# Mapping rules between Metanorma XML (presentation) and ISO/NISO STS

Note 1: This mapping is not full ISO/NISO STS standards comparison and based on 30 XML samples only:

* C070907e.xml
* C070908e.xml
* BSI STS sample XMLs

Note 2: ISO STS mapping rule is not given if it is identical to NISO STS structure.

Note 3: NISO STS description:

* <https://www.niso-sts.org/TagLibrary/niso-sts-TL-1-0-html/chapter/modules.html>

Note 4: in the tables below, the value ‘no’ means that there isn’t corresponding element in XML, but it doesn’t mean that the data will be ignored at all, the text inside this element will be copied in result xml in any case, or internal elements will be processed.

## Introduction

Mapping rules realized in the ‘mnconvert’ tool (<https://github.com/metanorma/mnconvert/>). This command line tool based on XSLT 1.0 (with Xalan extensions) and allows to convert data:

* from ISO/NISO STS xml to Metanorma ADOC (module sts2mn.adoc.xsl)
* from ISO/NISO STS xml to Metanorma XML (presentation) (module sts2mn.xsl)
* from Metanorma XML (presentation) to ISO/NISO STS XML (module mn2sts.xsl)

Stages of conversion from ISO/NISO STS xml to Metanorma ADOC (sts2mn.adoc.xsl):

* XML linearization – replace all line breaks (except text in the tags ‘code’, ‘preformat’ and ‘mml:’) to space, replace sequence of spaces into one space
* **replace ‘bold’, ‘italic’, ‘underline’, ‘sup’ and ‘sub’ tags to ‘bold2’, ‘italic2’, ‘underline2’, ‘sup2’ and ‘sub2’ for unconstrained formatting pair (**<https://docs.asciidoctor.org/asciidoc/latest/text/#unconstrained>) for further processing
* reference’s id normalization in the elements ‘ref’ and ‘std’
  + replacing a space, non-break space, colon to low line char ‘\_’ (U+5F)
  + replacing non-breaking hyphen minus to hyphen minus char ‘-‘ (U+2D)
  + adding ‘low line’ char ‘\_’ (U+5F) prefix if ‘id’ starts with digit
* conversion from ISO/NISO STS xml to Metanorma ADOC
* checking for repeated bibliography references
* copy images from the folder, specified by ‘imagesdir’ (command line parameter, default value ‘images’), to the output folder

Note: first page data and each section (first level clause/annex) saves in a standalone .adoc file.

Stages of conversion from ISO/NISO STS xml to Metanorma XML (presentation) (sts2mn.xsl):

* conversion from ISO/NISO STS xml to Metanorma XML (presentation)
* if non-processed elements in front page (bibliographic) data found, then it will be added at the end of ‘bibdata’ element with message ‘*WARNING! There are unprocessed elements in 'front'*’
* if non-processed elements found, then such element’s names will be added in comments at the end of xml file
* checking on repeated bibliography references
* copy images from the folder, setup by ‘imagesdir’ (command line parameter, default value ‘images’), to the output folder

Stages of conversion from Metanorma XML (presentation) to ISO/NISO STS xml to (mn2sts.xsl):

* conversion from Metanorma XML (presentation) to ISO/NISO STS xml
* if non-processed elements in front page (bibliographic) data found, then it will be added at the end of ‘bibdata’ element with message ‘*WARNING! There are unprocessed elements in bibdata*’

Legend:

**< >** - both direction rule (STS NISO/ISO XML to Metanorma XML (presentation), and vice versa)

**>** - one direction rule, from STS NISO/ISO XML to Metanorma XML (presentation)

**<** - one direction rule, from Metanorma XML (presentation) to STS NISO/ISO XML

## Main parts (front page, body, annexes)

### Root element

NISO STS: standard, adoption

Metanorma: abc-standard xmlns=” https://www.metanorma.org/ns/abc”

Where abc – type of standard – iso or bsi. Determines automatically in sts2mn.xsl:  
- if /standard/front/nat-meta/@originator = 'BSI' or /standard/front/iso-meta/secretariat = 'BSI', then ‘bsi’

- else ‘iso’,

OR can be set up any value in the command line via parameter *--type iso*

### Sections

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| front/sec [@sec-type=”intro”] | < > | preface/introduction | [[introduction]] |
| front/sec [@sec-type=” foreword”] | < > | preface/foreword | [.preface]  [[foreword]] |
| front/sec [@sec-type] | < > | preface/clause | [.preface]  [[id]]  if @sec-type = 'amendment', then  [.preface,type=corrigenda] |
| front/sec | < > | preface/clause | [.preface] |
| front//sec | < > | preface/clause | [[id]] |
| front//sec/@sec-type  Example: <sec sec-type="foreword\_publishing" id="sec\_foreword\_1"> | < > | preface/clause/@sec-type  Example: <clause id="sec\_foreword\_1" type="foreword\_publishing">  Note: Metanorma XML (presentation) model doesn’t have `sec-type` attribute. It’s added for round-trip conversion. |  |
| front/sec[ @id="sec\_pub\_info\_nat" @sec-type="publication\_info"]  Note: Publishing and copyright information block | < | boilerplate/ copyright-statement + docidentifier[@type = ‘ISBN’] + ext/ics + preface/ clause[ @type = 'related-refs']  + preface/ clause[ @type = 'corrigenda'] |  |
| front/sec[@sec-type=”publication\_info” and node()/@content-type = 'ace-table'] | > |  | [.preface,type=corrigenda]  == node() @content-type = 'ace-table']/caption/title |
| body | < > | sections |  |
| body/sec [@sec-type=”intro”] if there is /standard/front/nat-meta and there isn’t iso-meta or reg-meta | < > | preface/introduction | [[introduction]] |
| body/sec [@sec-type=”intro”] | < > | sections/clause[@type=”intro”] | [[introduction]] |
| body//sec[term-sec] | < > | terms |  |
| body//sec[term-sec]/@id  Example: <sec id="sec\_3" sec-type="terms">  <label>3</label><title>Terms and definitions</title> <p>For the purposes of this document…</p> … <term-sec id="sec\_3.1"><label>3.1</label> … | < > | terms/@id  Example: <terms id="sec\_3">  <title>3<tab/>Terms and definitions</title>  <p>For the purposes of this document</p> … <term id="term\_3.1">  <name>3.1</name> … | [[id]] |
| body//sec[array[count(table/col) = 2]] |  | definitions |  |
| body/sec[@sec-type=“scope”] | < > | sections/clause[@type=”scope”] |  |
| sec/@id | < > | clause| introduction |foreword | terms | indexsect | references | definitions /@id | [[id]] |
| sec/label (if there isn’t ‘title’) | > | clause[@inline-header=”true”]/name | ==== {blank} |
| sec/title | < > | clause/title  Note: title=sec/label<tab/>sec/name | === ...  Note: count of === depends of section’s depth. |
| sec[@sec-type = 'norm-refs'] | < > | bibliography/references[@normative="true"] | [bibliography]  [NOTE,type=boilerplate]  --  ..  -- |
| back/ref-list  no nested ref-list | < > | bibliography/references[@normative="false”] | [bibliography] |
| back/ref-list/@id  no nested ref-list | < > | bibliography/references[@normative="false”]/@id |  |
| back/ref-list  with nested ref-list | < > | bibliography/clause/references[@normative="false”] | [bibliography] |
| back/ref-list/@id  with nested ref-list | < > | bibliography/clause/@id | [[id]] |
| back/ref-list/@content-type  Example:  content-type="bibl" | > | no |  |
| back/ref-list/title | < > | references/title | === ...  Note: count of === depends of section’s depth. |
| back/ref-list/ref-list | < > | bibliography/clause/references[@normative="false”] |  |
| back/ref-list/ref-list/@id | < > | bibliography/clause/references[@normative="false”]/@id |  |
| sec[@sec-type = 'index'] | < > | indexsect |  |
| back/sec[@id = 'ind'] | > | indexsect |  |
| back | < > | *skip processing*  Note: see nested elements processing |  |
| app-group | < > | *skip processing*  Note: see nested elements processing |  |
| app | < > | annex | [appendix] |
| app/@id | < > | annex/@id | [[id]] |
| app/@content-type  Example:  content-type="inform-annex" | > | no |  |
| app/@content-type="inform-annex" | < | no |  |
| app/@content-type=" norm-annex"  Note: for BSI only | < | no |  |
| app/annex-type | < > | annex/@obligation  (without ‘()’) | [appendix, obligation=…]  Note: if annex-type != normative |
| app/@content-type=inform-annex | < | annex/@obligation = informative |  |
| app/@content-type | < | annex/@obligation |  |
| app//sec | < > | annex//clause |  |
| sec/@sec-type="titlepage","endorsement"  Note: conversion gap  <https://github.com/metanorma/mn-samples-bsi/issues/34> | > | no |  |
| no | < | foreword/review |  |
| sec[@sec-type="appendix"] | < | annex/appendix  Note: for more info, see <https://github.com/metanorma/mn2sts/issues/10> |  |
| sub-part  Note: realized partially yet (for adoc output only) | > | metanorma-collection  Note: to do <https://github.com/metanorma/mnconvert/issues/6> | yaml collection document  directives:  - documents-inline  bibdata:  type: collection  docid:  type: bsi  …  see <https://github.com/metanorma/sts2mn/issues/33> |
| no | < | clause/@obligation  foreword/@obligation  introduction/@obligation  terms/@obligation  Example:  obligation="informative"  obligation="normative" |  |

## Front page (bibliographic) data

STS xml can contains nat-meta, iso-meta, reg-meta and std-meta at the same time. The conversion rules are:

* if there is nat-meta, then
  + nat-meta goes to bibdata
  + iso-meta goes to bibdata/relation[@type = 'adopted-from']/bibitem/contributor[role/@type = 'publisher']/organization[abbreviation = 'xxx'] (xxx = ISO or IEC)
  + reg-meta goes to bibdata/relation[@type = 'adopted-from']/bibitem/contributor[role/@type = 'publisher']/organization[abbreviation = 'xxx'] (xxx = CEN or CENELEC)
  + std-meta goes to bibdata/relation[@type = 'adopted-from']/bibitem/contributor[role/@type = 'publisher']/organization[abbreviation = 'xxx'] (xxx != ISO, IEC, CEN, CENELEC)
* if there isn't nat-meta, then
  + iso-meta goes to bibdata
  + reg-meta goes to bibdata/relation[@type = 'adopted-from']/bibitem/contributor[role/@type = 'publisher']/organization[abbreviation = 'xxx'] (xxx = CEN or CENELEC)
  + std-meta goes to bibdata/relation[@type = 'adopted-from']/bibitem/contributor[role/@type = 'publisher']/organization[abbreviation = 'xxx'] (xxx != ISO, IEC, CEN, CENELEC)

In STS->Metanorma ADOC conversion, only first (by priority nat-, iso-, reg-, std-meta) element is using for destination adoc, for all another elements – only document number is populating into ‘:adopted-from: …’ (see <https://github.com/metanorma/sts2mn/issues/31>)

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| xxx-meta (nat-, iso-, reg-, std-meta) | > | bibdata[@type="standard"] |  |
| title-wrap/@xml:lang | < > | title/@language |  |
| title-wrap/full | > | title[@format="text/plain" @type="main"] |  |
| title-wrap/full  Note: values, separated by ‘ — ‘ | < | title[\*] |  |
| title-wrap/intro | < > | title[@format="text/plain" @type="title-intro"] | :title-intro-<@xml:lang>: … |
| title-wrap/main | < | title[@format="text/plain" @type="title-intro"]  Note: if title[@type="main"] is empty |  |
| title-wrap/main | < > | title[@format="text/plain" @type="title-main"] | :title-main-<@xml:lang>: … |
| title-wrap/compl  Note: For BSI, in source STS XML there aren’t all title’s components in standalone elements (full, into, main or compl), therefor XSLT tries to separate title by ‘dash’ char (decimal 8212) into component (see ‘mode="bibdata\_title\_full"’ templates). | < > | title[@format="text/plain" @type="title-part"] | :title-part-<@xml:lang>: … |
| title-wrap/compl Note: amendment title puts after ‘—' | < | title[@type=” title-amd”] |  |
| std-ref[@type='dated'] | > | docidentifier[@type="iso"] docidentifier[@type="iso-with-lang"] + (first upper-cased char from doc-ident/language)  docidentifier[@type="iso-reference"] + (first upper-cased char from doc-ident/language) |  |
| std-ref[@type='dated'] | < | docidentifier[1] |  |
| std-ref[@type='undated'] | > | no |  |
| std-ref[@type='undated'] | < | docidentifier[1], before ‘:’ |  |
| doc-ref | > | docidentifier[@type="iso-reference"] |  |
| doc-ref | < | docidentifier[@type='iso-reference'][last()] or docidentifier @type='iso-with-lang'] |  |
| custom-meta-group/ custom-meta[meta-name = 'ISBN']/ meta-value | < > | docidentifier[@type="ISBN"] | :isbn: … |
| custom-meta-group/custom-meta [meta-name != 'ISBN']  Example: <custom-meta>  <meta-name>price-ref-pages</meta-name>  <meta-value>38</meta-value>  </custom-meta>  <custom-meta>  <meta-name>generation-date</meta-name>  <meta-value>2018-10-29 11:07:31</meta-value>  </custom-meta>  Example: <custom-meta-group>  <custom-meta>  <meta-name>Perinorm ID</meta-name>  <meta-value>000000000030387671</meta-value>  </custom-meta>  <custom-meta>  <meta-name>UPI</meta-name>  <meta-value>000000000030387671</meta-value>  </custom-meta>  …  </custom-meta-group>  Note: conversion gap <https://github.com/metanorma/mn-samples-bsi/issues/20> | > | no |  |
| std-ident | < > | *skip processing*  Note: see nested elements processing | = originator doc-number-part-number |
| std-ident/doc-number | < > | docnumber | :docnumber: … |
| pub-date  Example: <pub-date>2019-03-15</pub-date>  Note: Comment from NISO-STS-common-metadata1.ent:  The <pub-date> was used in ISO STS to name  the date this standard was (or was to be)  published. This element has been superseded  by <release-date> but may still appear in  documents for backward compatibility with  ISO STS.  Remarks: While this element may still be  used, for better JATS compatibility, the  date of initial publication should be  recorded using <release-date> with the  appropriate attributes.  Remarks: For ISO STS published standards,  the <pub-date> recorded the date on which  the project reached ISO stage 60, and the  IS was first published. For the initial  publication, the <pub-date> and the  <release-date> were the same. For later ISO  publications (corrections, new editions,  versions in other languages), the <pub-date>  would be earlier than the <release-date> (and  was be the same across all versions of a  given standard). | < > | date[@type="published"]/on | :published-date: … |
| release-date  Example: <release-date>2018-10-31</release-date> | > | date[@type="release"]/on | :date: release … |
| release-date  Note: if output format is STS NISO | < | date[@type="release"]/on |  |
| release-date or empty: <release-date/>  Note: if output format is STS ISO | < | date[@type="release"]/on[last()] |  |
| doc-ident/sdo | < > | contributor[role type="author"]/organization/abbreviation | :mn-document-class: …  or iso |
| std-ident/originator | < > | contributor[role type="publisher"]/ organization/abbreviation |  |
| std-ident/edition | < > | edition | :edition: … |
| std-ident/version  Note: conversion gap <https://github.com/metanorma/mn-samples-bsi/issues/22> | > | version | no |
| std-ident/version  Note: conversion gap <https://github.com/metanorma/mn-samples-bsi/issues/22> | < | version/revision-date |  |
| content-language | < > | language |  |
| doc-ident/language | < | language |  |
| doc-ident/language | > | *skip processing* | :language: … |
| doc-ident/release-version | > | status stage: WD – 20  CD – 30 DIS – 40 FDIS – 50 IS – 60 substage: WD or CD or DIS or FDIS – 00 IS – 60 | WD: :docstage: 20 :docsubstage: 00  CD:  :docstage: 30 :docsubstage: 00  DIS: :docstage: 40 :docsubstage: 00  FDIS:  :docstage: 50 :docsubstage: 00  IS:  :docstage: 60 :docsubstage: 60  doctype=standard:  :docstage: 60 :docsubstage: 60  else:  :docstage: :docsubstage: |
| doc-ident/release-version | < > | status/stage/@abbreviation |  |
| doc-ident/release-version | > | ext/stagename |  |
| std-xref [@type= “isPublishedFormatOf”]  Example: <std-xref type= “isPublishedFormatOf”><std-ref>20/30387670 DC</std-ref></std-xref>  Note: for BSI only. | < > | relation[@type="related"]/bibitem/  title: -- docidentifier: Draft for comment ‘std-ref’  Example: <relation type="related">  <bibitem>  <title>--</title>  <docidentifier>Draft for comment 20/30387670 DC</docidentifier>  </bibitem>  </relation> | :bsi-related: Draft for comment ‘std-ref’ |
| comm-ref  Note: for BSI only. | < > | relation[@type="related"]/bibitem/  title: -- docidentifier: Committee reference ‘text’.  Example: <relation type="related">  <bibitem>  <title>--</title>  <docidentifier>Committee reference DEF/1</docidentifier>  </bibitem>  </relation> | :bsi-related: Committee reference ‘text’ |
| comm-ref  Note: not for BSI.  Example: <comm-ref>ISO/TC 154</comm-ref> | < > | tries to extract ‘TC’, ‘SC’, ‘WG` values and put them into: ext/editorialgroup/technical-committee | subcommittee | workgroup / @number  or  ext/editorialgroup/technical-committee/@number | :technical-committee-type: TC :technical-committee-number: … :subcommittee-type: SC :subcommittee-number: … :workgroup-type: WG :workgroup-number: …  :technical-committee-code: ... :technical-committee-name: ... |
| secretariat  Example: <secretariat>SAC</secretariat> | < > | editorialgroup/secretariat | :secretariat: … |
| std-xref | > | relation |  |
| std-xref/@type  Note: conversion gap  <https://github.com/metanorma/mn-samples-bsi/issues/8> |  | relation/@type  Note: value copied for round-trip conversion, in Metanorma XML (presentation) there aren’t these types. |  |
| std-xref/std-ref  Example: <std-xref type="informativelyReferences">  <std-ref type="dated">BS EN ISO 19011:2018</std-ref></std-xref> | < > | relation/bibitem  Example: <relation type="informativelyReferences">  <bibitem>BS EN ISO 19011:2018</bibitem>  </relation> |  |
| reg-meta  iso-meta std-meta | < > | relation[@type="adopted-from"] | :adopted-from: std-ident/originator std-ident/doc-number - std-ident/ part-number |
| permissions | < > | copyright |  |
| permissions/copyright-year | < > | copyright/from | :copyright-year: … |
| permissions/copyright-holder (if there is space) | < > | copyright/owner/organization/name |  |
| permissions/copyright-holder (if there isn’t space) | < > | copyright/owner/organization/abbreviation |  |
| permissions | < > | boilerplate/copyright-statement/clause |  |
| permissions/copyright-year permissions/copyright-holder | > | boilerplate/copyright-statement/clause / p[@id="boilerplate-year”]© {copyright-holder} {copyright-year} |  |
| permissions/copyright-statement | > | boilerplate/copyright-statement/clause / p[@id="boilerplate-message"] | Note: conversion gap (ADOC->XML)  <https://github.com/metanorma/mn-samples-bsi/issues/9> |
| permissions/copyright-statement | < | boilerplate/copyright-statement//title |  |
| permissions/copyright-statement | < | boilerplate/copyright-statement//p[@id!="boilerplate-year”] |  |
| std-ident/doc-type depends on std-ident/originator  Note: for BSI documents | > | ext/doctype originator starts with ‘BS’ and ‘standard’ -> standard originator starts with ‘PAS’ and ‘publicly available specification’ or ‘standard’ -> publicly-available-specification originator starts with ‘PD’ and ‘published document’ -> published-document else value ‘as-is’ | :doc-type: <see rules in left next one column> |
| std-ident/doc-type | > | ext/doctype  ‘is’ -> ‘international-standard’ ‘r’ -> ‘recommendation’ ‘spec’ -> ‘spec’ else value ‘as-is’ | :doc-type: … |
| std-ident/doc-type | < | ext/doctype  - international-standard -> IS - if BSI document -> capitalized value  - in another cases – value as is |  |
| ics  Example: <ics>01.140.30</ics> | < > | ext/ics/code | :library-ics: … |
| no  Note: commented in mn->sts conversion. | < | ext/ subdoctype |  |
| std-ident/part-number  Note: not for BSI. | < > | ext/structuredidentifier/project-number/@part | :partnumber: … |
| std-ident/originator  std-ident/doc-number  Note: not for BSI. |  | ext/structuredidentifier/project-number/ {originator} {doc-number} |  |
| std-ident/part-number | < > | ext/structuredidentifier/partnumber | :partnumber: … |
| doc-ident/proj-id | < > | ext/structuredidentifier/project-number |  |
| doc-ident/urn  Example: <urn>iso:std:iso-iec:29151:ed-1:v1:en</urn>  Note: conversion gap <https://github.com/metanorma/mn-samples-bsi/issues/21> |  | no |  |
| no |  | status/iteration |  |
| no |  | ext/structuredidentifier/docnumber |  |
| no |  | ext/updates-document-type |  |
| wi-number  Note: conversion gap <https://github.com/metanorma/mn-samples-bsi/issues/12> |  | no |  |
| release-version-id  Note: conversion gap <https://github.com/metanorma/mn-samples-bsi/issues/12> |  | no |  |
| page-count  Note: conversion gap <https://github.com/metanorma/mn-samples-bsi/issues/23> |  | no |  |
| std-ident/suppl-type std-ident/suppl-number std-ident/suppl-version  Note: conversion gap <https://github.com/metanorma/mn-samples-bsi/issues/27> |  | no |  |
| meta-date  meta-date/@type  Example: <meta-date type="DOR">2019-10-27</meta-date>  <meta-date type="DOW">2020-07-31</meta-date>  <meta-date type="DOP">2020-07-31</meta-date>  <meta-date type="DOA">2020-04-30</meta-date>  Example: <meta-date type="vote-start">2018-10-31</meta-date>  <meta-date type="vote-end">2018-12-26</meta-date>  Note: conversion gap <https://github.com/metanorma/mn-samples-bsi/issues/32> |  | no |  |
| permissions/license | < | boilerplate/license-statement |  |
| permissions/license/@xlink:title | < | boilerplate/license-statement/clause[1]/title |  |
| no  see nested elements processing | < | boilerplate/license-statement/clause |  |
| permissions/license/license-p | < | boilerplate/license-statement/clause/p |  |
| permissions/license/license-p/@id  Note: the attribute @id isn’r defined in ISO STS DTD | < | boilerplate/license-statement/clause/p/@id | [[id]] |
| license[@specific-use="legal"] | < | boilerplate/legal-statement |  |
| license[@specific-use="legal"]/@xlink:title | < | boilerplate/legal-statement /clause[1]/title |  |
| license-p | < | boilerplate/legal-statement/clause |  |
| license-p/@id  Note: the attribute @id isn’r defined in ISO STS DTD | < | boilerplate/legal-statement/clause/@id | [[id]] |
| no |  | script  Example: <script current="true">Latn</script> |  |
|  |  |  | fixed: :mn-output-extensions: xml,html :local-cache-only: :data-uri-image: |
|  |  |  | :imagesdir: value imagesdir from command line parameter (default ‘images’ |
|  |  |  | :coverpage-image: images/ coverpage.X.png  Note: only if doc-type = publicly-available-specification |

## Section title

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| sec/label  Example: <label>6.2.2.3</label> | < > | title  Examples:  - if there is ‘title’, then put it before ‘tab’:  <title>6.2.2.3<tab/> … </title>  - if there isn’t title:  <title>10.1</title> and add inline-header=”true” to parent clause: <clause inline-header=”true”… |  |
| sec/title  Example: <title>Prototype design</title> | < > | title (after tab)  Example: <title>…<tab/>Prototype design</title> | === ...  Note: count of === depends of section’s depth. |
| *skip processing* | < | title/@tab |  |
| *skip processing* | < | title/@depth |  |

## Terms

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| sec[@sec-type = 'terms']//title | > |  | [.boilerplate]  == ...  If there isn’t child ‘p’:  {blank}  else: My predefined text  Note: count of == depends on clause depth. |
| term-sec  Example: <term-sec id="sec\_3.1"> | > | *skip processing*  Note: sec ‘sec[term-sec]’ above |  |
|  |  |  |  |
|  |  |  |  |
| term-sec/@id | > | no |  |
| term-sec | < | term |  |
| term-sec/@id Note: id=sec\_ term/name | < | term |  |
| term-sec/label  Example: <label>3.1</label> | < > | term/name  Example: <name>3.1</name> |  |
| tbx:termEntry  Example:  <tbx:termEntry … | < > | term  Example:  <term … | === ...  Note: count of === depends of section’s depth. |
| tbx:termEntry/@id  Example:  <tbx:termEntry id="term\_3.1"> | < > | term/@id  Example: <term id=”term\_3.1”> | [[id]] |
| tbc:termEntry/tbx:langSet | < > | no |  |
| tbc:termEntry/tbx:langSet/@xml:lang  Example: <tbx:langSet xml:lang="en">  Note: in all STS examples xml:lang=”en” | < > | no |  |
| tbx:langSet/tbx:tig  Example: <tbx:tig> | > | *skip processing*  Note: see nested elements processing |  |
| tbx:tig/tbx:term [../tbx:normativeAuthorization/ @value = 'preferredTerm']  Example:  <tbx:tig> <tbx:term>geometrical product specification</tbx:term> … <tbx:normativeAuthorization value="preferredTerm"/> </tbx:tig> | < > | preferred  Example: <preferred>geometrical product specification</preferred> |  |
| tbx:tig/tbx:term  [../tbx:normativeAuthorization/ @value = 'admittedTerm']  Example: <tbx:tig>  <tbx:term>GPS</tbx:term> …  <tbx:normativeAuthorization value="admittedTerm"/>  </tbx:tig> | < > | admitted  Example: <admitted>GPS</admitted> | alt:[...] |
| tbx:tig/tbx:term  [../tbx:normativeAuthorization/ @value = 'deprecatedTerm'] | < > | deprecates | deprecated:[...] |
| tbx:tig/tbx:term (without ../tbx:normativeAuthorization) | < > | preferred |  |
| tbx:tig/tbx:definition  Example: <tbx:definition>aspect of … | > | definition/p  Example: <definition><p>aspect of … |  |
| tbx:tig/tbx:definition | < | definition |  |
| tbx:entailedTerm  Example:  <tbx:entailedTerm target="term\_3.8">objectives (3.8)</tbx:entailedTerm> | < > | <em>term without (x.y)</em> (<xref target=”@target”><strong>x.y if substring in @target after \_term equal to x.y</strong></xref>)  Example:  <em>objectives</em> (<xref target="term\_3.8"><strong>3.8</strong></xref>) | term:[term]  term:[term\_real,term]  if cited and referenced terms are different, for example: term[systems,system] |
| tbx:normativeAuthorization  Example: <tbx:normativeAuthorization value="preferredTerm"/> <tbx:normativeAuthorization value="admittedTerm"/> | > | see above |  |
| tbx:partOfSpeech  Example: <tbx:partOfSpeech value="noun"/>  Note: in all STS examples ‘value="noun"’.  Always creates.in mn->sts conversion. | < > | no |  |
| tbx:example  Example: <tbx:example>Product, service, | < > | termexample  Example: <termexample>Product, service, | [example] |
| tbx:source  Example:  <tbx:source>BS EN ISO 14044:2006+A1:2018, <bold>3.1</bold>, modified</tbx:source> | < > | termsource/origin/@citeas  Example: <termsource>  <origin citeas="BS EN ISO 14044:2006+A1:2018, 3.1, modified"/>  </termsource> | [.source]  <<value1, value2,..>>, modified  Note: value1, value2, - extracted values from text  To do: [https://github.com/metanorma/ mnconvert/issues/7](https://github.com/metanorma/%20mnconvert/issues/7) |
| tbx:source | < | termsource/origin/localityStack |  |
| tbx:source | < | termsource/origin/localityStack/ locality |  |
| tbx:source | < | termsource/origin/localityStack/ locality/@type |  |
| tbx:source | < | termsource/origin/localityStack/ locality/referenceFrom |  |
| tbx:source  Example: , modified — … | < | termsource/modification |  |
|  |  |  |  |
| tbx:note  Example: <tbx:note>For the definition of | > | termnote/p  Example: <termnote><p>For the definition of | NOTE: … |
| tbx:note | < | termnote |  |
| tbx:termType  Example: <tbx:termType value="abbreviation"/>  Note: in all examples ‘value="abbreviation"/>’  Note: conversion gap  <https://github.com/metanorma/mn-samples-bsi/issues/19> | < > | <!--STS: <tbx:termType value="abbreviation"/>-->  Note: commented for round-trip conversion |  |
| Example: <term-sec id="sec\_3.1.1.1">  <label>3.1.1.1</label>  <tbx:termEntry id="term\_3.1.1.1">  <tbx:langSet xml:lang="en">  <tbx:definition>  <tbx:entailedTerm target="term\_3.1.1.2">time (3.1.1.2)</tbx:entailedTerm> on the <tbx:entailedTerm target="term\_3.1.1.18">calendar (3.1.1.18)</tbx:entailedTerm>  </tbx:definition>  <tbx:note>Common forms of date include <tbx:entailedTerm target="term\_3.1.2.7">calendar date (3.1.2.7)</tbx:entailedTerm>, <tbx:entailedTerm target="term\_3.1.2.8">ordinal date (3.1.2.8)</tbx:entailedTerm> or <tbx:entailedTerm target="term\_3.1.2.9">week date (3.1.2.9)</tbx:entailedTerm>.</tbx:note>  <tbx:tig id="term\_3.1.1.1-1">  <tbx:term>date</tbx:term>  <tbx:partOfSpeech value="noun"/>  </tbx:tig>  </tbx:langSet>  </tbx:termEntry>  </term-sec> |  | Example:  <term id="term\_3.1.1.1">  <name>3.1.1.1</name>  <preferred>date</preferred>  <definition>  <p>  <em>time</em> (<xref target="term\_3.1.1.2"> <strong>3.1.1.2</strong></xref>) on the <em>calendar</em> (<xref target="term\_3.1.1.18">  <strong>3.1.1.18</strong></xref>) <em>time scale</em> (<xref target="term\_3.1.1.5"> <strong>3.1.1.5</strong></xref>)  </p>  </definition>  <termnote>  <p>Common forms of date include  <em>calendar date</em> (<xref target="term\_3.1.2.7"> <strong>3.1.2.7</strong></xref>),  <em>ordinal date</em> (<xref target="term\_3.1.2.8"> <strong>3.1.2.8</strong></xref>) or  <em>week date</em> (<xref target="term\_3.1.2.9"> <strong>3.1.2.9</strong></xref>).  </p>  </termnote>  </term> |  |

## Text formatting elements

Note for NISO STS->Metanorma ADOC conversion: is some cases **unconstrained formatting pair used, see** <https://docs.asciidoctor.org/asciidoc/latest/text/#unconstrained>

### Paragraph

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| p | > | p |  |
| p | < | p  Note: don’t create in output, if parent element is ‘termexample’, ‘definition’, ‘termnote’, ‘modification’ or ‘dd’ |  |
| p/@id | > | p/@id | [[id]] |
| p/@id  Note: if not BSI. (BSI doesn’t use @id for paragraphs) | < | p/@id |  |
| p/@style-type  Example: <p style-type=”indent”>… | < > | p/@style-type  Example: <p style-type=”indent”>…  Note: in Metanorma XML (presentation) model there isn’t the attribute @style-type. It’s added for round-trip conversion. |  |
| p/@specific-use  Example: <p specific-use="indent">…  Note: conversion gap  <https://github.com/metanorma/mn-samples-bsi/issues/14> | < > | p/@specific-use  Example: <p specific-use="indent">…  Note: in Metanorma XML (presentation) model there isn’t the attribute @specific-use. It’s added for round-trip conversion. |  |
| no | < | p/@align |  |

### Boldface text

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| bold  Example: <bold>core</bold> | < > | strong  Example: <strong>core</strong> | \*...\*  or  \*\*...\*\*  Example: \*core\* |

Note for NISO STS->Metanorma XML (presentation) conversion: in case of ‘bold’ inside ‘std’, then ‘bold’ will be moved outside std.

NISO STS: <std><bold><std-ref>ISO 10667 series</std-ref></bold></std>

Metanorma: <strong><eref citeas="ISO 10667 series" type="inline">ISO 10667 series</eref></strong>

### Italic text

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| italic  Example: <italic>security</italic> | < > | em  Example: <em>security</em> | \_...\_  or  \_\_...\_\_ |

Note for NISO STS->Metanorma XML (presentation) conversion: in case of ‘italic` inside ‘std’, then ‘italic` will be moved outside std.

NISO STS: <std><italic><std-ref>BS 5839‑1:2013</std-ref></italic></std>

Metanorma: <em><eref citeas="BS 5839‑1:2013" type="inline">BS 5839‑1:2013</eref></em>

### Underlined text

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| underline  Example: <underline>top</underline> | < > | underline  Example: <underline>top</underline> | [underline]#...# |

### Subscript formatting

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| sub  Example: <sub>t</sub> | < > | sub  Example: <sub>t</sub> | ~...~  or  ~~...~~ |

### Superscript formatting

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| sup  Example: <sup>a</sup> | < > | sup  Example: <sup>a</sup> | ^...^  or ^^...^^ |
| skip  Note: nested elements will be processed | < | sup[stem] |  |

Note for NISO STS->Metanorma XML (presentation) conversion: `sup` formatting will be ignored if it contains `xref[@ref-type='fn']` inside.

NISO STS: <sup><xref ref-type="fn" rid="fn\_1"><sup>1)</sup></xref></sup>

Metanorma: <fn reference="1"><p><eref citeas="BS 8889" type="inline" bibitemid="BS\_8889">BS 8889</eref> was in development at the time of publication.</p></fn>

### Monospaced text

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| monospace  Example: <monospace>.tbx.</monospace> | < > | tt  Example: <tt>.tbx.</tt> | `...`  or  ``...`` |

### Small caps text

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| sc  Example: P<sc>ublishers</sc> | < > | smallcap  Example: P<smallcap>ublishers</smallcap> | [smallcap]#...# |

### Line break

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| break  Example: <break/> | < > | br  Example: <br/> | + |

### Source code, pre-formatted text

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| code  Example: <code language="ruby">puts "Hello, world."  Note: in ISO STS DTD this element isn’t defined | < > | sourcecode  Example: <sourcecode lang="ruby">puts "Hello, world." | [source, *language]* --  ...  -- |
| code/@language | < > | sourcecode/@lang | [*source*, language] |
| preformat  Example: <preformat>&lt;?xml version="1.0" encoding="UTF-8"?&gt; | < > | sourcecode  Example: <sourcecode>&lt;?xml version="1.0" encoding="UTF-8"?&gt; | [source]  --  ...  -- |
| preformat/@preformat-type  Note: for ISO STS only | < | sourcecode/@lang |  |

### Note

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| non-normative-note  Example:  <non-normative-note> <label>NOTE 1</label> <p>Continuous improvement | < > | note  Example: <note>  <name>NOTE 1</name>  <p>Continuous improvement | NOTE:  …  a few nested elements: [NOTE] ==== … ====  in a list item: + -- NOTE: … -- |
| non-normative-note/label  Example: <non-normative-note><label>NOTE 1</label>… | < > | note/name  Example:  <note><name>NOTE 1</name>… |  |
| non-normative-note/label NOTE … | < | note[not(name)] |  |
| non-normative-note/p | < > | note/p |  |
| non-normative-note | < | admonition  Note: for more information, see <https://github.com/metanorma/mn2sts/issues/8> |  |
| non-normative-note/@id | < | admonition/@id |  |
| non-normative-note/label | < | admonition/@type  (@type=”danger”, "caution", “warning”, “important”, “safty\_precautions”)  Note: uppercased |  |

### Example

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| non-normative-example  Example:  <non-normative-example><label>EXAMPLE</label><p>The … | < > | example  Example: <example><p>The … | [example] |
| non-normative-example  /label  Example: <non-normative-example ><label> EXAMPLE </label>… | > | no |  |
| non-normative-example  /label: EXAMPLE … | < | example/name |  |
| element-citation |  | no |  |

### Link

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| ext-link | < > | link[text()] |  |
| ext-link/text() | < > | link/text() |  |
| ext-link/@xlink:type  Example: xlink:type="simple"  Note: conversion gap, fixed value in mn->sts conversion  <https://github.com/metanorma/mn-samples-bsi/issues/24> | < > | no |  |
| ext-link/@ext-link-type  Example: <ext-link ext-link-type="uri" xlink:href="http://www.iso.org/directives"> www.iso.org/directives</ext-link> | > | no |  |
| ext-link/@xlink:href  Example: <ext-link xlink:type="simple" xlink:href="http://bsigroup.com/standards"> bsigroup.com/standards</ext-link> | < > | link/@target  Example: <link target="http://bsigroup.com/standards"> bsigroup.com/standards</link> | @xlink:href[@xlink:href]  Note: for BSI documents, non-hyphen minus (U+2011) replaces by minus. |
| uri  Example: <uri>http://www.iso.org/obp</uri> | < > | link[no text()]/@target  Example: <link target="http://www.iso.org/obp"/> | uri[uri] |

### Reference

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| xref[@ref-type != 'fn’ and @ref-type != table-fn’] | < > | xref | if @ref-type = 'other': <..>  if @ref-type = 'sec' and reference to ‘term-sec’: term:[term name from term-sec]  else: <<@rid>> |
| xref[@ref-type != 'fn’ and @ref-type != table-fn’]/@rid | < > | xref/@target |  |
| xref/@ref-type | > | no |  |
| xref/@ref-type  Note: evaluate by xslt, depends on ‘referenced to’ element (fig, table, app, fn, bibr or sec(default)) | < | no |  |
| named-content | > | xref | term:[  IF @content-type = term and points to term-sec or termEntry, then put term  if text different than term, then put also text before term  ELSE put xlink:href and text  ] |
| named-content | < | xref[not contains digits or Annex] |  |
| named-content/@content-type | > | xref/@content-type  Note: in Metanorma XML (presentation) model there isn’t the attribute @content-type. It’s adding for round-trip conversion. |  |
| named-content/@content-type  - abbrev, if @target starts-with ‘abbrev’  - term | < | xref[not contains digits or Annex]/@content-type |  |
| named-content/@xlink:href  Note: if empty, then get text content of named-content | > | xref/@target |  |
| named-content/@xlink:href | < | xref[not contains digits or Annex]/@target |  |
| named-content[@content-type="ace-tag"]  Example:  <named-content content-type="ace-tag" specific-use="A1\_start"/>  Note: conversion gap https://github.com/metanorma/mn-samples-bsi/issues/31 | > | no | add:[] |
| std | < > | eref | <<id,locality,name>>  where  - id – id of ref item in bibliography, found by @std-id  - locality, name – parsed values from @std-id  Example:  xml: <std std-id="iso:std:iso:guide:73:ed-1:en:clause:3.5.1.3" type="dated"><std-ref>ISO Guide 73:2009</std-ref>, 3.5.1.3</std>  Adoc: <<ref\_3,clause=3.5.1.3>> |
| std/std-ref | > | eref/@citeas |  |
| std/std-ref | < | eref/@citeas  Note:  preliminary processing - replace double ‘--' to 8212(decimal) dash  remove ‘[‘ and ‘]’ around  Note: for BSI documents, replaces space to non-break space (A0(hex)) |  |
| std/@std-id  Note: conversion gap in @std-id format <https://github.com/metanorma/mn-samples-bsi/issues/37> (PAS\_2035\_2019 vs. PAS 2035:2019) | > | eref/@bibitemid |  |
| std/@std-id | < | eref/@bibitemid  Note:  instead of @bibitemid tries to get bibliography//bibitem[@id = @bibitemid]/urn if exist |  |
| std | < | eref/localityStack  Note: get text from nested locality elements |  |
| , modified — | < | eref/localityStack/locality/ modification |  |
| std/@std-id points to non-existing elements with @id=@std-id  Example: <std std-id="BS 0:2011" type="dated">  <std-ref>BS 0:2011</std-ref>  </std>  there isn't any entity with id="BS 0:2011".  Note: conversion gap  <https://github.com/metanorma/mn-samples-bsi/issues/41> |  | xref  (adoc->mn conversion) | Example: <<BS\_0\_2011,BS 0:2011>> |
| std/@type  Examples: type="dated"  type="undated" | > | no |  |
| std/@type  Examples: type="dated"  type="undated" | < | eref/@citeas  Note: determines programmatically, see template setDatedUndatedType in the module mn2sts.xsl |  |
| std/std-ref/processing-instruction(‘doi’)  Example: <std std-id="iso:std:iso:44001:en" type="undated">  <std-ref>ISO 44001<?doi https://doi.org/10.3403/30353016U?></std-ref>  </std>  Note: conversion gap <https://github.com/metanorma/mn-samples-bsi/issues/28> | > | no |  |
| xref[@ref-type="other"] | < | callout |  |
| xref[@ref-type="other"]/@rid | < | callout/@target |  |

### Formula

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| disp-formula | < > | formula/stem | stem:[...] |
| mml:\* | < > | \* xmlns=”http://www.w3.org/1998/Math/MathML” | as is in mathml |
| mml:math/@id | > | formula/@id | [[id]] |
| disp-formula/@id | < | formula/@id |  |
| Example: <disp-formula>  <mml:math id=”mml\_m1”>… |  | Example: <formula id=”mml\_m1”>  <stem type=”MathML”> <math xmlns=http://www.w3.org/1998/Math/MathML”>… |  |
| inline-formula  Example: <inline-formula> <mml:math id=”mml\_m1”><mml:mrow> |  | stem  Example: <stem type=”MathML”>  <math xmlns=”http://www.w3.org/1998/Math/MathML”>  <mrow>  NO id | stem:[...] |

### Quote

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| disp-quote | < > | quote | [quote, *related-object*]  \_\_\_\_  ... \_\_\_\_ |
| disp-quote | < | quote/@id |  |
| disp-quote/related-object | < > | quote/source | [*quote*, related-object] |
| disp-quote/related-object/ text value | < | quote/source/@citeas |  |
| no | < | quote/source/@type |  |
| no | < | quote/source/@bibitemid |  |
| disp-quote/related-object | < | quote/author |  |
| disp-quote/p | < > | quote/p | *...* |
| Example:  <disp-quote>  <p>This International Standard …</p>  <related-object>ISO, ISO 7301:2011, Clause 1</related-object>  </disp-quote> | > | Example:  <quote> <source>ISO, ISO 7301:2011, Clause 1</source> <p> This International Standard …</p> </quote> |  |
| *skip processing*  Note: see nested elements processing | < | quote/localityStack |  |
| *skip processing*  Note: see nested elements processing | < | quote/localityStack/locality |  |
| disp-quote/related-object/  Clause Annex text value  (delimited by ‘;’) | < | quote/localityStack/locality/@type=’clause’  quote/localityStack/locality/@type= annex quote/localityStack/locality/@type |  |
| disp-quote/related-object/ text value | < | quote/localityStack/locality/ referenceFrom |  |
| Example:  <disp-quote id=”...">  <p id… ">This International Standard ...</p>  <related-object>ISO, ISO 7301:2011, Clause 1</related-object>  </disp-quote> | < | Example:  <quote id="…">  <source type="inline" bibitemid="ISO7301" citeas="ISO 7301:2011"><localityStack><locality type="clause"> <referenceFrom>1</referenceFrom></locality> </localityStack></source>  <author>ISO</author>  <p id…">This International Standard ...</p>  </quote> |  |

### Footnote

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| fn | < > | fn | footnote:[...] |
| fn/@id | < > | no |  |
| fn/label | < > | no |  |
| xref[@ref-type = “fn”][following-sibling::\*[fn][@id = xref/@rid]]/text() | < > | fn/@reference |  |
| Example:  <xref ref-type="fn" rid="fn\_1"><sup>1</sup>  </xref> <fn id="fn\_1">  <label>1</label><p>The Fire…</p></fn> |  | Example: <fn reference="1"><p>The Fire …</p></fn> |  |
| table-wrap-foot/fn  Note: moving up to the text | > | fn | footnote:[...] |
| back/fn-group Note: moving up to the text  (possible conversion gap <https://github.com/metanorma/mn-samples-bsi/issues/18>) | > | fn | footnote:[...] |
| xref[@ref-type=“fn”] | < > | fn | footnote:[...] |
| xref[@ref-type=“fn”]/text() Note: removing ‘)’ | < > | fn/@reference |  |
| If xref[@ref-type=“fn”] in table-wrap: table-wrap//fn[@id = xref/@rid]/\* | < > | fn/\* |  |
| else: //fn-group/fn[@id = xrefrid]/\* | < > | fn/\* |  |
| xref[@ref-type=”table-fn”] | < > | fn |  |
| xref[@ref-type=“table-fn”]/text() Note: removing ‘)’ | < > | fn/@reference |  |
| table-wrap//fn[@id = xref/@rid]/\* | < > | fn/\* |  |
| Example: <thead>…  <xref ref-type="table-fn" rid="table-fn\_1.1"> <sup>A)</sup> </xref>… </thead>  …  <table-wrap-foot>  <fn id="table-fn\_1.1">  <label><sup>A)</sup></label>  <p>These items shall appear in every design brief.</p>  </fn>  </table-wrap-foot>  </table-wrap> |  | Example: <thead> … <fn reference="A"><p>These items shall appear in every design brief.</p></fn> … </thead> |  |
| Example: <list-item>  … <xref ref-type="fn" rid="fn\_1"> <sup>1)</sup> </xref> … </body> … <fn-group>  <fn id="fn\_1">  <label>1)</label>  <p>Marking…</p>  </fn-group>  </back> |  | Example:  <fn reference="1"><p>Marking…</p></fn> |  |

### Amendment

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| no  see nested elements processing | < | amend |  |
| editing-instruction  Note: this rule for NISO STS only. | < | amend/description |  |
| editing-instruction/@id  Note: this rule for NISO STS only. | < | amend/@id |  |
| editing-instruction/@content-type  Note: this rule for NISO STS only. | < | amend/@change |  |
| editing-instruction/p  Note: this rule for NISO STS only. | < | amend/description/p |  |
| p | < | amend/description/p |  |
| p/@content-type  Note: this rule for ISO STS only. | < | amend/@change |  |
| no  see nested elements processing | < | amend/newcontent |  |
| p | < | amend/newcontent/p |  |
| label  Note: adding suffix to label for element @type | < | amend/autonumber/@type |  |
|  |  |  |  |

### Annotation

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| element-citation  Note: this rule for ISO STS only | < | annotation |  |
| element-citation/@id  Note: this rule for ISO STS only | < | annotation/@id |  |
| element-citation/annotation  Note: this rule for NISO STS only | < | annotation |  |
| element-citation/ annotation/@id  Note: this rule for NISO STS only | < | annotation/@id |  |
| element-citation/ annotation/@id | > |  | <text from node with @rid = annotation/@id> |

### Other

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| styled-content  Examples: <styled-content style-type="normal">(This annex forms an integral part of this Recommendation | International Standard.)</styled-content>  <styled-content style="color:#009fe3;">•</styled-content> | < > | <!--STS: <styled-content …>-->….<!--STS: </styled-content>-->  Note: tags commented for round-trip conversion.  Examples: <!--STS: <styled-content style-type="normal">-->(This annex forms an integral part of this Recommendation | International Standard.)<!--STS: </styled-content>-->  <!--STS: <styled-content style="color:#009fe3;">-->•<!--STS: </styled-content>--> | no |
| boxed-text  Example: <boxed-text>  <p>Objective: To provide management direction… </p> </boxed-text>  Note: conversion gap  <https://github.com/metanorma/mn-samples-bsi/issues/38> |  | no |  |
| th/break th/strong | < | th//br/strong |  |

## Tables

Note: In Metanorma XML (presentation) there aren’t elements for NISO/ISO STS XML elements ‘table-wrap’, ‘table-wrap/label’, ‘table-wrap/caption’, therefore they transform to data inside Metanorma XML (presentation) element ‘table’.

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| table-wrap  Example: <table-wrap … | < > | *skip processing*  Note: see nested elements processing |  |
| table-wrap/@id  Example: <table-wrap id="tab\_D.1" … | < > | table/@id  Example: <table id="tab\_D.1" … | [[id]] |
| table-wrap/@content-type  Example: content-type="ace-table"  Note: conversion gap <https://github.com/metanorma/mn-samples-bsi/issues/39> | > | no | options=”unnumbered” |
| table-wrap/@position  Example: position="float"  Note: conversion gap  <https://github.com/metanorma/mn-samples-bsi/issues/35> |  | no |  |
| table-wrap/@orientation  Example: orientation="landscape" | < > | <pagebreak orientation="…"/>  …  <pagebreak orientation=”portrait"/>  Example:  <pagebreak orientation="landscape"/>  …  <pagebreak orientation="portrait"/> | before table: [%...] <<<   after table: [%portrait] <<< |
| table-wrap/label  Example: <label>Table D.1</label> | < > | table/name  Example: <name>Table D.1 — …</name> |  |
| table-wrap/caption  Example: <caption> |  | *skip processing*  Note: see nested elements processing |  |
| table-wrap/caption/title  Example: <title>Table of requirements</title> | < > | table/name  Example: <name>…Table of requirements</name> | . … |
| table  Example: <table … | < > | table  Example: <table … | if parent is array: [[array\_ + array/@id]] [%unnumbered]  […]  |===  …  |=== |
| array/table | < | figure//table |  |
| table/@id  Example: <table id=”calendar” | < > | table/@id  Example: <table id=”calendar”  Note: if there isn’t source STS table/@id, then it populates from STS table-wrap/@id or array/@id | [[id]] |
| table/@width  Examples: <table width="650">  <table width="80%">  <table width="500px"> | < > | table/@width  Examples: <table width="650px">  <table width="80%">  <table width="500px">  Note: if in source STS table/@width there isn’t ‘px’ or ‘%’, then ‘px’ will be added. | width=...  Note: if value doesn’t contain ‘%’ or ‘px’, then ‘px’ will be added. |
| table-wrap/table-wrap-foot  Example: </table>  <table-wrap-foot>  <p>  <non-normative-note>  <label>NOTE</label>  <p>Actual sensitivity might vary by activity or partner.</p>  </non-normative-note>  </p>  </table-wrap-foot>  </table-wrap> | < > | </tbody> content of table-wrap-foot </table>  Example: </tbody>  <p>  <note>  <name>NOTE</name>  <p>Actual sensitivity might vary by activity or partner.</p>  </note>  </p>  </table> | options=”footer” |
| table/@style  Example: <table style="border-collapse:collapse;border-color:#009fe3;"  Note: conversion gap, <https://github.com/metanorma/mn-samples-bsi/issues/11> | < > | table/@style  Example:  <table style="border-collapse:collapse;border-color:#009fe3;"  Note: in Metanorma XML (presentation) model there isn’t the attribute @style. It adds for round-trip conversion. | no |
| table/@frame  Example: <table frame="hsides"  Note: conversion gap, <https://github.com/metanorma/mn-samples-bsi/issues/11> | < > | table/@frame  Example: <table frame="hsides"  Note: in Metanorma XML (presentation) model there isn’t the attribute @frame. It added for round-trip conversion. | no |
| table/@rules  Example:  <table rules="rows"  Note: conversion gap, <https://github.com/metanorma/mn-samples-bsi/issues/11> | < > | table/@rules  Example:  <table rules="rows"  Note: in Metanorma XML (presentation) model there isn’t the attribute @rules. It added for round-trip conversion. | no |
| thead  Example: <thead> | < > | thead  Example: <thead> | options=”header”, headerrows=count(tr) |
| tbody  Example: <tbody> | < > | tbody  Example: <tbody> |  |
| tfoot | < | tfoot |  |
| colgroup  Example: <colgroup>  Note: conversion gap <https://github.com/metanorma/mn-samples-bsi/issues/26> | < > | colgroup  Example: <colgroup>  Note: In some STS XML documents there isn’t ‘colgroup’ element, but in Metanorma XML (presentation) it adds always. |  |
| col  Example: <col …>  Note: conversion gap <https://github.com/metanorma/mn-samples-bsi/issues/26> | < > | col  Example: <col …> |  |
| col/@width  Example: <col width="11%"/> | < > | col/@width  Example: <col width="11%"/> | [cols=”...”]  Note: comma separated widths  Note: removes %, cm, and multiply by 100 |
| tr  Example: <tr> | < > | tr  Example: <tr> |  |
| th,td  Example: <th … | < > | th,td  Example: <th … | | |
| th,td/@align  Example: <th align="center" | < > | th,td/@align  Example: <th align=" center " | ^ > |
| th,td/@valign  Example: <th valign="middle" | < > | th,td/@valign  Example: <th valign="middle" | .^ .> |
| th,td/@style  Example: <th style="background-color:#009fe3;"  Note: there is conversion gap <https://github.com/metanorma/mn-samples-bsi/issues/11> | < > | th,td /@style  Example: <th style="background-color:#009fe3;"  Note: in Metanorma XML (presentation) model there isn’t the attribute @style. It added for round-trip conversion. |  |
| th,td/@rowspan  Example: <th rowspan="6" | < > | th,td/@rowspan  Example: <th rowspan="6" | .x+ |
| th,td/@colspan  Example: <th colspan="3" | < > | th,td/@colspan  Example: <th colspan="3" | x+ |
| th,td/@scope  Example: <th scope="row" | < > | th,td/@scope  Example: <th scope="row"  Note: in Metanorma XML (presentation) model there isn’t the attribute @scope. It added for round-trip conversion. |  |

## List

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| list[@list-type = 'bullet' or @list-type = 'simple'] | < > | ul |  |
| list[@list-type = 'bullet'] | < | ul[not(@list-type)] |  |
| list[not (@list-type = 'bullet' or @list-type = 'simple')] | > | ol |  |
| list[@list-type = ' alpha-lower'] | < | ol[not(@list-type)] |  |
| list/@list-type=’alpha-lower’ | < > | ol/@type=”alphabet”] |  |
| list/@list-type=’alpha-upper’ | < > | ol/@type=”alphabet\_upper” | [upperalpha] |
| list/@list-type=’roman-lower | < > | ol/@type=”roman” | [lowerroman] |
| list/@list-type=’roman-upper’ | < > | ol/@type=”roman\_upper” | [upperroman] |
| list/@list-type=’ arabic’ | > | ol/@type=”arabic” | [arabic] |
| list/ first char of list/label[1] in 1234567890] | > | ol/@type=”arabic” | [arabic] |
| list/@list-type=’order’ | < | ol/@type=”arabic” |  |
| list/ first char of list/label[1] in ixvcm] | > | @type=”roman” | [lowerroman] |
| list first char of list/label[1] in IXVCM] | > | @type=”roman\_upper” | [upperroman] |
| list/ first char of list/label[1] in a-z] | > | @type=”alphabet” |  |
| first char of list/label[1] in A-Z | > | @type=”alphabet\_upper” | [upperalpha] |
| list/@list-type | < > | @type |  |
| list/@list-type="simple"  Note: conversion gap <https://github.com/metanorma/mn-samples-bsi/issues/40> |  | no | no |
| first char of list/label[1] != 1 or a or A or i or I | > | @start  decimal number for start value determines in the Java function org.metanorma.utils.Util.getListStartValue | no To do: [https://github.com/metanorma/ mnconvert/issues/24](https://github.com/metanorma/mnconvert/issues/24) |
| list-item | < > | li | if list/@list-type = 'bullet' ‘dash’ or simple: \* ...  else . ...  Note: count of ‘\*’ and ‘,’ depends on level depth |
| list-item/label | > | no |  |
| list-item/label  Note: Evaluate in xslt type of list (ul or ol), @type and @start values.  See <xsl:template match="\*[local-name() = 'li']"> for rules. | < | Evaluate by ul,ol/@type, @start |  |

## Definitions list

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| def-list  Example: <def-list … | > | dl  Example: <dl> |  |
| def-list/title | > | no | \*...\* |
| def-list/def-item  Example: <def-item … | > | *skip processing*  Note: see nested elements processing |  |
| def-item/term  Example: <term>CBRMS</term> | > | dt  Example: <dt> CBRMS</dt> | ...:: |
| def-list/def-item/id  Example: <def-item id="abbrev\_1"> | > | dt/@id  Example: <dt id=”abbrev\_1”>… | [[id]] |
| def-item/def  Example:  <def> <p>intellectual property right</p> </def> | > | dd  Example: <dd> <p>intellectual property right</p> </dd> | text |
| array [count of table/col =2 or count of table/colgroup/col = 2]  Example: <array id="tab\_a">  <table>  <tbody>  <tr>  <td valign="top" scope="row" align="left">cent</td>  <td valign="top" align="left">time scale component century  </td>  </tr>  </tr>  <tr>  <td align="left" valign="top">  <italic>C</italic><sub>dev</sub>  </td>  <td align="left" valign="top">is the cost of developing the product (for total anticipated quantity);</td>  </tr>….  Note: this rule is not applicable for BSI documents | < > | dl  @key = ‘true’ if STS `label` = 'Key'  Example:  <dl id=”tab\_a">  <dt>cent</dt>  <dd>  <p id="…">time scale component century</p>  </dd> | tr/td[1]::tr/td[2] |
| array [count of table/col =2 or count of table/colgroup/col = 2]/@id | < > | dl/@id |  |
| array [count of table/col =2 or count of table/colgroup/col = 2]/label = ‘Key’ | > | dl/@key=”true” | \*label\* |

## Image

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| fig | > | figure | ====  ...  ==== |
| fig/@id | > | figure/@id | [[id]] |
| fig  Note: for BSI only | < | figure[figure] |  |
| fig/@id  Note: for BSI only | < | figure[figure]/@id |  |
| fig-group | < | figure[figure] |  |
| fig-group/@id | < | figure[figure]/@id |  |
| fig-group/@ content-type=figures | < | figure[figure] |  |
| fig/@fig-type  Example: <fig fig-type="figure" |  | no |  |
| fig/label  Example: <label>Figure 1</label> | < > | figure/name  Example: <name>Figure 1 — … | if starts with 0: [number=…] |
| fig/caption  Example: <caption><title>Multidimensional relationships</title></caption> | < > | figure/name  Example: <name>… Multidimensional relationships</name> | . … |
| fig-group | > | figure | ====  ...  ==== |
| fig-group/@id | > | figure/@id | [[id]] |
| graphic[caption]  Example:  <graphic xlink:type="simple" xlink:href="30238432\_FigA01a">  <label>a)</label><caption><title>Imposed loads for treads with a span of up to 1 200 mm</title></caption>  </graphic> | < > | figure | ====  ...  ==== |
| graphic/label  (if there isn’t graphic/caption/title) | < > | figure/name | . … |
| graphic/label  Example: <label>a)</label><caption>… | < > | figure/name  Example: <name>a) &#160; … </name) |  |
| graphic/caption/title  Example:  <label>a)</label><caption><title>Imposed loads for treads with a span of up to 1 200 mm</title></caption> | < > | figure/name  Example: <name>… Imposed loads for treads with a span of up to 1 200 mm | . … |
| graphic | < > | image height=”auto” width=”auto” | image:: … |
| inline-graphic | > | image height=”auto” width=”auto” | image:: … |
| no |  | image/@height  Example: height=”auto” |  |
| no |  | image/@width  Example: width=”auto” |  |
| inline-graphic  | graphic/@xlink:href and not processing-instruction('isoimg-id')  Note: if there isn’t processing-instruction('isoimg-id'), then @xlink:href contains image file name. | > | image/@src string in base64 encoding OR imagedir path + @xlink:href + .png(optional, if @xlink:href doesn’t have file extension) | image:: … |
| graphic/processing-instruction('isoimg-id')  Note: if there is this PI, then it contains image file name (but not @xlink:href)  Note: graphic/@xlink:href conversion gap  <https://github.com/metanorma/mn-samples-bsi/issues/25> | > | image/@src imagedir path + value from ‘isoimg-id` | image:: … |
| graphic/@xlink:href | < | image/@id |  |
| graphic/alt-text  inline-graphic/alt-text | < > | image/@alt | image:: … [alt-text] |
| no |  | image/@mimetype |  |
| graphic/object-id[@pub-id-type = 'publisher-id']  Example:  <object-id pub-id-type="fpi">+//ISO 9070/RA::A00007::GE::DIN German Institute for  Standardization::Standards//NONSGML ISO 13849-1:2006-11-00::Figure 1//EN</object-id>  <object-id pub-id-type="publisher-id">fig-1</object-id> | > | no | image:: … [object-id] |

## Bibliography

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| ref | < > | bibitem | \* [[[id,(ref)label]]], title  or  \* [[[id,ref]]], title  Note: see below for component’s source data. |
| ref/@id | < > | bibitem/@id | \* [[[**id**,(ref)label]]] |
| ref/std/@std-id | > |  | \* [[[**id**,(ref)label]]] |
| ref/std/@std-ref | > |  | \* [[[**id**,(ref)label]]] |
| ref/std/@std-ref | > |  | \* [[[id,(**ref**)label]]] |
| ref/std//title | > |  | \* [[[id,(ref)label]]], \_**title\_** |
| ref/@content-type | < > | bibitem/@type |  |
| ref/label | < > | docidentifier[@type="metanorma"] | \* [[[id,(ref)**label**]]] |
| mixed-citation  Note: there is a conversion gap for sub-elements, see <https://github.com/metanorma/mn-samples-bsi/issues/10> | < > | bibitem/title |  |
| mixed-citation/std | > |  | \* [[[id,(**ref**)label]]] |
| mixed-citation/@publication-type  Note: conversion gap <https://github.com/metanorma/mn-samples-bsi/issues/30> | > | no |  |
| ref-list/ref/std/@std-id | < > | bibitem/docidentifier[@type="URN"] |  |
| no |  | bibitem/fetched |  |
| no |  | bibitem/title/@type |  |
| no |  | bibitem/title/@format |  |
| no |  | bibitem/title/@language |  |
| no |  | bibitem/title/script |  |
| no |  | bibitem/uri |  |
| no |  | bibitem/uri/@type |  |
| no  Note: commented | < | bibitem/docnumber |  |
| no  Note: commented | < | bibitem/date |  |
| no  Note: commented | < | bibitem/date/@type |  |
| no  Note: commented | < | bibitem/date/on |  |
| no  Note: commented | < | bibitem/contributor |  |
| no  Note: commented | < | bibitem/contributor/role |  |
| no  Note: commented | < | bibitem/contributor/role/@type |  |
| no  Note: commented | < | bibitem/contributor/organization |  |
| no  Note: commented | < | bibitem/contributor/organization/name |  |
| no  Note: commented | < | bibitem/contributor/organization/abbreviation |  |
| no  Note: commented | < | bibitem/contributor/organization/uri |  |
| no |  | bibitem/edition |  |
| no |  | bibitem/language |  |
| no |  | bibitem/script |  |
| no |  | bibitem/abstract |  |
| no |  | bibitem/abstract/@format |  |
| no |  | bibitem/abstract/@language |  |
| no |  | bibitem/abstract/@script |  |
| no |  | bibitem/status |  |
| no |  | bibitem/status/stage |  |
| no |  | bibitem/status/substage |  |
| no |  | bibitem/copyright |  |
| no |  | bibitem/copyright/from |  |
| no |  | bibitem/copyright/owner |  |
| no |  | bibitem/copyright/owner/organization |  |
| no |  | bibitem/copyright/owner/organization/name |  |
| no |  | bibitem/relation |  |
| no |  | bibitem/relation/@type |  |
| no |  | bibitem/relation/bibitem /… |  |
| no |  | bibitem/place |  |
| ref/std/xref  ref/std/fn | < | bibitem/note |  |
| no |  | bibitem/note/@format |  |
| ref/title | < | bibitem/formattedref |  |
| no |  | bibitem/formattedref/@format |  |
| see rules for text formatting elements |  | bibitem/formattedref/ formatted text |  |

## Processing instructions

|  |  |  |  |
| --- | --- | --- | --- |
| **ISO/NISO STS** |  | **Metanorma XML (presentation)** | **Metanorma ADOC** |
| <?foreward …  Example: <?foreward metadata-error?> | > | no, ignores |  |
| <?doi …  Example: <?doi https://doi.org/10.3403/30130607?> | < > | copy ‘as is’, for round-trip conversion |  |
| <?QAout …  <?QA …  Example: <?QAout enhancements="En-2 En-3 En-4"?> <?QA valid="true" features="E-1 E-3 M-13 M-15 F-6 En-1" release-version="v2.1"?> | < > | copy ‘as is’, for round-trip conversion |  |
| <?... Page\_Break?>  <?... Page-Break?>  Examples: <?Para Page\_Break?>  <?Table Page\_Break?>  <?Table Page-Break?> | > | <pagebreak /> | <<<  Note: conversion gap for page break between table’s rows: <https://github.com/metanorma/mn-samples-bsi/issues/47> |
| <?Para Page\_Break?> | < | <pagebreak />  Note: for pagebreak before table, see rule for  <pagebreak orientation="…"/> |  |
| <?isoimg-id …  Example: <?isoimg-id 44001\_ed1fig1.eps?> | > | image/@src .eps, .EPS extensions replacing by .png  Example: <image src=”images/44001\_ed1fig1.png”… | image::... |