akoma Ntoso Naming convention also assumes that it is the job of the author of the linked-to document to use identifiers that are consistent with the naming convention. This is necessary because, in HTTP, the fragment identifier is never sent with the request and is only known and handled by the user agent, so we must assume that identifiers are present in the response and have the correct form. ~~There is very little we can do otherwise.~~ In particular, it would make no sense to convert all fragment identifiers in references using the syntax of the destination documents, as these syntaxes can be quite innumerable.

Identifiers are the main way ~~to identify~~Akoma Ntoso identifies fragments and parts of the document in an unambiguous form. They can be used in document references (e.g., links and amendment commands) as a precise pointer to the actual part of the document mentioned (as opposed to simply referring to a document as a whole).

Identifiers ~~can be~~are systematically used in Akoma Ntoso. All Akoma Ntoso elements allow up to three identifiers. Even internal links need to use identifiers. Most relevant elements and sections require at least one identifier.

3. Scope

The Akoma Ntoso naming convention identifies, in a unique way, all Akoma Ntoso concepts and resources on the Internet and, in general, all collections thereof. These principles and characteristics should be respected in the naming convention:

1. *MEANINGFULNESS*: the name is a meaningful and logical description of the resource and not of its physical path.

2. *PERMANENCE*: the name must be permanent and stable over time.

3. *INVARIANCE*: the name must derive from invariant properties of the resource so as to provide some degree of certainty in obtaining the same name for the same resource regardless of process, tool and person.

FRBR concepts are used differently when referring to documents in a variety of situations. In each case it is important to use the IRI for the correct FRBR level of document. Here, we describe a few particularly frequent situations:

1. Legislative references will most probably refer to Works --: acts referring to other acts do so regardless of the actual version, and references must be to something independent of all possible expressions, e.g., to the Work.

2. The list of attachments and schedules belong to a specific Expression, so references to ExpressionComponents are specific to the Expression-level.

3. The specific Manifestation that is the Akoma Ntoso XML format uses an XML-based syntax to refer to ExpressionComponents, and associate them to the corresponding ManifestationComponents containing the appropriate content. Therefore, within XML files the IRI of the ManifestationComponents must be used to refer to all components including the main document, all attachments, and all schedules.

4. Multimedia fragments within an XML Manifestation (e.g., a drawing, a schema, or a map.) do not exist as independent ExpressionComponents as they are only a part of some ExpressionComponent (even when they are the only part). In fact, they are only ManifestationComponents and as such are referred to in `<object>` and `<img>` elements with the appropriate ManifestationComponent IRI. Even if the same multimedia content appears in different parts of the content of a Manifestation, each instance of that content must correspond to a different ManifestationComponent, and must be considered independently of the other.

5. It is an Item-level decision, once ascertained that the content is exactly identical, to provide space-saving policies by storing only one copy of the multimedia content. This Item-level decision has no impact on references and names, which are still individually different from each other.

6. Non-document concepts are referred to within the metadata and content of Akoma Ntoso documents. References are always performed in two steps: the first step ties the reference point in the document to an item in the References section using internal (and not standardized) identifiers; the second step ties the item in the reference section to the actual concept through the IRI of the concept as specified in this ~~document~~Naming Convention.

Since the most primary concepts in Akoma Ntoso are connected to documents, the main part of this section is devoted to detailing the IRIs of document-related concepts, and in particular Works, Expressions, and Manifestations. Items are, by definition, outside of the scope of this standard and are only briefly described. The final part of the section provides an IRI-based naming mechanism for non-document entities (as well as for document entities when they are handled in a similar way to non-document entities).

4. IRI[1] (Normative)

4.1 Document IRIs

All resources are identified by a unique name. Resources are categorized as Work, Expression, Manifestation and Item, and each of these categories has a different naming structure. The actual syntax of the resource is specified in the following section, the "AKOMA NTOSO Naming Convention", which is an integral part of the Akoma Ntoso standard.

The Akoma Ntoso standard defines a number of referenceable concepts that are used in many situations in the lifecycle of legal documents. The purpose of this section is to provide a standard referencing mechanism to these concepts through the use of IRI references associated to classes and instances of an *ad hoc* ontology. The referencing mechanism discussed in this document is meant to be generic and evolving with the evolution of the underlying ontology.

The most important concepts of the Akoma Ntoso ontology are related to documents that have legal status. All discourses and all descriptions of legal sources can be characterized as referring to one of the four levels of a document as introduced by IFLA FRBR *(International Federation of Library Associations (IFLA) - Functional Requirements for Bibliographic Records (FRBR))*:(**[FRBR]**):

   (a) Work – the abstract concept of the legal resource (e.g., act 3 of 2015).

   (b) Expression - any version of the Work whose content is specified and different from others for any reason: language, versions, etc. (e.g., act 3 of 2015 as in the version following the amendments entered into force on July 3rd, 2016).

   (c) Manifestation - any electronic or physical format of the Expression: MS Word, Open Office, XML, TIFF, PDF, etc. (e.g., PDF representation of act 3 of 2015 as in the version following the amendments entered into force on July 3rd, 2016).

   (d) Item – the physical copy of any Manifestation in the form of a file stored somewhere in some computer on the network or disconnected (e.g., the file called act32015.pdf on my computer containing a PDF representation of act 3, 2015 as in the version following the amendments entered into force on July 3rd, 2016).

All documents at all levels can be composed of sub-elements that, when combined, form the whole document. These are called components and abstractly represent the notion that several independent *subdocuments* form the whole document as it appears to the reader (i.e., a main body possibly followed by a number of attachments such as schedules and tables):

   - WorkComponents (e.g., main, schedule, annexes, table) - the WorkComponents are abstract entities that can be referenced to refer to different ExpressionComponents in time.

   - ExpressionComponent (e.g., main, schedule, annexes, or table.) - the ExpressionComponents represent the visible division of the document as generated by the content author (Parliament, etc.) for a specific Expression of the Work.

   - ManifestationComponent (e.g., xml files, PDF files, or TIFF images.) - the ManifestationComponents represent the division of the document as generated by the Manifestation author (e.g., the XML editor.) for a specific Manifestation of a specific Expression.

   - ItemComponent - the actual files corresponding to the ManifestationComponents

Other concepts dealt by the Akoma Ntoso ontology also derive from the IFLA FRBR ontology, and include, but are not limited to, individuals (Person), organizations (Corporate Body), actions and occurrences (Event), locations (Place), ideas (Concept) and physical objects (Object).

---

[1] IRI: International Resource Identifiers as per RFC 3987 (). IRI: see **[IRI].**

4.2 Absolute and Relative IRIs

At all levels, the Akoma Ntoso IRIs belong to the http:// scheme and are normally resolved using mechanisms widely available in browsers and web servers. Within documents, IRIs are used as references to addressable resources, and are thus called *IRI references*.

According to the authoritative source RFC 3986[2], all http:// IRI references are divided into *absolute IRI* and *relative IRI references*. An absolute IRI starts with the string "http://", which is then followed by an officially registered domain name, and the local part that starts off the first individual "/" character. A relative reference, on the other hand, has no indication of the scheme, no indication of the domain name, and may have further missing parts at the beginning of the whole string (no missing parts on the end, though). Browsers are able to build the absolute IRI corresponding to the relative IRI by adding at the beginning of the provided IRI the missing parts that are taken from the IRI of a base resource.

In XML manifestations of Akoma Ntoso documents, IRI references shallare always be expressed in relative forms.

This implies that any resolution is carried out by the source of the base document (e.g., the one where the IRI reference is stored). This makes all IRIs independent of the actual resolution mechanism, and allows for very flexible storage, access, and reference mechanisms. This means that all resolution mechanisms used to access an Akoma Ntoso document from another Akoma Ntoso document will rely on the same resolution mechanism as the original one, regardless of the resolution mechanism employed to generate the documents themselves. In case the hosting document lacks a base IRI, it is the responsibility of the active application to provide a base IRI in its stead.

Since it is a requirement of Akoma Ntoso that all existing FRBR items of a Manifestation be are byte-per-byte identical to each other, it is a natural consequence that it is not abstractly relevant which resolution engine dereferences the actual Item whose IRI is resolved out of a Work-level, an Expression-level, or a Manifestation-level IRI reference. This, in practice, means that protocol and authority are , in resolution, not contributing information, to the resolution, and are thus interchangeable. Any partyParties interested in absolute IRIs for Akoma Ntoso are required to produce their own resolution engine and use its protocol and authority for the purpose.

Another distinction is between *global* and *local* IRI references[3]. A global IRI refreference is a relative IRI refreference where all parts are present except for protocol and authority (i.e., domain name). Thus, a global IRI refreference *always* starts with a slash, to indicate that *all* other parts are explicitly specified. A local IRI refreference, on the other hand, may have one or more parts missing (necessarily from left to right), and the corresponding global (and, subsequently, absolute) IRI reference is determined by adding the corresponding parts taken from the base document, as usual with relative IRI references with missing parts. In the following we will call all IRI references as simply IRI (they are *all* references, after all), and distinguish between absolute IRIs, global IRIs and local IRIs.

In XML manifestations of Akoma Ntoso documents, all Work, Expression, and Manifestation-level references to whole documents must be *global*, and all references to individual components within the same level (or lower levels) must be *local* and are stored simply as the name of the corresponding component.

Thus, for instance, "/akn/kn/act/200715-01-01/1/schedule1/!schedule_1" is the relative, global Work-level IRI for schedule 1 of act 1/2007 of Kenya. However, a Work-level reference to schedule 1 placed within the main document of the act will only contain the local IRI "1/!schedule1". This guarantees that these references continue to work even after new expressions are created of the same Work, either if the part containing the reference is changed or if it remains untouched.

Akoma Ntoso XML elements refer to other documents according to different levels of the FRBR hierarchy. In particular, `<ref>`, `<mref>`, and `<rref>` point to Work-level and occasionally Expression-level IRIs only, while `<object>`, `<img>`, and `<attachment>` always point to Manifestation-level IRIs. As the global/local distinction is involved, `<ref>`, `<mref>, and <rref>` elements always use global IRIs for

---

[2] http://tools.ietf.org/html/rfc3986

[3] in fact, this is a simplification of RFC 3986, that calls global IRI refs as "absolute path references" and local IRI refs as "relative path references".

documents different than the host, while <img> or <attachment> always refer to components of the host document, and thus always use local references.

A In the Akoma Ntoso Naming Convention, a reference to a different actdocument is always global:

```
<ref href="/akn/kn/act/200615-08-10/123#/!main~sec_12">section 12 of act
13/2006</ref>
```

A reference to a specific attachment of the same act is always local:

```
<ref href="schedule01#para_12">paragraph 12 of schedule 1 of this act</ref>
```

Analogously, multimedia fragments (e.g., images) within the main document are specified using aThis means that whenever referring to a different document it is necessary to provide the full list of identification parts even when they are identical to the ones of the base document. On the other hand, local IRI references are used in the Akoma Ntoso naming convention for references to components or fragments, and implicitly make use of the identification parts of the base document for the correct resolution of the IRI reference.

Global IRI references are discussed in sections 4.4 through 4.8, while local IRI:

```
<img src="media/logo.tiff"/>
```

The only exception to this rule is for external attachments, (i.e., components that are external to the Akoma Ntoso XML package).

In general, all Manifestation components are stored within a package, and thus have an IRI that is very similar to that of the Manifestation itself. Sometimes, though, it may be appropriate to store the individual component elsewhere as an independent document. Such a situation may arise, for instance, when a document specifies another full document as one of its attachments, e.g., a ratification decree placing an international treaty as an attachment. Since it is more appropriate to consider the important document the international treaty, it will constitute a Work on its own and have its own IRI of a completely different form than that of the attachment would have.

In cases where components references are not stored within a package, it is more appropriate that all references to the external attachment are global at the Work-level as well as at the Expression and Manifestation-level. Furthermore, in the cases where we have external attachments, the <attachment> and <attachmentOf> elements of the Referencesdiscussed in section need to be used. In fact, these two elements are ONLY and ALWAYS to be used for external attachments4.9.

4.3 Resolving Akoma Ntoso IRIsIRI references

The Akoma Ntoso naming architecture is built so as not to rely on the existence of a single storage architecture, since the IRIs stored within documents are differentiated from the ones physically representing the resource being sought.

The mapping from architecture-independent IRIs into accessible architecture-dependent URLs (representing the best Item for the document being sought) areis realized through specific applications called *IRI resolvers*. The Akoma Ntoso naming architecture is built so as not to rely on the existence of any individual IRI resolver, but assumes that all IRIs are always correctly resolved to the best available Item regardless of the resolving mechanisms. In fact, each naming authority is given the global task of resolving any possible Akoma Ntoso IRIs, regardless of whether it belongs or not to the country or countries managed by the naming authority. This implies that the authority-specific details of IRIs are purposefully omitted in this specification, and need to be considered only when first accessing a document.

For this reason, all IRIs in this specification are prefixed with the arbitrary domain name *[http://www.authority.org]* that stands for any of an arbitrarily large number of equivalent naming authorities.

4.4 The IRI for fragment specifications

Legal citations usually point to a specific fragment of the abstract document as shown in this text:

"Article 3 of Directive 2003/87/EC"

The correct syntax for specifying the fragment using Akoma Ntoso IRSs, in accordance with the protocol specified by the HTTP specifications, is as follows:

- [http://www.authority.org]/akn/eu/act/2003-11-13/87#art_3
  Fragment article 3 of the Work of the European Directive 2003/87/EC

- [http://www.authority.org]/akn/eu/act/2003-11-13/87/eng@#art_3
  Fragment article 3 of the Expression of the European Directive 2003/87/EC in English

- [http://www.authority.org]/akn/eu/act/2003-11-13/87/eng@2015-01-20#art_3
  Fragment article 3 of the Expression of the European Directive 2003/87/EC in English at the version dated 2015-01-20

- [http://www.authority.org]/akn/eu/act/2003-11-13/87/eng@2015-01-20/main.xml#art_3
  Fragment article 3 of the XML Manifestation of the European Directive 2003/87/EC in English at the version dated 2015-01-20.

This syntax permits the server to return the whole document, while the fragment information is used only by the client.

For permitting the specification of a query to a portion in AKN IRIs, the correct syntax is specified below (cf. for the IRI for the work-level portion query and for the IRI for the Expression-level portion query).

4.5 The IRI of a Work

4.5.1 The IRI for the Work as a Whole

The IRIreference for the Work is the baseline for building the IRI reference for the Expression, (section 4.5), which is the baseline for the IRI of the Manifestation.reference of the Manifestation (section 4.6). Additionally, all IRI references can be enriched with information about the component and the fragment requested (section 4.8). IRI references for items are beyond the scope of this Naming Convention.

4.4 The IRI reference of a Work

The IRI for the Work consists of the following pieces., separated by forward slashes "/":

- The base URL of a naming authority with IRI-resolving capabilities *(not relevant for the Naming Convention)).* In all our examples this will always be described as *[http://www.authority.org],* as explained in section 4.3.

- A detail fragment that organizes additional data in a hierarchical fashion:

  – The /akn prefix to allow the identification of the IRI as belonging to the Akoma Ntoso Naming Convention (required)

  – Country or subdivision (a two-letter or code according to ISO 3166-1 or a four-letter code[ISO3166] or a short and unique alphanumeric codes according to ISO 3166-2). [ISO3166-2]). For an Akoma Ntoso XML representation, this value mustMUST correspond to the content of the element <FRBRcountry> in the metadata. (required)

  – Type of document. For an Akoma Ntoso XML representation, this value mustMUST correspond to the element immediately below the akomaNtoso root element (e.g., act, bill, or debateReport).,.) (required)

- Any specification of document subtype, if appropriate. For an Akoma Ntoso XML representation, this value ~~must~~MUST correspond to the content of the element `<FRBRsubtype>` in the metadata~~.~~ or, in its absence, to the "name" attribute of the document type (optional).
- The emanating actor, unless implicitly deducible by the document type (e.g., acts and bills do not usually require actor, while ministerial decrees and European legislation do). For an Akoma Ntoso XML representation, this value ~~must~~MUST correspond to the content of the element `<FRBRauthor>` in the `<FRBRWork>` section of the metadata~~.~~ (optional).
- Original creation date (expressed in YYYY-MM-DD format or just YYYY if the year is ~~enough~~sufficient for identification purposes). For an Akoma Ntoso XML representation, this value ~~must~~MUST correspond to the content of the element `<FRBRdate>` in the `<FRBRExpression>` section of the metadata~~.~~ (required).
- Number or title or other disambiguating feature of the Work (when appropriate, otherwise optionally the string `nn`). For an Akoma Ntoso XML representation, this value ~~must~~MUST correspond to the content of element `<FRBRnumber>` or `<FRBRname>`, respectively, in the metadata~~.~~ (required when necessary for disambiguation, optional otherwise).

~~All components are separated by forward slashes ("/") so as to exploit relative IRIs in references.~~

- Component and fragment specifications, as specified in sections 4.7 and 4.8 (optional)

For example:

- *[http://www.authority.org]***/akn/dz/debaterecord/2004-12-21**
  Algerian parliamentary debate record, 21st December 2004.

- *[http://www.authority.org]***/akn/sl/act/2004-02-13/2**
  Sierra Leone enacted Legislation. Act number 2 of 2004.

- *[http://www.authority.org]***/akn/ng/bill/2003-05-14/19**
  Namibia Bill number 19 of 2003

- *[http://www.authority.org]***/akn/mg/act/2003-03-12/3**
  Madagascar. Act 3 from 2003

- *[http://www.authority.org]***/akn/ke/act/decree/MinistryForeignAffairs/2005-07-12/3**
  Kenya, Decree n. 3 of 2005 by the Ministry of Foreign Affairs

~~11.0.0 The IRI for WorkComponents~~

~~Although components really only belong to expressions, it often happens that legislation makes Work-level references to components, which thus need to have Work-level IRIs as well. It may happen (and it has happened in the past) that the component (e.g., an attachment) may change name, or position, or even hierarchical placement, from time to time. For instance, suppose we have an original act that refers to table A of schedule 1. Suppose further, that after a little time, schedule 1 is completely abrogated and that table A thus becomes (implicitly) an attachment of the main document.~~ ~~As such, it is important that all references to table A of schedule 1 are considered as references to table A of the main document after that event.~~~~This brings about the necessity to have IRIs for Work Components.~~ ~~These are to be used when referring in a Work-level fashion to components that have official names and positions, but may have a change in name and position with time. One problem is that a Work-level component IRI has no Expression-level part and yet the component part is AFTER the Expression-level part. Therefore, it is necessary to make sure that a Work-level IRI fragment is never mistaken for an Expression-level or a component-level IRI fragment.~~

~~Since:~~

14. ~~The number part of the Work-level IRI (`/nn/`) is required even in unnumbered documents ("`/nn/`" for *not numbered*) and~~

15. ~~The Expression fragment, if present, always has at least the language and the "@" character, and the @ character can only be used for Expression fragments, the absence of a part containing the "@" character indicates a Work-level component reference after the 4th component (the number). For an Akoma Ntoso XML representation, this value must correspond to the content of element `<docTitle>` of the document.~~

- *[http://www.authority.org]*/akn/~~kn/act/2007-01-01/1/schedule1~~
  ~~Kenya, schedule 1 of act 1 from 2007 (WorkComponent)~~**it-45/act/legge/consiglio/2004-05-24/11**
  Regione Emilia-Romagna (it-45) Act n. 11 of 24 May 2014

~~17.0.0 The IRI for Work-level portion queries~~

~~For querying a portion of a document at Work-level, we use a query language composed by the tilde symbol "~" following the fragment name (e.g., art_13). This syntax permits the server to manage the fragment information and so to detect the best manifestation (or all the manifestations) available AND to extract the portion requested:~~

- [http://www.authority.org]/akn/~~sl/act/2004-02-13/2~art_3~~UN/doc/standard/FAO/1981/CODEXSTAN33-1981/
  FAO standard CODEXSTAN33-1981 of 1981.

~~4.6~~4.5 The IRI reference of an Expression

Characterizing the Expression is the specific identification of ~~content with respect to another piece of~~ the features of the content. This includes specifications of the version and the language of the Expression. Therefore, different versions of the same Work, or the same version of the same Work expressed in different languages correspond to different Expressions and will have different IRIs. Expressions are organized in components (the ExpressionComponents), and therefore we need to identify separately the Expression as a whole from the individual IRIs for each ExpressionComponent. All of them are all immediately derived from the baseline, which is the IRI for the Work.

~~4.6.1~~4.5.1 The IRI for the Expression as a Whole

The IRI for the Expression as a whole consists of the following pieces:

- The IRI of the corresponding Work as discussed in section 4.4

- The character "/"

- The human language code in which the Expression is drafted (a three-letter code according to ISO 639-2 alpha-3). For an Akoma Ntoso XML representation, this value ~~must~~MUST correspond to the content of the first element `<FRBRlanguage>` in the metadata section. According with ISO 639-2 alpha-3 "mul" means multilingual document (text with different languages), while "und" means undetermined language

- The "@" character (required)

- Zero or more ~~comma~~semicolon-separated version identifiers as follows:

  - If an approved act, the version date of the Expression in syntax YYYY-MM-DD. For an Akoma Ntoso XML representation, this value ~~must~~MUST correspond to the content of element `<FRBRdate>` in the `<FRBRExpression>` section of the metadata. If appropriate, the date can be integrated with a time using values for the XSD:dataTime datatype: Thh:mm:ss±hh:mm. The difference between the local time and Coordinated Universal Time (UTC) is specified using the sign + or - followed by the difference from UTC represented as hh:mm (note: the minutes part is required). See ISO 8601 Date and Time Formats and XML Schema Part 2: Datatypes (http://www.w3.org/TR/xmlschema-2/~~(http://www.w3.org/TR/xmlschema-2/).~~).

  - If ~~a bill~~any other document, the presentation date is appropriate, or the ~~stage in the approval process that the current draft is the result of.~~

    - ~~If an official~~ version number if one exists, ~~the~~or any other disambiguating string that helps identifying the specific version ~~number preceded by "ver_"....~~ of the document.

For an Akoma Ntoso XML representation, this value ~~must~~MUST correspond to the content of element `<FRBRversionNumber>`

- • Any content-specification date (as opposed to validity dates) (optional).

- • ~~The "!" character (required only if an optional part is added)~~

  - • Any content authoring information to determine the ~~authoritat~~authorit~~liveness~~y of the text content~~.~~ (optional). This is separate and independent of the authoring information relative to the metadata and markup, which are among the features of the Manifestation (optional). For an Akoma Ntoso XML representation, ~~these values must~~this value MUST correspond to the content of ~~elements~~element `<FRBRauthor>` in the `<FRBRExpression>` section of the metadata.

  - • ~~Any content-specification date (as opposed to validity dates) (optional).~~

The absence of the version identifiers signals two different situations depending on the type of document:

- - If the document is not versioned (e.g., the debate record of an assembly) then version identifier need not and cannot be present.

- - If the document is versioned (e.g., an act in force), then the lack of version identifiers refers to the version in force at the moment of the resolution of the IRI (i.e., the "current" version of the act, where "current" refers to the moment in time in which the IRI is dereferenced, rather than the moment in time in which the document containing the IRI was created: *today* for the reader, as opposed to *today* for the author of the references).

A particular Expression is the first version of a Work. This Expression should not be confused with the Work itself (which considers the first Expression in no special way to all other possible expressions), and it is a very specific, although peculiar, Expression. The ~~original~~first version of an Expression is referred to with an IRI with a dangling "@" character (which implies that the actual version date is the first appropriate date for that Work).

- - *[http://www.authority.org]***/akn/dz/debaterecord/2004-12-21/nn/fra**
    Algerian parliamentary debate record, 21st December 2004~~.,~~, French version, with nn optional undefined number.

- - *[http://www.authority.org]***/akn/sl/act/2004-02-13/2/eng**
    Sierra Leone enacted Legislation. Act number 2 of 2004. English version, current version (as accessed today [according to the reader])

- - *[http://www.authority.org]***/akn/sl/act/2004-02-13/2/eng@**
    Sierra Leone enacted Legislation. Act number 2 of 2004. English version, original version

- - *[http://www.authority.org]***/akn/sl/act/2004-02-13/2/eng@2004-07-21**
    Sierra Leone enacted Legislation. Act number 2 of 2004. English version, as amended on July 2004

- - *[http://www.authority.org]***/ /akn/uy/bill/ejecutivo/carpeta/2005-04-04/137-2005/esp@2005-05-02T13:30:00-03:00**~~/bill/~~
    Uruguay bill. Number 137-2005, at 2005-04-04. Spanish version, as amended on May 2nd, 2005, at 10.30 Montevideo time.

- - *[http://www.authority.org]***/akn/ng/bill/2003-05-14/19/eng@first**
    Namibia Bill number 19 of 2003, first ~~stage~~version, English version

- - *[http://www.authority.org]***/akn/mg/act/2003-03-12/3/mul**
    Madagascar. Act 3 from 2003, current version (as accessed today [according to the reader]) in multilingual text (in this case, French and Malagasy~~. This means having a document with multilingual text.~~).

- - *[http://www.auth.org]***/akn/sl/act/2004-02-13/2/eng@2004-07-21**~~!official/2004-07-25/official~~**publisher**
    Sierra Leone enacted Legislation. Act number 2 of 2004. English version, as amended on July 2004. Official version ~~generated on 25 July 2004~~. Emitted by publisher body.

- *[http://www.authority.org]***/akn///**eu/bill/~~directive/cn~~DIR/CONSIL/2013/COM(2013)344/eng@~~ver second~~final_2
Proposal for an European directive of the Council – English variant, ~~second~~ version~~. .~~ "final/2".

11.0.0 The IRIs for ExpressionComponents

Some expressions have many components; some are only composed of a main document. In order to explicitly refer to individual components, it is therefore necessary to introduce a naming convention that identifies individual components, and still allows an easy connection between the component and the Expression it belongs to.

There are therefore two subcases following explained.

The Expression is Only Composed of One Component

In this case, the IRI for the Expression as a whole and for its main component are identical plus the name "main".

The Expression is Composed of Many Components

In this case, the IRI for each ExpressionComponent consists of the following pieces:

- The IRI of the corresponding Expression as a whole
- The character "/"
- Either:
- 1. A unique name for the attachment
- 2. The name "main" which is reserved for the main document. It we have different main they are numbered sequentially: main1, main2, etc.

Some examples:

- *[http://www.authority.org]***/akn/**dz/minutes/2004-12-21/fra/main
Algerian parliamentary debate record, 21st December 2004., French version, main document
- *[http://www.authority.org]***/akn/sl**~~uy~~**/act/**2004-02-13/2/eng/main
Main body of the Sierra Leone enacted Legislation. **2008-08-11/18331/esp@2009-12-12;2010-01-01~art_3__para_5__point_c**
Act ~~number 2 of 2004. English version, current version (as accessed today)~~
- *[http://www.authority.org]***/akn/sl/act/2004-02-13/2/eng@2004-07-21/main/schedule1**
Attachment "schedule01" of Sierra Leone enacted Legislation. Act number 2 of 2004. English version, as amended on July 2004
- *[http://www.authority.org]***/akn/ng/bill/2003-05-14/19/eng@first/main/schedule3**
Third attachment of Namibia Bill number 19 of 2003, first stage, English version.

28.0.0 Hierarchies of Components~~of Uruguay n. 18331~~ in ExpressionComponents

A frequent situation occurs when an attachment has itself further attachments. This creates a complex hierarchical situation in which the component should be considered, in a way, as an Expression by itself, whose components need to be listed and properly differentiated. The process can be further iterated whenever not only an attachment has further attachments, but its attachments also have further attachments and so on. The situation must also foresee the situation in which attachments at different levels of the hierarchy end up having the same name (e.g., table A in schedule 1 and table A in schedule 2).

In such situations, each ExpressionComponent must be considered as an Expression by itself. Recursively, the IRI of attachments are as follows:

- If the attachment does not have further attachments, its IRI is provided as detailed in the previous section, without further addenda.

- If the attachment has further attachments, the IRI, as detailed in the previous section, refers to the whole attachment, including its own attachments.
- To refer to the main document of an attachment that has further attachments, a further "/main" part should be added.
- To refer to any further attachment of an attachment, a further "/" followed by a unique name for the attachment must be added to the attachment itself.

Some examples:

- *[http://www.authority.org]*/akn/sl/act/2004-02-13/2/eng@2004-07-21/main/schedule1
  Whole attachment "schedule01" of the Sierra Leone enacted Legislation. Act number 2 of 2004. English version, English version, as amended on July 2004.

- *[http://www.authority.org]*/akn/sl/act/2004-02-13/2/eng@2004-07-21/main/schedule1/main
  Main document of the attachment "schedule01" of Sierra Leone enacted Legislation. the version of 2009, Dec 12, Act number 2 of 2004. English version, as amended on July 2004.

- *[http://www.authority.org]*/akn/sl/act/2004-02-13/2/eng@2004-07-21/main/schedule1/tableA
  Attachment "Table A" of the attachment "schedule01" of Sierra Leone enacted Legislation. Act number 2 of 2004. English version, as amended on July 2004.

- *[http://www.authority.org]*/akn/sl/act/2004-02-13/2/eng@2004-07-21/main/schedule1/attachment1/main
  Main document of the attachment "attachment01" of the attachment "schedule01" of Sierra Leone enacted Legislation. Act number 2 of 2004. English version, amended on July 2004.

- *[http://www.authority.org]*/akn/eu/debate/2004-02-13/2/mul@/main
  Main document of the European parliament debate report with multiple languages embedded.

- *[http://www.authority.org]*/akn/eu/bill/directive/cnl/2013/eng@ver_second/annex_1
  Annex 1 of the proposal for a Council directive, in the second versiona retroactive modification happened in 2010.

4.6.24.5.2 The IRIs for Virtual Expressions

In some situations, the information such as the actual enter-in-force date of the Expression or the language of an Expression is not known in advance, andor it is necessary to create references or mentions of documents whose IRI isfeatures are now known completely (possibly, because their exact delivery date is not known yet). These are called *virtual expressions* (i.e., references to expressions that probably do not exist yet or ever, but can be unambiguously deduced once all relevant information is made available.)

There are at least three cases where such a situation may arise:

4. The information is not known by the author of the Expressionreference (e.g., the legislator), in which case the act of actually retrieving the correct information is in itself an act of interpretation.

5. The information is not known by the editor of the Expressiontext containing the reference (e.g., the publisher of the XML version of the document), in which case the information can theoretically be available, but is too much of a burden for the publisher to retrieve it.

6. The information is not known by the query system when evaluating the reference.

In these cases, the syntax for the IRI of the virtual Expression uses a similar syntax to the specification of the actual Expression, but the character ":" is used before each unknown value and instead of the "@" at the end of the specification of the Work-level IRI. For instance, if we need to reference the Expression of an act in force on date "1/1/2007", we will probably need to refer to some Expression whose enter in force date was in a previous date to 1/1/2007.

- *[http://www.authority.org]*/akn/sl/act/2004-02-13/2/eng:2004-07-212007-01-01
  Sierra Leone enacted Legislation. Act number 2 of 2004. English version, as amended on the closest date before July 21, 2004January 1st, 2007

- *[http://www.authority.org]***/akn/eu/act/2004-11-13/87/und:2015-01-10**
  European Directive number 2004/87/EC of 2004. All the language versions, as amended on the closest date before January 10, 2015

Similarly, if we need to refer dynamically to the expressions in German of a specific act, we need to make a virtual reference whose date is left unspecified, and the language is forced to be German, as follows (`deu` is SO 639-2 alpha-3 code for German).

- *[http://www.authority.org]***/akn/ch/act/2009-05-09/432/:/deu:**
  Swiss enacted Legislation. Act number 432 of 2009. Dynamic reference to any of the German versions.

11.0.0 The IRI for Expression-level portion queries

For querying a portion of a document at Expression-level, we use a query language composed by the tilde symbol "~" following the fragment name (e.g., art_13) after the Expression fragment. This syntax permits the server to manage the fragment information and so to detect the best manifestation (or all the manifestations) available AND to extract the portion requested:

- *[http://www.authority.org]***/akn/~~slit~~it/act/~~2004-02-13/2~~2005-03-07/82/eng~~@2004-07-21~art_3~~:2010-01-01->2015-12-31**
  Italian enacted Legislation. Act number 82 of 2005-03-07. Dynamic reference to any of the Italian versions valid within the interval January 1st 2010 to December, 31st 2015.

4.7~~4.6~~ The IRI reference of a Manifestation

Characterizing the Manifestation is the specific process that generate~~ds~~ an electronic document in ~~some~~a specific format~~(s)~~. This includes specifications of the data format~~(s)~~ used. Therefore, different ~~m~~Manifestations of the same Expression generated ~~using~~ different data formats ~~correspond to different manifestations and~~ will have different ~~IRIs~~IRI references.

~~Manifestations are organized in components (the ManifestationComponents), and therefore we must identify separately the Manifestation as a whole and the individual IRIs for each ManifestationComponent.~~ All of them are all immediately derived from the baseline, which is the IRI for the Expression.

4.7.1 The IRI for the Manifestation as a Whole

The IRI for the Manifestation as a whole consists of the following pieces:

- The IRI of the corresponding Expression as a whole
- ~~The IRI of the corresponding Expression as a whole~~
- ~~The character "!" (only required if any of the optional parts is added)~~

  - The markup authoring information (useful to determine the authoritativeness of the markup and metadata) (optional). For an Akoma Ntoso XML representation, this value ~~must~~MUST correspond to the content of element `<FRBRauthor>` in the `<FRBRManifestation>` section of the metadata.

  - Any relevant markup-specific date (optional). For an Akoma Ntoso XML representation, this value ~~must~~MUST correspond to the content of element `<FRBRdate>` in the `<FRBRManifestation>` section of the metadata.

  - Any additional markup-related annotation (e.g., the existence of multiple versions or of annotations.) (optional)

  - The character "." (required)

  - A unique three or four letter ~~acronym of~~extension signifying the data format in which the Manifestation is drafted~~. The acronym~~ (required). For instance, such extension can be "pdf" for PDF, "doc" or "docx" for MS Word, "htm" or "html" for HTML "xml" for ~~the~~an XML Manifestation, or "akn" for the package of all documents including XML version~~s~~ of the main document(s) according to the Akoma Ntoso ~~rules (required).~~ vocabulary. For an Akoma

Ntoso XML representation, this value ~~must~~MUST correspond to the content of element `<FRBRformat>` in the `<FRBRManifestation>` section of the metadata.

Some examples:

- *[http://www.authority.org]***/akn/dz/debaterecord/2004-12-21/fra**~~/main.~~**@.doc**
  Word version of the Algerian parliamentary debate record, 21st December 2004~~.,~~ Original French version

- *[http://www.authority.org]***/akn/sl/act/2004-02-13/2/eng**~~/main~~**.pdf**
  PDF version of the Sierra Leone ~~enacted Legislation. Act~~act number 2 of 2004~~.,~~ English version, current version (as accessed today)

- *[http://www.authority.org]***/akn/sl/act/2004-02-13/2/eng@2004-07-21**~~/main~~**.akn**
  Package of all documents ~~including~~in Akoma Ntoso XML ~~versions~~ of the Sierra Leone ~~enacted Legislation. Act~~act number 2 of 2004. English version, as amended in July 7th 2004~~.~~

- *[http://www.authority.org]***/akn/sl/act/2004-02-13/2/eng@2004-07-21**~~!~~**/CIRSFID/2011-07-15**~~/main~~**.akn**
  Package of all documents including XML versions of the Sierra Leone enacted Legislation. Act number 2 of 2004. English version, as amended in July 2004. Rendered in Akoma Ntoso by CIRSFID on 15 July 2011.

~~4.8.0 The IRI for Manifestation-level portion naming~~

~~The syntax for naming a portion at the Manifestation level is the following:~~

~~In the case of the US Code, the chapter 3 portion of Title 9 is specified as:~~

~~/akn/us/usc/title_9/eng@2013-07-26-chp_3/main.akn~~

~~4.13.0 The IRIs for ManifestationComponents~~

4.7 ~~Each ManifestationComponent is an independent electronic structure (e.g., a file) in a single data format. Every type of Manifestation has, of course, a different data structure and file structure. Therefore the actual format of the IRIs of the~~Specifying components ~~of the Manifestation depend on the data format and cannot be formalized in general. In~~and portions

Referring to within-documents destination is possible by adding to a whole-document reference an additional item pointing to the specific destination. Within this specification we distinguish between *components* and *portions*.

Some documents are made of many components; some are only composed of a main document. Within each component, structured portions can be localized and referred to by means of identifying information. Legal citations often point to a specific portions of the abstract document, as in "Article 3 of Directive 2003/87/EC".

In order to explicitly refer to individual components and portions, it is therefore ~~provide a grammar but not an exhaustive list of formats~~necessary to introduce a naming convention that identifies them, and allows an easy connection between the component and the document it belongs to.

Components and portions are a way to organize content. As such, they are naturally Expression-level concepts, which is where content is actually expressed and structured. Yet, it is sometimes appropriate to refer to components and portions in Work-level references, and therefore a syntax for these situations is provided. For instance, consider an original act that refers to table A of schedule 1. After a little time, schedule 1 is completely abrogated and table A thus becomes (implicitly) an attachment of the main document. As such, it is important that all references to table A of schedule 1 are considered as

references to table A of the main document after that event. This brings about the necessity to have IRIs for Work Components. depends onThese are to be used when referring in a Work-level style to components that have official names and positions, but may change in name and position with time.

Components and portions are specified at the end of the IRI for Work-level and Expression-level references, and just before the data format for Manifestation-level references. They are introduced by a special character to separate the data format chosen forparts of the Manifestation. The IRI for each ManifestationComponentreference identifying the document as a whole from the parts identifying the component and/or portion.

4.7.1 Specifying components in IRI references

A Work-level IRI reference including a component specification consists of the following pieces:

- The IRI of the corresponding Work as a whole
- The character "/"
- The character "!" and
- Either:
  – The name "main" if the component is the only component or the main component in a hierarchy
  – A unique name for the attachment, optionally followed by the number associated to this component in the document. If this number is provided, it is separated from the name with "_".

For instance:

  – *[http://www.authority.org]***/akn/kn/act/2007-01-01/1/!main**
    Kenya, main document of act 1, 2007
  – *[http://www.authority.org]***/akn/eu/act/DIR/2014-03-01/16/!annex_1**
    annex 1 of the European directive 16 of 1 March 2014

An Expression-level IRI reference including a component specification consists of the following pieces:

- The IRI of the corresponding Expression as a whole
- The character "!" (only required if any of the optional parts is added)"/"
51. The markup authoring information to determine the authoritativeness of the markup and metadata (optional)
52. Any relevant markup-specific date (optional)
53. Any additional markup-related annotation (e.g., the existence of multiple versions or of annotations) (optional)
- The character "/""!" and
55. Some unique identification of the ManifestationComponent with respect either to the Manifestation as a whole or to the ExpressionComponent the component is the Manifestation of.
56. The character "."
- Either:
  – The name "main" if the component is the only component or the main component in a hierarchy
  – A unique name for the attachment, optionally followed by the number associated to this component in the document. If this number is provided, it is separated from the name with "_".

For instance:

  – *[http://www.authority.org]***/akn/dz/minutes/2004-12-21/nn/fra/!main**
  – Algerian parliamentary debate record, 21st December 2004, French version, main document, current version

- *[http://www.authority.org]***/akn/sl/act/2004-02-13/2/eng@/!main**
  Main body of the Sierra Leone enacted Legislation. Act number 2 of 2004. English version, original version
- *[http://www.authority.org]***/akn/sl/act/2004-02-13/2/eng@2004-07-21/!schedule_1**
  Attachment "schedule 1" of Sierra Leone enacted Legislation. Act number 2 of 2004. English version, as amended on July extension 21st, 2004

In order to allow the specification of the data format of a Manifestation to be the last item of the IRI as usual in web addresses, Manifestation-level IRI references are built differently than Work-level and Expression-level IRI references.

In fact, a Manifestation-level IRI reference including a component specification consists of the following pieces:

- The IRI of the corresponding Manifestation as a whole, minus the "." and the specification of the format,
- The character "/"
- The character "!" and
- Either:
  o The name "main" if the component is the only component or the main component in a hierarchy
  o A unique name for the attachment, optionally followed by the number associated to this component in the document. If this number is provided, it is separated from the name with "_".
- The character "." (required)
- A unique three or four letter extension signifying the data format in which the Manifestation is drafted. The acronym (required). For instance, such extension can be "pdf" for PDF, "doc" or "docx" for MS Word, "htm" or "html" for HTML "xml" for an XML Manifestation, or "akn" for the package of all documents, "tif" for image formats, etc including XML versions of the main document(s) according to the Akoma Ntoso vocabulary. For an Akoma Ntoso XML representation, this value MUST correspond to the content of element `<FRBRformat>` in the `<FRBRManifestation>` section of the metadata.

For instance:

- *[http://www.authority.org]***/akn/it/bill/2004-02-13/C245/ita@2/official/2004-02-15/publisher/!annex_1.pdf**
  Italian bill number C245 as of 2004-02-13, second version, official expression, manifestation released on 2004-02-15 by publisher in pdf format.

4.7.2 Hierarchies of components in component specifications

A frequent situation occurs when an attachment has itself further attachments. This creates a complex hierarchical situation in which the component should be considered, in a way, as a document by itself, composed of components which need to be listed and properly differentiated. The process is further iterated whenever its attachments also have further attachments and so on. The situation must also foresee the situation in which attachments at different levels of the hierarchy end up having the same name (e.g., table A in schedule 1 and table A in schedule 2).

In such situations, each component must be considered as a document by itself. Recursively, the specification of components in IRI references are as follows:

- If the component does not have attachments, its IRI reference is determined as detailed in the previous sections, without further additions.
- If the component has attachments, the IRI reference detailed in the previous sections is meant to refer to the component as a whole, including its attachments.
- To refer to the main document of a component that has further attachments, an additional "/main" part is specified.

- To refer to any further attachment of component, a further "/" followed by a unique name for the attachment is added.

Some examples:

In the next section we will examine the format of the package and the relevant IRIs for a specific Manifestation of Akoma Ntoso documents, the XML format.

- The IRIs for the *[http://www.authority.org]***/akn/sl/act/2004-02-13/2/eng@2004-07-21/!main/schedule_1 and equivalently** *[http://www.authority.org]***/akn/sl/act/2004-02-13/2/eng@2004-07-21/!schedule_1** Attachment "schedule01" of the Sierra Leone Act number 2 of 2004. English version, as amended on July 21st, 2004.

- *[http://www.authority.org]***/akn/sl/act/2004-02-13/2/eng@2004-07-21/!main/schedule_1/main and equivalently** *[http://www.authority.org]***/akn/sl/act/2004-02-13/2/eng@2004-07-21/!schedule_1/main** Main component of the attachment "schedule01" of Sierra Leone Act number 2 of 2004, English version, as amended on July 21st, 2004.

- *[http://www.authority.org]***/akn/sl/act/2004-02-13/2/eng@2004-07-21/!main/schedule_1/table_A and equivalently** *[http://www.authority.org]***/akn/sl/act/2004-02-13/2/eng@2004-07-21/!schedule_1/table_A** Attachment "Table A" of attachment "schedule01" of Sierra Leone act number 2 of 2004, English version, as amended on July 21st, 2004.

- *[http://www.authority.org]***/akn/eu/debate/2004-02-13/2/mul@/!main** Main document of the European parliament debate report number 2 of 2004, containing multiple languages, original version

- *[http://www.authority.org]***/akn//eu/bill/DIR/consil/2013/COM(2013)366/eng@second/!annex_1** Annex 1 of the proposal for a Council directive, second version.

4.13.14.7.3 *Describing* Components in the Akoma Ntoso Package Manifestation

The Akoma Ntoso package Manifestation is a very specific Manifestation using a number of data formats (mainly XML but could include other multimedia formats could be present as needed) with a very specific organization of parts and components. Since it makes explicit choices in terms of data formats and reciprocal references, it is important to provide clear and non-ambiguous rules as to the internal naming mechanism and its overall structure. An Akoma Ntoso package Mmanifestation is a package composed of one or more files organized in a flat fashion. The transportable format is a ZIP file whose extension is ".akn". Other formats are possible and acceptable as long as they adhere to these rules.

The following are alternative options for the Akoma Ntoso package:

7. If the document is just composed of text and does not refer to any multimedia fragment of any form, then the ZIP package contains a single document called "main.xml".

8. If the document is composed of many ManifestationComponentsManifestation-level components but does not refer to any multimedia fragment of any form, then the zip package is composed of many XML files, **one for each ExpressionComponent Expression-level component**. Each ManifestationComponentManifestation-level component is then called as its corresponding ExpressionComponentExpression-level component, plus the ".xml" extension. The name "main" is reserved for the main component. Numbers are never used except when they are already part of the ExpressionComponent'sExpression-level component's name.

9. If the document contains multimedia fragments of any kind, then each individual fragment doesthese fragments do not have a corresponding ExpressionComponentExpression-level component, but isare just a ManifestationComponent referred toManifestation-level components specified in the"src" attributes, for instance in `<img>` or `<object>` element.elements. All multimedia components mustMUST be stored within an inner structure (e.g., a folder) called "media". Multimedia components can be called freely, but mustMUST use the appropriate

extension to refer to their content type. Thus a logo can be called "logo.tif" or any other name, as long as the extension is correctly specifying the content type.

Reciprocal references to ~~ManifestationComponents~~Manifestation-level components are necessary within a specific Manifestation. For instance, the Manifestation of the main document refers to the manifestations of its attachments via the `<attachment>` elements, and the schedule showing an image refers to the file of the image via the `<img>` element. In these cases, all references MUST be relative to the package (i.e., the Manifestation as a whole):

- attachment1.xml
  Manifestation of the first attachment of the current document

- schedule3.xml
  Manifestation of the third attachment of the current document

- media/logo.tif
  Manifestation of an image within the current document

4.7.4 ~~References to ManifestationComponents~~Specifying portions

A portion is a specific document type in Akoma Ntoso XML, that contains a only a part of a whole document. Portion documents are ~~rarely, if ever, needed outside of~~ used when the ~~Manifestation themselves. But if needed, they will refer~~full document is large, and only a smaller part is necessary, such as a few sections or articles.

Requesting a portion is therefore only appropriate for Akoma Ntoso XML Manifestations, and is not guaranteed to be supported by all servers. A request for a portion therefore MUST return a manifestations containing at least the requested part, but possibly containing much more than the requested part, up to ~~the file as follows:~~and including the whole document the part belongs to.

In an Akoma Ntoso IRI reference, the specification for the portion of a document is introduced by the special character "~".

Therefore, a Work-level IRI reference (or an Expression-level IRI reference) including a portion specification consists of the following pieces:

- The IRI of the corresponding Work (or, respectively, of the corresponding Expression) as a whole,

~~62.~~ The character "/"

- ~~The relative reference to the~~ (required ~~ManifestationComponent as specified above.~~),

- A specification of the component according to section 4.7.1 (optional),

- The character "~" (required),

- Either:

  – The eId of the element containing the subtree being requested, or

  – A pair of eIds of the first and last elements of the sequence of parts being requested, separated by a dash "->".

In order to allow the specification of the data format of a Manifestation to be the last item of the IRI as usual in web addresses, Manifestation-level IRI references are built differently than Work-level and Expression-level IRI references.

In fact, a Manifestation-level IRI reference including a portion specification consists of the following pieces:

- The IRI of the corresponding Manifestation as a whole, minus the "." and the specification of the format,

- The character "/" (required),

- A specification of the component according to section 4.7.1 (optional),

- The character "~" (required),

- Either:

- The eId of the element containing the subtree being requested, or
- A pair of eIds of the first and last elements of the sequence of parts being requested, separated by a dash "->".

- The character "." (required)

- A unique three or four letter extension signifying the data format in which the Manifestation is drafted (required). For instance, such extension can be "pdf" for PDF, "doc" or "docx" for MS Word, "htm" or "html" for HTML "xml" for an XML Manifestation, or "akn" for the package of all documents including XML versions of the main document(s) according to the Akoma Ntoso vocabulary. For an Akoma Ntoso XML representation, this value MUST correspond to the content of element `<FRBRformat>` in the `<FRBRManifestation>` section of the metadata.

For instance:

- *[http://www.authority.org]*/akn/eu/act/2003-11-13/87/!main~art_3
  The portion containing article 3 of the Work of the European Directive 2003/87/EC

- *[http://www.authority.org]*/akn/eu/act/2003-11-13/87/eng@/!main~art_3
  The portion containing article 3 of the Expression of the European Directive 2003/87/EC in English

- *[http://www.authority.org]*/akn/eu/act/2003-11-13/87/eng@2015-01-20/!main~art_3->art_5
  The portion containing article 3 to article 5 of the Expression of the European Directive 2003/87/EC in English at the version dated 2015-01-20

- *[http://www.authority.org]*/akn/eu/act/2003-11-13/87/eng@2015-01-20/!main/schedule_1~art_3.xml
  The portion containing article 3 of schedule 1 of the XML Manifestation of the European Directive 2003/87/EC in English at the version dated 2015-01-20.

When the component is the main component of the document, the "!main" part can be omitted. So the following IRIs are equivalent to the previous examples:

- [http://www.authority.org]/akn/eu/act/2003-11-13/87/~art_3

- [http://www.authority.org]/akn/eu/act/2003-11-13/87/eng@/~art_3

- [http://www.authority.org]/akn/eu/act/2003-11-13/87/eng@2015-01-20/~art_3->art_5

- [http://www.authority.org]/akn/eu/act/2003-11-13/87/eng@2015-01-20/!schedule_1~art_3.xml

4.144.8 The IRI of an Item

Akoma Ntoso makes no assumption ~~on~~about the physical storage mechanism employed to record actual manifestations. As such, there is NO rule for IRIs of the items, which are free to assume any form whatsoever and correspond to whatever storage mechanism has been employed locally.

On the other hand, the actual URL for the Item must be provided to a resolution mechanism in order for the hyper-textual feature of the Akoma Ntoso publication systems to work correctly and automatically.

4.9 Local IRI references

A global IRI reference is a relative IRI reference where all parts are present except for protocol and authority (i.e., domain name). A global IRI reference *always* starts with a slash, to indicate that *all* other parts are explicitly specified. A local IRI reference, on the other hand, has one or more parts missing (necessarily from left to right), and the corresponding global (and, subsequently, absolute) IRI reference is determined by adding the corresponding parts taken from the base document, as described in RFC 3986.

In the Akoma Ntoso Naming Convention, a reference to a different document is always global. Local IRI references are used for references to components, portions or fragments, and implicitly make use of the identification parts of the base document for the correct resolution of the IRI reference.

There are three types of local IRI references in this specification:

4.9.1 Fragment references

A fragment reference is composed of just the fragment part of the Akoma Ntoso IRI reference, according to the specification of RFC 3986. It is composed of the following parts:

- The character "#" (required)
- The eId of the element requested

For instance:

- #art_5

As per RFC 3986, dereferencing this IRI reference does not imply making a request to the origin server. This request scrolls the current document so as to bring into view the part of the document whose eId has been requested.

The implication of using this type of local references is that the requested part of the document is already dereferenced and available at the client. On the other hand, any Akoma Ntoso document can contain this type of references.

4.9.2 Local component references

A local component reference is composed of just the component part of the Akoma Ntoso IRI reference, according to the syntax in section 4.7.1. It is composed of the following parts:

- The character "!" and
- Either:
    - The name "main" if the component is the only component or the main component in a hierarchy
    - A unique name for the attachment, optionally followed by the number associated to this component in the document. If this number is provided, it is separated from the name with "_".

For instance:

- !schedule_3
- !main

Dereferencing this IRI reference implies making a new request to the origin server. The server may choose to return the requested component either as an autonomous document or as a component of a document collection document type.

The implication of using this type of local references is that the base IRI MUST already contain either the component or the portion part. It is unspecified and usually incorrect to use a local component reference in a document whose base IRI does NOT include a component or portion part.

4.9.3 Local portion references

A local portion reference is composed of just the portion part of the Akoma Ntoso IRI reference, according to the syntax in section 4.7.4. It is composed of the following parts:

- The character "~" (required),
- Either:
    - The eId of the element containing the subtree being requested, or
    - A pair of eIds of the first and last elements of the sequence of parts being requested, separated by a dash "->".

For instance:

- ~art_5
- ~sec_3->sec_5

Dereferencing this IRI reference implies making a new request to the origin server. The server MUST either return the requested part as a portion document type containing at least the part being requested,

OR the full document or component containing the part being requested, even when they correspond to the current document.

The implication of using this type of local references is that the base IRI MUST already contain either the component or the portion part. It is unspecified and usually incorrect to use a local component reference in a document whose base IRI does NOT include a component or portion part.

4.9.4 Mixed local references

It s always possible to combine the three different types of local references into a complex local IRI reference. For instance:

- **!main/schedule_1~art_3->art_5#art_4**
  a portion document containing at least the sequence from article 3 to article 5 of the component called schedule 1 of the current document. When dereferenced, scroll to article 4.

4.154.10 The IRI of Non-Document Entities

The object of all discourses within the Akoma Ntoso framework can be described as a set of abstract classes and their instances and of the relationship among them. Cumulatively, definition of classes, relationships and instances are called an *ontology*.

The four most important classes of the Akoma Ntoso ontology (Work, Expression, Manifestation, and Item) are surely connected to documents, but many more exist, even if they are not connected directly to physical documents. The purpose of this section is to provide syntax for non-document entities (i.e., instances of non-document classes such as people, organizations, or concepts.) Furthermore, the syntax described here can also be used for document entities as an equivalent syntax to the one specified in the previous sections.

Akoma Ntoso entities are always associated to a class, providing a structure of properties and relationships to other instances of the same and other classes. Classes in the Akoma Ntoso ontology are organized in a complex maze of sub/superclasses. These are useful to give shape and meaning to a domain, and to provide structure to the overall set of instances of a base class. It is important to notice that sub/superclasses do not form necessarily a tree, but can form a more complex structure, namely a *directed graph*.

For instance, the class of Kenyan judges can be considered a sub class of both Kenyan persons and of persons whose job description is judge. That is, there is a (implicit or explicit) subclass of Judges and (implicit or explicit) subclasses of Kenyans, both of which are, in turn, subclasses of Person, and Kenyan Judges is a subclass of both. In fact, we immediately derive the principle that every different value in every different property or relationship implicitly generates a class, that turns into an explicit class only because of our whim or need. For instance, the class of all persons named "Joe" exists implicitly, identifies all persons whose first name is "Joe", and, if so desired, can be made explicit through the definition of a subclass of Person.

While this is very useful for determining relationships between entities, it affects the mechanism to associate IRIs to such entities. In particular, being that there is no single hierarchy of classes, it is not appropriate to propose a single path of specifications from the super class to the final class. As such, ideally `/person/judge/ken/JoeSmith` must point to the same individual as `/person/ken/judge/JoeSmith`.

In order to maintain meaningfulness, permanence and invariance (which are the main requirements for our naming convention, as specified in the introduction of this document) we need to find a reliable naming mechanism for clearly identifying entities that does not depend on the sub/superclass organization except when strictly necessary.

In particular, we define the concept of *Top Level Classes* (TLC) that are guaranteed to be a partition of the overall domain of the Akoma Ntoso standard. TLC include Work, Expression, Manifestation, Item, Person, Organization, Concept, Object, Event, Process, Role, Term and Location. The list of TLC may, in the future, include more, as long as they keep on generating a partition (i.e., that they are disjoint and

cumulatively describe all possible instance of the Akoma Ntoso domain). Members of the TLC classes can be subclassed at will and with no theoretical constraints.

Given the high number of foreseeable subclasses of the TLC, and the pointlessness of determining a fixed hierarchy in such number, the naming of entities should not depend on the presence or absence of a given class except for TLC. This means that it is necessary that each instance of each TLC is provided with an ID string that is guaranteed to be unique within the TLC. The syntax of this ID is dependent of the TLC class, and the syntax for each of the existing TLC is provided in the next section.

Therefore, the IRI for non-document entities consists of the following pieces:

- The base URL of a naming authority with IRI-resolving capabilities
- A detail fragment organizing in a hierarchical fashion the additional data:
  – The string "/ontology"
  – The official name of the appropriate TLC
  – Any number (including none) of slash-separated subclasses of the TLC, as long as they all refer to correct properties of the corresponding instance
  – The ID of the instance, guaranteed to be unique within the TLC.

All components are separated by forward slashes ("/") so as to exploit relative IRIs in references.

  – *[http://www.authority.org]*/akn/ontology/person/kn.joe.smith.1964-12-22
    Joe Smith
  – *[http://www.authority.org]*/akn/ontology/person/kn/kn.joe.smith.1964-12-22
    Joe Smith (implying that he is a Kenyan)
  – *[http://www.authority.org]*/akn/ontology/person/kn/judge/kn.joe.smith.1964-12-22
    Joe Smith (implying that he is a Kenyan who is a judge)
  – *[http://www.authority.org]*/akn/ontology/person/judge/kn/kn.joe.smith.1964-12-22
    Joe Smith (implying that he is a judge who is a Kenyan)
  – *[http://www.authority.org]*/akn/ontology/person/kenyanjudge/kn.joe.smith.1964-12-22
    Joe Smith (implying that he is a Kenyan judge)

Please note that the classes Work, Expression, Manifestation, and Item belong to the ontology as much as the other classes. As such, each Work, Expression, and Manifestation can also be indicated with an ontology-based IRI that refers to exactly the same entity. Therefore, the following IRIs are equivalent pair-wise, and refer to the same entities:

  – *[http://www.authority.org]*/akn/sl/act/2004-02-13/2
    [http://www.authority.org]/akn/ontology/work/sl.act.2004-02-13.2
  – *[http://www.authority.org]*/akn/sl/act/2004-02-13/2/eng@2004-07-21
    [http://www.authority.org]/akn/ontology/expression/sl.act.2004-02-13.2.eng@2004-07-21
  – *[http://www.authority.org]*/akn/sl/act/2004-02-13/2/eng@2004-07-21/main/schedule1!schedule_1
    [http://www.authority.org]/akn/ontology/expression.component/sl.act.2004-02-13.2.eng@2004-07-21.main.schedule1!schedule_1
  – *[http://www.authority.org]*/akn/sl/act/2004-02-13/2/eng@2004-07-21/main.akn
    [http://www.authority.org]/akn/ontology/manifestation/sl.act.2004-02-13.2.eng@2004-07-21.main.akn

4.15.14.10.1 The Identifiers for Top Level Classes

As mentioned in the previous section, the hierarchy of path elements is of no use for identifying instances of each TLC, given the fact that there can be no unique hierarchy of subclasses in the Akoma Ntoso ontology. Thus, each instance of the ontology needs to be provided with an ID guaranteed to be unique

within the TLC it belongs to. The syntax of the ID depends on the actual TLC, and is briefly explained in the following schema.

TLCPerson

A dot-separated string composed of the country of citizenship, the first name, the family name, the birth date in `yyyy-mm-dd` format, and an optional arbitrary string if ambiguity exists (e.g., if two individuals with the same name and the same birth date exist in the same country).

- `kn.joe.smith.1964-12-22`
  Mr. Joe Smith, the only Kenyan citizen with that name born on December 22nd, 1964

TLCOrganization

A dot-separated string composed of the country of registration (or the string "`int`" if international, or the string "`unreg`" if not registered anywhere), a recognizable form of the organization name and an optional arbitrary string if ambiguity exists (e.g., if two organizations with the same name exist in the same country).

- `kn.parliament`
  the Kenyan Parliament

TLCConcept

Concepts differ from terms as they are refer to a specific word or collection of words embodying some concept, rather than to the concept embodied by different words. Therefore, for instance, pope and pontiff are different terms for the same concept, while date is a single term referring to two different concepts (a calendar date as opposed to a type of fruit). Concepts must refer to a specific reference resource that can be used to disambiguate the object being referred. This must be a thesaurus, an encyclopedia, or a commonly available dictionary. A unique form of the terms specifying the concept joined with dots preceded by an unambiguous name for the resource being used. No country specifications are necessary for concepts.

- `wikipedia.Presidential.election`
  the concept of Presidential Election as defined in Wikipedia

TLCObject

Objects must refer to a specific reference resource that can be used to disambiguate the object being referred to. This must be a thesaurus, an encyclopedia, or a commonly available dictionary. A unique form of the terms specifying the concept joined with dots preceded by an unambiguous name for the resource being used. No country specifications are necessary for objects.

- `wikipedia.weapon`
  a weapon (as a physical object) as defined in Wikipedia

TLCEvent

Events must refer to a specific reference resource that can be used to disambiguate the object being referred to. This must be a thesaurus, an encyclopedia, or a commonly available dictionary. A unique form of the terms specifying the concept joined with dots preceded by an unambiguous name for the resource being used. No country specifications are necessary for events.

- `wikipedia.world.war.ii`
  The second World War as defined in Wikipedia

TLCLocation

Places must refer to a specific reference resource that can be used to disambiguate the object being referred to. This must be a thesaurus, an encyclopedia, or a commonly available dictionary. A unique form of the terms specifying the concept joined with dots preceded by an unambiguous name for the resource being used. No country specifications are necessary for places.

- `wikipedia.rome`
  The city of Rome as defined in Wikipedia

TLCProcess

Processes must refer to a specific reference resource that can be used to disambiguate the object being referred to. This must be a thesaurus, an encyclopedia, or a commonly available dictionary. A unique form of the terms specifying the concept joined with dots preceded by an unambiguous name for the resource being used. Country specifications are necessary for processes since processes with the same name may exist with different steps across different countries.

- `wikipedia.kn.promulgation`
  The promulgation as defined in Wikipedia and as carried out in Kenya.

TLCRole

Roles must refer to a specific reference resource that can be used to disambiguate the object being referred to. This must be a thesaurus, an encyclopedia, or a commonly available dictionary. A unique form of the terms specifying the concept joined with dots preceded by an unambiguous name for the resource being used. Country specifications are necessary for roles since roles with the same name may exist with different characteristics across different countries.

- `wikipedia.kn.speaker`
  The role of the speaker of the house as defined in Wikipedia and as conceived in Kenya.

TLCTerm

Terms differ from concepts as they are referring to a specific word or collection of words embodying some concept, rather than to the concept embodied by different words. Therefore, for instance, pope and pontiff are different terms for the same concept, while date is a single terms referring to two different concepts (a calendar date as opposed to a type of fruit). Terms must refer to a specific reference resource that can be used to disambiguate the object being referred to. This must be a thesaurus, an encyclopedia, or a commonly available dictionary. A unique form of the terms specifying the concept joined with dots preceded by an unambiguous name for the resource being used. No country specifications are necessary for places but a language reference is necessary for the correct attribution.

- `wikipedia.eng.speaker`
  The role of the speaker of the house as defined in Wikipedia and expressed in English.

TLCReference

The domain-less IRI of the Work, Expression, Manifestation, as specified in this document, or the full IRI of the Item, with all slash substituted with dots.

- `sl.act.2004-02-13.2`

Sierra Leone enacted Legislation. Act number 2 of 2004.

- `sl.act.2004-02-13.2.eng@2004-07-21`

Sierra Leone enacted Legislation. Act number 2 of 2004. English version, as amended on July 2004

- eu.bill.directive.~~cnl~~CONSIL.2013.eng@ver_second
European proposal for a Council directive. English variant in second version.

- `sl.act.2004-02-13.2.eng@2004-07-21.!schedule_1`

Attachment "schedule01" of Sierra Leone enacted Legislation. Act number 2 of 2004. English version, as amended on July 2004

- `sl.act.2004-02-13.2.eng@2004-07-21.akn`

Package of all documents including XML versions of the Sierra Leone enacted Legislation. Act number 2 of 2004. English version, as amended in July 2004

- `sl.act.2004-02-13.2.eng@2004-07-21.xml`

The main document (in XML) of the Sierra Leone enacted Legislation. Act number 2 of 2004. English version, as amended in July 2004

5. Identifying elements of document (Normative)

Many references need to point to items smaller than whole documents. In sections 4.7, 4.9 and 4.10 the syntax is provided to create IRI references to subdocument units, such as fragments, components and portions.

The description of these syntaxes include the specification for the name or for the identifier of the component or portion being referred to. In this section we discuss how the name and the identifier of these parts should be created and managed, in case the Manifestation being referred to uses the Akoma Ntoso XML vocabulary.

The Akoma Ntoso XML vocabulary uses three different attributes for the attribution of ids to elements. All of them are optional for basic conformance, and required for higher level conformance. For one of these attributes, `quid`, no syntax is prescribed. The two others, `eId` and `wId`, have a prescribed syntax. Therefore, while the use of `eIds` and `wIds` is not required for basic conformance, they MUST use the syntax described here IF used.

Non XML-based manifestations that use identifiers SHOULD adopt values for ids that are consistent with that specified in this section.

5.1 Fundamental principles identifiers in Akoma Ntoso

Concerning the Ids policy, our solution is based on some fundamental principles:

The Akoma Ntoso Naming Convention identifies, in a unique way, all Akoma Ntoso concepts and resources on the Internet and, in general, all collections thereof. These principles and characteristics should be respected in the use of identifiers for such documents:

- *Universality*: the approach taken works forthis specification applies to all document types that Akoma Ntoso deals with now or will deal with in the future. It works for amendable parts as well as non-amendable parts, for frequently modified parts as well as never modified parts. It works for original versions, single versions, multiple versions, and chains of versions.

- *Proportionality of impact*: the approach taken for a rare occurrence does not affect the solutions taken for the more frequent occurrences. It is better for the solution of a rare occurrence to be very convoluted than for the solution to a frequent occurrence to be even only mildly convoluted.

- *Uniqueness*: the identifier of a part is unique within the document.

- *Persistency*: the identifier of a part is persistent across versions, i.e., across all expressions of the same work. The persistency refers to the identity of the part, and not of the name or the number (i.e., if a part is moved and renamed the identifier accompanies the part, and does not stay with the number), so that it remainsis possible to track across versions the movement of the part across versions.

- *Navigability*: the identifier of a part is usable in an IRI reference as the fragment part (after the # signcharacter) or as the portion part (after the ~ character), even in the hypertext link oflinks to a separate document, and the link remains traversable to the right place regardless of what happened to the destination document.

- *Self-sufficiency*: the identifier of a part is the only information needed to perform the basic operations (in particular, navigation and tracking). Explanatory metadata are always optional so that it is not necessary to deal with tracking in a separate metadata section.

- *Contiguity*: the identifier of a part is near the part it refers to, e.g., as an attribute to the relevant element[4].

---

[4] Contiguity does NOT mean that the identifier must be called "id", or that it must be the only attribute to exhibit identification characteristics

- *Meaningfulness*: the identifier of a part expresses, as much as possible, basic facts about its nature, position, or relation to superior and/or neighboring elements that are meaningful to the local tradition.
- *Transferability*: the identifier of a part is transferable when the part is transferred from a document to another, or from a document type to another. For instance, identifiers of bills ~~is~~are transferable to the identifiers of the corresponding act once the document has been promulgated, and similarly, the identifier of a structure within an amendment proposal, possibly even in an oral discussion reported in a Hansard, is transferable to the identifier of the structure in a new version of the bill.[5]

5.2 Id attributes in the Akoma Ntoso XML vocabulary

There are three different attributes in Akoma Ntoso XML to identify content:

- `eId` **attribute** ("E*xpression-level" identifier*)
  This is the first and most important identifier. An `eId` attribute provides uniqueness of an element within a specific Expression. The value of `eId` is specified as connected to the structural role of the corresponding element. So, it needs to be updated regularly whenever the structural role of the element changes in a new Expression (i.e., if the element is renumbered or changed in nature, e.g., from article to clause).

- `wId` **attribute** ("*Work-level*" identifier)
  This attribute is only needed if the `eId` is not also a Work-level identifier. It is meant for mapping the identity and position of the same elements in different Expressions and variants of the same Work – `wId` identifier will be ~~added~~explicitly included when the `eId` changes from one Expression to another. The value of the `wId` attribute never changes; it must be the same values for the same elements in all the Expressions of a document. In order to allow this, a master Expression ~~could~~needs to be identified, i.e., the Expression whose `eId` attribute becomes the references for the `wId` attribute of all other Expressions.

- `GUID` **attribute** (*Globally Unique Identifiers*)
  This attribute is an application-specific identifier that a local implementation may need to add to elements according to local rules and syntaxes. `GUID` is not a required attribute. Its use and specification is totally dependent on the representation and storage requirements of the author of the Manifestation. ~~The usage of `GUID` attribute is not part of the second level of compliance to the Akoma Ntoso schema~~Despite the name, GUID do not even have to be globally unique across documents of the same collection. They are meant as a way to place ids from legacy formats and collections and that are still in use. The usage of `GUID` attribute has no required syntax and does not impact on compliance with this specification.

5.3 Syntax for eId and wId attributes

~~This section specifies~~Most elements in the Akoma Ntoso XML vocabulary allow two optional attributes, called `eId` and `wId`. They are optional for compliance level 1, and are required for compliance levels 2 and above.

The syntax for these attributes is identical and is built in the exact same way, but are based on different elements:

- for the `eId` attribute~~.~~, the reference element is the one in the current Expression

~~The second level of compliance to Akoma Ntoso requires following this syntax.~~

- for the `wId` attribute, the reference element is the one in the master Expression, as specified in section 5.4.

---

[5] Transferability does not mean "identical value", but only that a transformation between the two values must be possible in an automatic way.

The generic syntax for ~~an eId~~these attributes is the following:

```
[prefix "__"] element_ref ["_"number]
```

- `prefix` is a (possibly empty) string providing uniqueness to the remaining part of the identifier, and it is based on the context in which the element appears;
- `element_ref` is ~~an~~a (required) identifier of the type of the element;
- `number` is a (possibly empty) representation of the numbering of the element within its context.

5.3.1 Prefix

Frequently elements appear in different parts of the same document with the same number and they need to be identified by the context in which they appear: for instance, a *Chapter 2* may exist within both *Book I* and *Book II*, and a line 5 most probably exists in every page of some document.

The context is the part of the identifier that provides the required uniqueness. Thus, the context of the two instances of Chapter 2 is Book I or Book II respectively, while the context for each line 5 (i.e., each `<eol>` element) will be the page number of the page in which it appears (i.e., the immediately preceding `<eop>` element).

The prefix is a (possibly empty) string providing uniqueness to the remaining part of the identifier, and it is based on the context in which the element appears. In fact, by construction the prefix ~~of an identifier~~ is *the full identifier of the context element.*

For example:

`book_I__chp_2,`
The ~~elements that repeat with~~prefix part of Chapter 2 within Book I is "`book_I`", which is the ~~same number in different parts~~identifier of the ~~same document are frequent and need to be identified (i.e., a Chapter 2 may exist within both Tome I and Tome II, and line 5 most probably exists in every page of the document).~~

- ~~The concept of context has been introduced as the~~ element "Book I" that provides the ~~required uniqueness for an identifier. Thus, the context of the two instances of Chapter 2 will be Tome I and Tome II respectively, while the context for each line 5 (i.e., each `<eol>` element) will be the page in which it appears (i.e., the immediately previous `<eop>` element).~~context for the rest of the identifier.
- `eop_2__eol_4`
  The prefix part of the line 5 of page 3 is "`eop_2`", which is the identifier of the element between page 2 and page 3 that provides the context for the rest of the identifier.

Composite documents make it more complex to reach uniqueness of identifiers over the whole XML document, since they might be the result of composing individual documents where the same identifiers ~~where created independently~~may exist independently. For instance, an element called *section 2* may exist independently in the main component and in an annex of the same document.

The identifier of an element ~~must~~MUST therefore include the full identifier of the context element that guarantees its uniqueness, ~~be~~and it CAN be the identifier of the individual document in a composite document, the identifier of a wrapping element that restarts the numbering, or the identifier of a preceding element that restarts the numbering.

Structures within the `<quotedStructure>` and `<embeddedStructure>` elements add the relevant mod identifier before their "natural" identifiers. In a way, `<quotedStructure>` and `<embeddedStructure>` act as the context for the contained structures. So for instance if clause 3 of article 15 has an amendment that adds article 4/a to a different act, the identifier of the `<quotedStructure>` element that contains the new article will be "`art_15__cl_3__mod_1__qstr_1`", and the identifier of article 4/a inside it will be "`art_15__cl_3__mod_1__qstr_1__art_4a`". Of course, automatic systems that create current versions of texts should and will remove the prefix belonging to the modification law and will only keep the identifier "`art_4a`" in the final result.

The following are usual cases of contexts:

- All document classes (`<act>`, `<bill>`, `<doc>`, etc.) are always contexts. This means that, except particular cases, all numbers restart whenever a new document class is started (e.g., in a composite document each document component has its own local numbering).
- Elements `<quotedStructure>` and `<embeddedStructure>` are always contexts, even if they do not force a restart of the numbering, but just a different numbering context within themselves.
- Plain inline elements are never contexts. Exception: element `<mod>` is always a context.

5.3.2 element_ref

The `element_ref`, part of the `eId` attribute value, is generally based on the name of the element; however, sometimes the `element_ref` could be based on the content of the element.

For example, this is the case for elements that play the role of "reference table."

Elements Based on the Name of the Element

In general, the *element_ref* part is the full name of the element. There are ~~two~~three exceptions:

1. For some elements, an abbreviation is used. This abbreviation is a well-known abbreviation.

~~2.~~1. Some elements share the same *element_ref*. This ~~is the case~~happens when the two elements has the common user semantic but are found in different structural context~~s~~. These elements are:

- `<list>` and `<blockList>` (identifier "list")
- `<intro>` and `<listIntroduction>` (identifier "intro")
- `<wrapUp>` and `<listWrapUp>` (identifier "wrapup").

The reason for these elements to have the same `element_ref` is to reduce the dependence on a technical markup choice where the structure is functionally identical.

5.3.2.1.1 Abbreviations

The list of abbreviations:

XML elements	~~Abbreviation~~*element_ref*
alinea	al
article	art
attachment	att
`<list>` and `<blockList>`	list
chapter	chp
citation	cit
citations	cits
clause	cl
component	cmp

XML elements	Abbreviation element_ref
components	cmpnts
componentRef	cref
debateSection	dbsect
division	dvs
documentRef	dref
eventRef	eref
intro	intro
list	list
`<intro>` and `<listIntroduction>`	intro
listWrapUp	wrap
paragraph	para
quotedStructure	qstr
quotedText	qtext
recital	rec
recitals	recs
section	sec
subchapter	subchp
subclause	subcl
subdivision	subdvs
subparagraph	subpara
subsection	subsec
temporalGroup	tmpg
`<wrapUp>` and `<listWrapUp>`	wrapup
`<body>`, `<mainBody>`, `<amendmentBody>`, `<debateBody>`, `<judgmentBody>`	body

~~Elements Based on the Content~~

~~Theses~~The reason for these elements ~~are:~~

- ~~<TLCConcept>~~
- ~~<TLCEvent>~~
- ~~<TLCLocation>~~
- ~~<TLCObject>~~
- ~~<TLCOrganization>~~
- ~~<TLCPerson>~~
- ~~<TLCProcess>~~
- ~~<TLCReference>~~
- ~~<TLCRole>~~
- ~~<TLCTerm>~~
- ~~<componentData>~~
- ~~<keyword>~~

~~<component>: depending on~~to have the same *element_ref* is to reduce the ~~document inside,~~complexity whenever the ~~identifier can be "annex" or "attachment" or … , with explicit or implicit number~~structures are functionally identical.

2. For some elements, an abbreviation is used. This abbreviation is a well-known shorthand of the full element name that is already used in legal citations. The following is the full list of abbreviations allowed:

XML element	element_ref
<alinea>	al
<amendmentBody>	body
<article>	art
<attachment>	att
<blockList>	list
<chapter>	chp
<citation>	cit
<citations>	cits
<clause>	cl
<component>	cmp
<components>	cmpnts
<componentRef>	cref
<debateBody>	body
<debateSection>	dbsect
<division>	dvs
<documentRef>	dref
<eventRef>	eref
<judgmentBody>	body

XML element	element_ref
`<intro>`	`intro`
`<list>`	`list`
`<listIntroduction>`	`intro`
`<listWrapUp>`	`wrap`
`<mainBody>`	`body`
`<paragraph>`	`para`
`<quotedStructure>`	`qstr`
`<quotedText>`	`qtext`
`<recital>`	`rec`
`<recitals>`	`recs`
`<section>`	`sec`
`<subchapter>`	`subchp`
`<subclause>`	`subcl`
`<subdivision>`	`subdvs`
`<subparagraph>`	`subpara`
`<subsection>`	`subsec`
`<temporalGroup>`	`tmpg`
`<wrapUp>`	`wrapup`

3. For some elements, a string representing some information about the content of the element is used. These elements are:

XML element	element_ref
`<TLCConcept>`	a label of the concept;
`<TLCEvent>`	a label of the event;
`<TLCLocation>`	a label of the location;
`<TLCObject>`	a label of the object;
`<TLCOrganization>`	a label of the organization;
`<TLCPerson>`	a label of the person;
`<TLCProcess>`	a label of the process;
`<TLCReference>`	a label of the reference;
`<TLCRole>`	a label of the role;
`<TLCTerm>`	a label of the term;
`<componentData>`	a name for the component;
`<keyword>`	a label of the keyword;
`<component>`	depending on the document inside, the identifier can be "annex" or "attachment" or the name of the component , with an explicit number.

5.3.3 Number

The `number` part of an identifier is a (possibly empty) representation of the numbering of the element within its context.

There are three subcases:

10. Globally and locally unique elements:

    If the element is necessarily unique within its context, no numbering is used, and therefore there is no `number` part. For instance, since there is exactly one `body` in acts and bills, its identifier can be simply "`body`" (or, of course, "`doc_1__body`" in case of a composite document). Analogously, since there is at most one `content` element inside articles or sections, the identifier of the `<content>` element of article 12 will be simply "`art_12__content`".

11. Explicitly numbered elements

    An explicitly numbered element has its number determined in the Expression itself in the form of a `<num~~ber~~>` sub-element. The `number` part of the identifiers of such elements corresponds to the stripping of all <u>final</u> punctuation, <u>meaningless</u> separations as well as redundant characters in the content of the `<num>` element. The representation is case-sensitive. For instance, if article 12 contains `<num>Art. ~~12~~11.2 bis</num>` then the `number` part of the identifier will be "~~12bis~~<u>11-2bis</u>". It is the job of the author of the Manifestation to determine whether the numbering expressed in the `<num>` element is global (i.e., it starts at 1 at the beginning of the document) or local (i.e., it restarts at 1 inside or after every instance of an intermediate element). This is usually made clear within every legal tradition and usually can be established by briefly examining a few or even just one document in its original form. <u>Meaningful separations and punctuation are never stripped. For instance, article -1 (in some jurisdictions it represents a late insertion before article 1) will be "`art_-1`".</u>

12. Implicitly numbered elements.

    An implicitly numbered element has no `<num>` sub-element, and its numbering is established by counting the occurrences of similar elements within the same context, ~~necessarily~~<u>always</u> using Arabic numbers. It is the job of the author of the Manifestation to determine whether the best way to count these elements is globally (i.e., starting at 1 at the beginning of the document class) or locally (i.e., restarting at 1 inside or after every instance of an intermediate element). This naming convention provides no rules on this choice, but there are a few common sense approaches. For instance, it is very natural that `<eop>` elements are globally counted, and `<eol>` are locally counted by their preceding `<eop>` element, and as such, the third `<eop>` element (the one separating the third page from the fourth) has identifier "`eop_3`" (note no prefix), while the fifteenth end of line after such `eop` (the one separating the fifteenth line from the sixteenth) will have as identifier "`eop_3__eol_15`". On the other hand, `<p>` elements within a given structure are probably counted locally (as in "third `<p>` of section 12"). This is not necessarily the immediately containing element (which in this case would be the `<content>` element), but any containing or preceding element that in the opinion of the author of the Manifestation provides context for the counting. Thus the third `<p>` of section 12 could reasonably have "`sec_12__p_3`" as its identifier.

5.4 Usage Rules for "eId" and "wId"

Documents are complex structures. Sometime, it is important to record the fact that the (conceptually) same structure may have different content (e.g., for different languages, different versions or different audiences).

Permanent identifiers are the basic tool to be able to identify the concept of *sameness* across situations that require different content to be known as really being the same. Unfortunately, relying only on a permanent identifier prevents some common and very useful operations to be performed on documents that present multiple instances of the same structure. For this reason the concepts of Expression identifiers (`eId`) and Work identifiers (`wId`) have been introduced.

5.4.1 Elements That Require an eId Attribute

The use of attribute `eId` is optional for conformance level 1, and required for conformance level 2 or more. If attribute `eId` is used, then it ~~must~~MUST be used according to the syntax in section ~~6~~5.2. If attribute `eId` is used in a document, then it is required for all elements that use or include attribute group `idreq`, and optional in all elements using or including attribute group `idfac`.

~~5.4.2 wId Attribute Usage~~

5.4.2 The Master Expression

13. Whether an XML document does or does not have Work-level identifiers is NOT a decision of the ~~marker~~author of the Manifestation, but a characteristic of the nature of the document. In fact, if an XML document does NOT have Work-level identifiers, then it is assumed that:

    (a) ~~this~~it is the Master Expression (the one whose Expression-level identifiers will be used as a map for the Work-level identifiers of all the other expressions) and

    (b) its Work-level identifiers are the same as Expression-level identifiers.

If this is NOT the Master Expression, then the Work-level identifiers NEED to be present as soon as they become different from the corresponding Expression-level Identifiers.

Master Expressions are necessarily the FIRST (or the ONLY) time-related versions of a document that either is intrinsically MONOLINGUAL or is expressed in the MASTER LANGUAGE, which is country- and jurisdiction- dependent and may even not exist (as in EU).

A ~~marker~~Manifestation author must know whether the document ~~he/she is marking~~being marked up ~~is~~should be considered as the Master Expression ~~or not~~ for a Work or not.

The Expression-level identifiers use a semantic naming convention based on the structure of their Expression. The Work-level identifiers use a semantic naming convention based on the structure of their Master Expression, if one exists, or of a conceptual Ur-Expression, if none exists.

5.4.3 wId Attribute Use Cases

The ~~risk here~~decision to use two separate identifiers is to ~~collapse~~avoid the risk of collapsing two potentially very different meanings of "identification" into just one identifier: the identifier of the *right place* (*the one that I mean now when I use this identifier)* and the identifier of the *same place* (*the one that had such an identifier in a different version or in a different variant of this document*).

In fact there are really two identifiers at work: one has the purpose of matching the evolving nature of the fragment with respect to the internal structure of the document and the other must guarantee the persistency of the identity of the fragment across versions and variants. They are usually the same, and diverge only when one of the four following situations occur:

14. In multilingual works, the concurrence of multiple similarly named structures in multiple expressions, say article 2 in the French version of a document and section 2 in the English version of the same document, both referring to the same conceptual structure. A first subcase of this situation is when a master language can be identified, which provides a clear Master Expression, and a second subcase is when no master language is provided (e.g., in European legislation), so that no Master Expression exists.

15. In a multi-version file, the co-occurrence of two similarly named structures from two versions, say article 2 in the past version and article 2 in the current version, both contained in the same (multi-version) Manifestation.

16. In a modification act, the concurrence of two similarly named structures of the amending and of the amended document, say I am amending art.2 of the amended act, and of course an art.2 that exists already somewhere else in the amending act.

17. In a chain of versions, the requirement to renumber a few structures that completely desynchronizes the old identification mechanisms from the new one, e.g., article 2 is from now on known as article 15. Such renumbering is frequent in bills, and rare in acts. But external references to bills are mainly

static (i.e., they refer to a specific version of a bill), while external references to acts are often dynamic (i.e., they refer to any of a number of versions depending on the nature of the quest).

- First use case – renumbering in bills: an approved amendment A inserts a new article between art.1 and art.2 of version 1 of bill B. Because of this decision, art.2 is known in version 2 of B as art.3, art.3 is known as art.4, etc.

- Second use case – renumbering in acts: while act Y is in version 1, on date D1 act X makes a (dynamic) reference to art.2 of act Y. Subsequently, on date D2, act Y gets renumbered, and in version 2 art.2 becomes art.15 and a new art.2 is introduced in its stead. Subsequently, on date 3, act W makes a (dynamic) reference to art.15 of act Y (which is the new name for art.2) and on date D4 act Z makes a reference to art.2 of act Y (which is a new article that did not exist previously).

Given the above-mentioned principles, the natural solution is to have two identifiers to deal with. One is persistent and associated to the Work, while the other is evolving in time and associated to the Expression. Whenever the persistent identifier and the evolving identifier do not differ, only one of them is specified in the document, but when they differ, then the evolving identifier follows the structure of the Expression, while the permanent identifier is anchored to the structure of one specific Expression, called *Master Expression*, which is considered as the fundamental Expression for the permanent identification of fragments.

When a situation occurs that requires the two identifiers to differ from each other, such as one of the above-mentioned situations 1, 2, 3 or 4, then the `eId` attribute is set to reflect the new role and number of the element in the structure, while the `wId` attribute is added and set to reflect the identity that such fragment had, has or would have in the Master Expression. Thus, after any change in the document, the Work-level identifier (`wId`) is added and never changes, and the Expression identifier (`eId`) keeps on being updated according to the ~~new data~~condition of the Expression it belongs to.

Tracking is always based on the `wId`, navigation is always based on the identifier that was the `eId` at the moment, and transfer is always based on the `eId`. ~~Since~~If the evolving identifier ~~may change~~changes in time more than once, ~~a~~the Akoma Ntoso vocabulary provides the `<mapping>` metadata ~~structure has been added~~ to hold a complete map in time of the relationships between the persistent identifier `wId` and each of the evolving identifiers `eId`.

For instance~~, using the following simplified naming convention: `doc@vers#fragment`~~, we can describe the four situations as follows.

Multi-Lingual Document

Subcase a

Two expressions exist in two different languages. One is in the ~~standard~~master, or default language, and the other is an additional variant~~s~~ in a different language. As such, the version in the default language is the master Expression, and the other version uses the master Expression's identifiers as `wIds`.

~~How do we represent~~Representing the fact that article 2 in the ~~French version~~secondary variant contains the same text as section 2 in the ~~English version, which is the~~ master Expression~~?~~ is performed as follows:

Master Expression (e.g., in English)

```
<section eId="sec_2">
   <num>2</num>
   <content>
            <p>Some text in English</p>
   </content>
</section>
```

Variant (e.g., in French)

```
<art wId="sec_2" eId="art_2">
   <num>2</num>
```

```
      <content>
                <p>Du texte en Français</p>
      </content>
</art>
```

In this context, a reference such as doc#sect_2 points by default to the default destination, but a client-side script could, upon signaling that the user has a specific language preference, locally fiddle with the identifiers to have the destination change.

Subcase b

Two expressions exist in two different languages, but neither can be determined as the default or master language. This happens, for instance, in European legislation. As such, the master Expression does not exist in a concrete Expression, but must be determined abstractly (it would also could be called an *UR-Expression*), and both versions uses the UR-Expression's identifiers as `wIds`.

How do we represent that article 2 in the French version contains the same text as section 2 in the English version, and neither is the master Expression?:

Variant (e.g., in English)

```
<section wId="elm_2" eId="sec_2">
   <num>2</num>
   <content>
                <p>Some text in English</p>
   </content>
</section>
```

Variant (e.g., in French)

```
<art wId="elm_2" eId="art_2">
   <num>2</num>
   <content>
                <p>Du texte en Français</p>
   </content>
</art>
```

Multi-Version Document

In a multi-version file, two similarly named structures exist coming from two different versions, say article 2 in the past version and article 2 in the current version, both must be placed in the same (multi-version) Manifestation.

The "default" fragmentstructure (e.g., the current version) uses a plain `eId` identifier; the "secondary" destinationstructure (e.g., the past version) uses a modified `eId` identifier and is associated to the default structure via the `wId` identifier.

```
<art eId="art_2">
<num>2</num>
   <content>
                <p>New version of art.2</p>
   </content>
</art>
```

```
<art wId="art_2" eId="art_2v1">
   <num>2</num>
   <content>
```

```
                    <p>Old version of art.2</p>
    </content>
</art>
```

In this situation it is assumed that the expected default behavior when traversing documents is to go to the newer version of the document, and if the navigation mechanism knows something more specific about the needs of the user, it would lead to the older version of the fragment instead.

Amending Act

The structured content uses `wId` as a suggestion of the identifier that the structure will have in the new version of the amended document:

```
<mod eId="mod_1">
   Art. 5 is changed as follows:
   <quotedStructure eId="mod_1__qstr_1">
        <art eId="mod_1__qsrt_1__art_5" wId="art5art 5">
             ...
        </art>
   </quotedStructure>
</mod>
```

Renumbering of a Bill

The first version of the bill has simple identifiers:

```
<article eId="art_1">
   <num>1</num>
   <content><p>Originally article 1</p></content>
</article>
<article eId= »"art 2 »>">
   <num>2</num>
   <content><p>Originally article 2</p></content>
</article>
<article eId="art_3">
   <num>3</num>
   <content><p>Originally article 3</p></content>
</article>
```

TheIn the second version of the bill, after a new article 2 was inserted, generatingwhich generated a renumbering of the subsequent articles,. This version uses identifiers to specify the original identifiers, regardless of the position, and uses `eId` to specify the identifier each article should have if this were a new document.

```
<article eId="art_1">
   <num>1</num>
   <content><p>Originally article 1</p></content>
</article>
<article eId= »"art_2 »>">
   <num>2</num>
   <content><p>New article 2</p></content>
</article>
<article wId="art_2" eId="art_3">
   <num>3</num>
```

```
   <content><p>Originally article 2</p></content>
</article>
<article wId="art_3" eId="art_4">
   <num>4</num>
   <content><p>Originally article 3</p></content>
</article>
```

Since bills mostly receive static references, and since static references always include the version number, it is always very clear what refers to what – ~~bill@v1#~~references to art_2 ~~refers~~in version 1 of the bill will lead to the same article as ~~bill@v2#~~references to art_3 ~~and different than bill@v2#~~in version 2, while references to art_2~~.~~ in version 2 will lead to the newly inserted article.

Renumbering of Acts

~~The structure of the document is similar to the bills' example, and given the use case "~~Suppose that, while act Y is in version 1, on date D1 act X makes a (~~dynamic~~Work-level) reference to art.2 of act Y. Subsequently, on date D2, act Y gets renumbered, and in version 2 art.2 becomes art.15 and a new art.2 is introduced in its stead. Subsequently, on date 3, act W makes a (dynamic) reference to art.15 of act Y (which is the new name for art.2) and on date D4 act Z makes a reference to art.2 of act Y (which is a new article that did not exist previously)~~", then the following are ideas for the XML conversion:~~)".

The structure of the document is similar to the previous example. Then the following facts would be true[6]:

- In act X the Work-level reference ~~is Y#~~to art~~.~~2~~.~~ of act Y would be is Y/~art_2. This corresponds to the structure that in version 1 had eId="art_2".

- In act W the Work-level reference ~~is Y#~~to art~~.~~15~~.~~ of act Y is Y/~art_15. This corresponds to the same structure that in version 2 had eId="art_15" and wId="art_2"

- In act Z the Work-level reference to art.2 of act Y is Y~~#~~/~art_2. This corresponds to the structure that in version 2 had eId="art_2" and no wId.

When accessing one of these references, the date of the source Expression (i.e., the one containing the reference must be compared with the date of the Expression of the destination document. If the date of the source is before the date of the destination, then the wId must be used for navigation. Otherwise the eId must be used.

---

[6] We use here a simplified version of the Naming Convention, [doc]/~[portion], where [doc] is the IRI reference for the whole document, and [portion] is the identifier of the portion.

6. Conformance

This chapter defines ~~four~~ Akoma Ntoso conformance clauses. ~~In order to conform to the~~

#1 an XML document is compliant with Akoma Ntoso specs~~:~~ in level 1 if

The XML file MUST be valid according to the XML schema: http://docs.oasis-open.org/legaldocml/ns/akn/3.0/WD17~~http://docs.oasis-open.org/legaldocml/ns/akn/3.0/CSD13;~~;

#2 an XML document is compliant with Akoma Ntoso specs in level 2 if it is compliant at level 1 and

> 2.1 The values of the `eId` and `wId` attributes MUST follow the Akoma Ntoso naming convention as formulated in chapters 4 and 5;

> 2.2 The values of the `FRBRuri` and `FBRRthis` elements MUST follow the specification detailed in chapter 4;

> 2.3 The values of the `href` and `src` attributes in ALL elements (except <a>) MUST follow the specifications detailed in chapter 4 and 5.

Appendix A. Acknowledgments

The following individuals have participated in the creation of this specification and are gratefully acknowledged:

Participants:

Aisenberg, Michael, Mitre Corporation
Arocena, María de la Paz, Uruguay Parliament
Barysheva, Nataliya, LexisNexis a Division of Reed Elsevier
Bbaale, Fred, Uganda Parliament
Beatch, Richard, Bloomberg Finance L.P.
Bennett, Daniel, Individual Member
Briotti, Giuseppe, Senato della Repubblica d'Italia
Bruce, Tom, Cornell Law School, Legal Information Institute
Cabral, James, MTG Management Consultants, LLC.
Dohaini, Bassel, Lebanese Parliament
Dunning, John, LexisNexis a Division of Reed Elsevier
Fabiani, Claudio, EU Parliament
Ferguson, Kimberly, Library of Congress
Ferreira, Daniel, Uruguay Parliament
Fiagome, Shirley-Ann, Ghana Parliament
Gheen, Tina, Library of Congress
Greenwood, Dazza, M.I.T.
Hardjono, Thoma, M.I.T.
Hariharan, Ashok, Africa i-Parliaments Action Plan (UN/DESA)
Harris, Jim, National Center for State Courts
Joergensen, John, Cornell Law School, Legal Information Institute
Junge, Peter, Beijing Sursen Electronic Technology Co, Ltd.
Khamis, Mr. Maan, LexisNexis a Division of Reed Elsevier
Marchetti, Carlo, Senato della Repubblica d'Italia
Mattocks, Carl, Individual Member
Murungi, Michael, Kenya National Council for Law Reporting
Otto Eridan, Biblioteca del Congreso Nacional de Chile
Palmirani, Monica, University of Bologna
Parisse, Véronique, Aubay S.A.
Petri, Steve, LexisNexis a Division of Reed Elsevier
Pham, Kim, US Military Health Services
Ramsahye-Rakha, Saseeta, Mauritius National Assembly
Sandoval, Alvaro, Biblioteca del Congreso Nacional de Chile
Shifrin, Laurel, LexisNexis a Division of Reed Elsevier
Sifaqui, Christian, Biblioteca del Congreso Nacional de Chile
Sosa, Raquel, Uruguay Parliament
Sperberg, Roger, LexisNexis a Division of Reed Elsevier
Tosar Piaggio, Sylvia, Uruguay Parliament
Vergottini, Grant, Xcential Group, LLC.
Vitali, Fabio, University of Bologna
Waldt, Dale, LexisNexis a Division of Reed Elsevier
Weber, Andrew, Library of Congress
Wemer, Jason, Wells Fargo
Wintermann, John, Bloomberg Finance L.P.
Zeni, Flavio, Africa i-Parliaments Action Plan (UN/DESA)

Appendix B. Revision History

Revision	Date	Editor	Changes Made
[01]	11 June, 2014	[Veronique Parisse]	[Original]
[02]	25 June, 2014	[Veronique Parisse]	[Add scope section. Inverted the part of ID with the URI/IRI]
[03]	25 June, 2014	[Monica Palmirani]	[Added the Conformance section]
[04]	1 July, 2014	[Veronique Parisse]	[replace attribute "id" by eId in 6.3.3.4; wrap -> wrapUp; correct formatting of list in 6.2.3]
[05]	1 July, 2014	[Monica Palmirani]	[add AKN in any URI, some micro typos]
[06]	1 July, 2014	[Tina Gheen]	[English language cleanup]
[07]	8 July, 2014	[Grant Vergottini]	[English language cleanup]
[08]	16 July, 2014	[Jason Wemer] [Veronique Parisse] [Monica Palmirani]	[English language cleanup] [Change in the naming convention character] [Consolidation and styling]
[09]	23 July, 2014	[Jason Wemer] [Monica Palmirani]	[English language cleanup] [Consolidation and styling]
[10]	24 July, 2014	[Monica Palmirani]	[pagg. 26 and 29 "num" is substituted with "number"]
[11]	29 July, 2014	[Monica Palmirani]	[inclusion of Chet Ensign comments]
[12]	30 July, 2014	[Fabio Vitali]	[minor modifications]
[13]	17 September, 2014	[Veronique Parisse]	[minor error in the syntax of the examples]
[14]	22 December 2014	[Monica Palmirani, Veronique Parisse]	[inclusion of the "mul" and "und" examples for the language expressions; inclusion of the part related to the query server-side of the portion; inclusion of the examples of time in the expressions; clarification of the TLCReference]
[15]	8 January, 2015	[Grant Vergottini]	[linguistic revision and quality check]
[16]	13 January, 2015	[Veronique Parisse]	["ver" prefix for the versions not defined by a precise date]
[17]	14 January 2015	[all]	detailed
[18]	24 June, 2015	[Monica Palmirani]	Inclusion of comment refinement
[19]	26 October 2015	[all]	[fragment and component refinement: introduction of ~]

[20]	4 November 2015		[harmonization of examples, component refinement, introduction of ~, change of the ! meaning]
[21]	17 December 2015		[harmonization of examples, component refinement, introduction of ~, change of the ! meaning]
[22]		[Veronique Parisse]	
[23]		[Fabio Vitali]	Strong revision of sections 2,3 and 4, inclusion of /!component~portion
[24]		Monica Palmirani	Minor modifications
[25]	9 March 2016	Fabio Vitali	Strong revision of section 5
[26]	16 March 2016	Monica Palmirani	Refinement. -> for intervals.
[27]	31 March 2016	Monica Palmirani	Editing.
[csd02]	4 May 2016	Monica Palmirani	Put in approval as csd02