BS ISO 21720:2024

BSI Standards Publication

XLIFF (XML Localization Interchange File Format)

Documents
Document Sharing Hub

Full document details: BS ISO 21720:2024

National foreword

This British Standard is the UK implementation of ISO 21720:2024. It supersedes BS ISO 21720:2017, which is withdrawn.

The UK participation in its preparation was entrusted to Technical Committee TS/1, Terminology.

A list of organizations represented on this committee can be obtained on request to its committee manager.

Contractual and legal considerations

This publication has been prepared in good faith, however no representation, warranty, assurance or undertaking (express or implied) is or will be made, and no responsibility or liability is or will be accepted by BSI in relation to the adequacy, accuracy, completeness or reasonableness of this publication. All and any such responsibility and liability is expressly disclaimed to the full extent permitted by the law.

This publication is provided as is, and is to be used at the recipient's own risk.

The recipient is advised to consider seeking professional guidance with respect to its use of this publication.

This publication is not intended to constitute a contract. Users are responsible for its correct application.

© The British Standards Institution 2024
Published by BSI Standards Limited 2024

ISBN 978 0 539 27586 5

ICS 35.240.30

Compliance with a British Standard cannot confer immunity from legal obligations.

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 31 July 2024.

Amendments/corrigenda issued since publication

Date	Text affected

Documents
Document Sharing Hub

Full document details: BS ISO 21720:2024

International Standard

ISO 21720

XLIFF (XML Localization Interchange File Format)

XLIFF (Format de fichier XML pour l'échange de données de localisation)

Second edition
2024-07

Reference number
ISO 21720:2024(en)

© ISO 2024

Documents
Document Sharing Hub

Full document details: BS ISO 21720:2024

COPYRIGHT PROTECTED DOCUMENT

© ISO 2024

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO document should be noted (see www.iso.org/directives).

ISO draws attention to the possibility that the implementation of this document may involve the use of (a) patent(s). ISO takes no position concerning the evidence, validity or applicability of any claimed patent rights in respect thereof. As of the date of publication of this document, ISO had not received notice of (a) patent(s) which may be required to implement this document. However, implementers are cautioned that this may not represent the latest information, which may be obtained from the patent database available at www.iso.org/patents. ISO shall not be held responsible for identifying any or all such patent rights.

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html.

This document was prepared by OASIS (as XLIFF Version 2.1, February 2018) and drafted in accordance with its editorial rules. It was assigned to Technical Committee ISO/TC 37, *Language and terminology*, Subcommittee SC 5, *Translation, interpreting and related technology*, and adopted under the "fast-track procedure".

This second edition cancels and replaces the first edition (ISO 21720:2017), which has been technically revised.

The main changes are as follows (see also Appendix C):

— Two major features are being added in XLIFF Version 2.1:

  — Advanced Validation methods;

  — Native Support for ITS 2.0.

— The Change Tracking Module was demoted to an extension to free hands of the TC and other implementers while working on a new version of the Change Tracking Module for XLIFF 2.2.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

— A major bug fix was performed on the core xsd. The core xsd now enforces the xs:language data type on the srcLang and trgLang attributes. It was critical to make this fix, because -- as per OASIS policy -- validation artifacts would prevail over the prose provisions that are correct in both XLIFF 2.1 and XLIFF 2.0.

— Also an erroneously omitted Constraint of the xml:lang attribute on the <source> element has been added/restored in the normative text.

— Apart from the five (5) major changes mentioned above, numerous editorial bugfixes were made to secure greater clarity, either by fixing example errors or omissions, or by reorganizing normative content, so that the intent becomes clear and unequivocal at some troublesome places highlighted by XLIFF 2.0 implementers.

— Importantly, the TC decided to drop informative listings of the validation artifacts that had bloated the spec extent unnecessarily, were hard to keep in sync with the actual normative artifacts, while their actual usability proved rather limited -- readers who were able to read schema languages would not actually read them as printed listings and would anyways refer to the actual validation artifacts that are now referenced more prominently.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

Table of Contents

1 Introduction
- 1.0 IPR Policy
- 1.1 Terminology
- 1.1.1 Key words
- 1.1.2 Definitions
- 1.1.3 Key concepts
- 1.2 Normative References
- 1.3 Non-Normative References

2 Conformance

3 Fragment Identification
- 3.1 Selectors for Core Elements
- 3.2 Selectors for Modules and Extensions
- 3.3 Relative References
- 3.4 Examples

4 The Core Specification
- 4.1 General Processing Requirements
- 4.2 Elements
- 4.2.1 Tree Structure
- 4.2.2 Structural Elements
- 4.2.3 Inline Elements
- 4.3 Attributes
- 4.3.1 XLIFF Attributes
- 4.3.2 XML namespace
- 4.4 CDATA sections
- 4.5 XML Comments
- 4.6 XML Processing Instructions
- 4.7 Inline Content
- 4.7.1 Text
- 4.7.2 Inline Codes
- 4.7.3 Annotations
- 4.7.4 Sub-Flows
- 4.7.5 White Spaces
- 4.7.6 Bidirectional Text
- 4.7.7 Target Content Modification
- 4.7.8 Content Comparison
- 4.8 Segmentation
- 4.8.1 Segments Representation
- 4.8.2 Segments Order
- 4.8.3 Segmentation Modification
- 4.8.4 Best Practice for *Mergers* (Informative)
- 4.9 Extension Mechanisms
- 4.9.1 Extension Points
- 4.9.2 Constraints
- 4.9.3 Processing Requirements

5 The Modules Specifications
- 5.1 Translation Candidates Module
- 5.1.1 Introduction
- 5.1.2 Module Namespace and Validation Artifacts
- 5.1.3 Module Fragment Identification Prefix
- 5.1.4 XLIFF Core Reuse
- 5.1.5 Translation Candidate Annotation
- 5.1.6 Module Elements
- 5.1.7 Module Attributes

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

5.1.8 Example
5.2 Glossary Module
5.2.1 Introduction
5.2.2 Module Namespace and Validation Artifacts
5.2.3 Module Fragment Identification Prefix
5.2.4 Module Elements
5.2.5 Module Attributes
5.2.6 Example
5.3 Format Style Module
5.3.1 Introduction
5.3.2 Module Namespace and Validation Artifacts
5.3.3 Module Fragment Identification Prefix
5.3.4 Module Specification
5.3.5 Module Attributes
5.4 Metadata Module
5.4.1 Introduction
5.4.2 Module Namespace and Validation Artifacts
5.4.3 Module Fragment Identification Prefix
5.4.4 Module Elements
5.4.5 Module Attributes
5.5 Resource Data Module
5.5.1 Introduction
5.5.2 Module Namespace and Validation Artifacts
5.5.3 Module Fragment Identification Prefix
5.5.4 Module Elements
5.5.5 Module Attributes
5.5.6 Examples
5.6 Change Tracking Extension (Informative)
5.6.1 Introduction
5.6.2 Module Namespace and Validation Artifacts
5.6.3 Module Fragment Identification Prefix
5.6.4 Module Elements
5.6.5 Module Attributes
5.6.6 Example
5.7 Size and Length Restriction Module
5.7.1 Introduction
5.7.2 Module Namespace and Validation Artifacts
5.7.3 Module Fragment Identification Prefix
5.7.4 Module Elements
5.7.5 Module Attributes
5.7.6 Standard profiles
5.7.7 Third party profiles
5.7.8 Conformance
5.7.9 Example
5.8 Validation Module
5.8.1 Introduction
5.8.2 Module Namespace and Validation Artifacts
5.8.3 Module Fragment Identification Prefix
5.8.4 Module Elements
5.8.5 Module Attributes
5.9 ITS Module
5.9.1 Introduction
5.9.2 Module Namespace and Validation Artifacts
5.9.3 Module Fragment Identification Prefix
5.9.4 Conformance to the ITS Module Specification

Documents
Document Sharing Hub

© ISO 2024 – All rights reserved

Full document details: BS ISO 21720:2024

5.9.5 ITS Tools Referencing
5.9.6 ITS data categories defined in the ITS Module
5.9.7 ITS data categories that have a partial overlap with XLIFF features
5.9.8 ITS data categories available through XLIFF Core and other Modules
5.9.9 ITS data categories not represented in XLIFF
5.9.10 ITS Mapping Annotations
5.9.11 Module Elements
5.9.12 Module Attributes
5.9.13 Example file

Appendixes

A Media Type Registration Template for XLIFF Version 2.0 and higher Versions
A.1 Registration Template
A.1.1 Detailed Security Considerations
A.1.1.1 Privacy, trust and integrity
A.1.1.2 Core
A.1.1.3 Resource Data Module
A.1.1.4 ITS Module
A.1.1.5 Other potentially security sensitive constructs
B Machine Readable Validation Artifacts (Informative)
B.1 XML Schemas Tree
B.2 Support Schemas
C Specification Change Tracking (Informative)
C.1 High Level Summary of Changes made in Comparison to XLIFF Version 2.0
C.2 Tracking of changes made in response to Public Reviews
C.2.1 Tracking of changes in response to the Public Review of the Candidate OASIS Standard 01
C.2.2 Tracking of changes in response to the 4th Public Review
C.2.3 Tracking of changes in response to the 3rd Public Review
C.2.4 Tracking of changes in response to the 2nd Public Review
C.2.5 Tracking of changes in response to the 1st Public Review
D Acknowledgements (Informative)

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

Documents
Document Sharing Hub

1 Introduction

XLIFF is the *XML Localization Interchange File Format* designed by a group of multilingual content publishers, software providers, localization service providers, localization tools providers, and researchers. It is intended to give any multilingual content owner a single interchange file format that can be understood by any localization provider, using any conformant localization tool. While the primary focus is on being a lossless interchange format, usage of XLIFF as a processing format is neither encouraged nor discouraged or prohibited.

All text is normative unless otherwise labeled. The following common methods are used for labeling portions of this specification as informative and hence non-normative:

Appendices and sections marked as "(Informative)" or "Non-Normative" in title,

Notes (sections with the "Note" title),

Warnings (sections with the "Warning" title),

Examples (mainly example code listings, tree diagrams, but also any inline examples or illustrative exemplary lists in otherwise normative text),

Schema and other validation artifacts listings (the corresponding artifacts are normative, not their listings).

1.0 IPR Policy

This specification is provided under the RF on RAND Terms Mode of the OASIS IPR Policy, the mode chosen when the Technical Committee was established. 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 Technical Committee web page (https://www.oasis-open.org/committees/xliff/ipr.php).

1.1 Terminology

1.1.1 Key words

The key words MUST, MUST NOT, REQUIRED, SHALL, SHALL NOT, SHOULD, SHOULD NOT, RECOMMENDED, MAY, and OPTIONAL are to be interpreted as described in [**RFC 2119**].

1.1.2 Definitions

Agent

any application or tool that generates (creates), reads, edits, writes, processes, stores, renders or otherwise handles *XLIFF Documents*.

Agent is the most general application conformance target that subsumes all other specialized user agents disregarding whether they are defined in this specification or not.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

Enrich, Enriching

the process of associating module and extension based metadata and resources with the *Extracted* XLIFF payload

Processing Requirements

- *Enriching* MAY happen at the time of *Extraction*.

Note

Extractor knowledge of the native format is not assumed while *Enriching*.

Enricher, Enricher Agent

any *Agent* that performs the *Enriching* process

Extract, Extraction

the process of encoding localizable content from a native content or User Interface format as XLIFF payload, so that localizable parts of the content in the source language are available for *Translation* into the target language along with the necessary context information

Extractor, Extractor Agent

any *Agent* that performs the *Extraction* process

Merge, Merging

the process of importing XLIFF payload back to the originating native format, based on the *full knowledge* of the *Extraction* mechanism, so that the localized content or User Interface strings replace the source language in the native format

Merger, Merger Agent

an *Agent* that performs the *Merge* process

Warning

Unless specified otherwise, any *Merger* is deemed to have the same knowledge of the native format as the *Extractor* throughout the specification.

Mergers independent of *Extractors* can succeed, but it is out of scope of this specification to specify interoperability for *Merging* back without the full *Extractor* knowledge of the native format.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

Modify, Modification

the process of changing core and module XLIFF structural and inline elements that were previously created by other *Writers*

Processing Requirements

- XLIFF elements MAY be *Modified* and *Enriched* at the same time.

  Note

  Extractor or *Enricher* knowledge of the native format is not assumed while *Modifying*.

Modifier, Modifier Agent

an *Agent* that performs the *Modification* process

Translation, Translate

a rendering of the meaning of the source text, expressed in the target language

Writer, Writer Agent

an *Agent* that creates, generates, or otherwise writes an *XLIFF Document* for whatever purpose, including but not limited to *Extractor*, *Modifier*, and *Enricher Agents*.

Note

Since XLIFF is intended as an exchange format rather than a processing format, many applications will need to generate *XLIFF Documents* from their internal processing formats, even in cases when they are processing *XLIFF Documents* created by another *Extractor*.

1.1.3 Key concepts

XLIFF Core

The core of XLIFF 2.1 consists of the minimum set of XML elements and attributes required to (a) prepare a document that contains text extracted from one or more files for localization, (b) allow it to be completed with the translation of the extracted text, and (c) allow the generation of *Translated* versions of the original document.

The XML namespace that corresponds to the core subset of XLIFF 2.1 is `"urn:oasis:names:tc:xliff:document:2.0"`.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

XLIFF-defined (elements and attributes)

The following is the list of allowed schema URI prefixes for *XLIFF-defined* elements and attributes:

```
urn:oasis:names:tc:xliff:
http://www.w3.org/2005/11/its
```

However, the following namespaces are NOT considered *XLIFF-defined* for the purposes of the XLIFF 2.1 specification:

```
urn:oasis:names:tc:xliff:document:1.0
urn:oasis:names:tc:xliff:document:1.1
urn:oasis:names:tc:xliff:document:1.2
urn:oasis:names:tc:xliff:changetracking:2.0
```

Elements and attributes from other namespaces are not *XLIFF-defined*.

XLIFF Document

Any XML document that declares the namespace `"urn:oasis:names:tc:xliff:document:2.0"` as its main namespace, has `<xliff>` as the root element and complies with the XML Schemas and the declared Constraints that are part of this specification.

XLIFF Module

A module is an OPTIONAL set of XML elements and attributes that stores information about a process applied to an *XLIFF Document* and the data incorporated into the document as result of that process.

Each official module defined for XLIFF 2.1 has its grammar defined in an independent XML Schema with a separate namespace.

1.2 Normative References

[**BCP 47**] M. Davis, *Tags for Identifying Languages*, http://tools.ietf.org/html/bcp47 IETF (Internet Engineering Task Force).

[**HTML5**] Ian Hickos, Robin Berjon, Steve Faulkner, Travis Leithead, Erika Doyle Navara, Edward O'Connor, Silvia Pfeiffer *HTML5. A vocabulary and associated APIs for HTML and XHTML*, http://www.w3.org/TR/html5/ W3C Recommendation 28 October 2014.

[**ITS**] David Filip, Shaun McCance, Dave Lewis, Christian Lieske, Arle Lommel, Jirka Kosek, Felix Sasaki, Yves Savourel *Internationalization Tag Set (ITS) Version 2.0*, http://www.w3.org/TR/its20/ W3C Recommendation 29 October 2013.

[**NOTE-datetime**] M. Wolf, C. Wicksteed, *Date and Time Formats*, http://www.w3.org/TR/NOTE-datetime W3C Note, 15th September 1997.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

[**NVDL**] International Standards Organization, *ISO/IEC 19757-4, Information Technology - Document Schema Definition Languages (DSDL) - Part 4: Namespace-based Validation Dispatching Language (NVDL)*, http://standards.iso.org/ittf/PubliclyAvailableStandards/c038615_ISO_IEC_19757-4_2006(E).zip ISO, June 1, 2006.

[**RFC 2119**] S. Bradner, *Key words for use in RFCs to Indicate Requirement Levels*, https://www.ietf.org/rfc/rfc2119.txt IETF (Internet Engineering Task Force) RFC 2119, March 1997.

[**RFC 3987**] M. Duerst and M. Suignard, *Internationalized Resource Identifiers (IRIs)*, https://www.ietf.org/rfc/rfc3987.txt IETF (Internet Engineering Task Force) RFC 3987, January 2005.

[**RFC 7303**] H. Thompson and C. Lilley, *XML Media Types*, https://www.tools.ietf.org/html/rfc7303 IETF (Internet Engineering Task Force) RFC 7303, July 2014.

[**Schematron**] International Standards Organization, *ISO/IEC 19757-3, Information Technology - Document Schema Definition Languages (DSDL) - Part 3: Rule-Based Validation — Schematron (Second Edition)*, http://standards.iso.org/ittf/PubliclyAvailableStandards/c055982_ISO_IEC_19757-3_2016.zip ISO, January 15, 2016.

[**UAX #9**] M. Davis, A. Lanin, A. Glass, *UNICODE BIDIRECTIONAL ALGORITHM*, http://www.unicode.org/reports/tr9/tr9-35.html Unicode Bidirectional Algorithm, May 18, 2016.

[**UAX #15**] M. Davis, K. Whistler, *UNICODE NORMALIZATION FORMS*, http://www.unicode.org/reports/tr15/tr15-44.html Unicode Normalization Forms, February 24, 2016.

[**Unicode**] The Unicode Consortium, *The Unicode Standard*, http://www.unicode.org/versions/Unicode9.0.0/ Mountain View, CA: The Unicode Consortium, June 21, 2016.

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

[**XML namespace**] W3C, *Schema document for namespace http://www.w3.org/XML/1998/namespace* http://www.w3.org/2001/xml.xsd [http://www.w3.org/2009/01/xml.xsd]. at http://docs.oasis-open.org/xliff/xliff-core/v2.1/os/schemas/informativeCopiesOf3rdPartySchemas/w3c/xml.xsd in this distribution

[**XML Catalogs**] Norman Walsh, *XML Catalogs*, https://www.oasis-open.org/committees/download.php/14809/xml-catalogs.html OASIS Standard V1.1, 07 October 2005.

[**XML Schema**] W3C, *XML Schema*, *refers to the two part standard comprising [XML Schema Structures] and [XML Schema Datatypes]* (Second Editions) W3C Recommendations 28 October 2004.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

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

[**XML Schema Structures**] W3C, *XML Schema Part 1: Structures*, https://www.w3.org/TR/xmlschema-1/ (Second Edition) W3C Recommendation 28 October 2004.

1.3 Non-Normative References

[**LDML**] *Unicode Locale Data Markup Language* http://unicode.org/reports/tr35/

[**SRX**] *Segmentation Rules eXchange* http://www.unicode.org/uli/pas/srx/

[**UAX #29**] M. Davis, *UNICODE TEXT SEGMENTATION*, http://www.unicode.org/reports/tr29/ Unicode text Segmentation.

[**XML I18N BP**] *Best Practices for XML Internationalization*, 13 February 2008, http://www.w3.org/TR/xml-i18n-bp/ W3C Working Group.

2 Conformance

1. *Document Conformance*
    a. XLIFF is an XML vocabulary, therefore conformant *XLIFF Documents* MUST be well formed and valid [XML] documents.
    b. Conformant *XLIFF Documents* MUST be valid instances of the official Core XML Schema (http://docs.oasis-open.org/xliff/xliff-core/v2.1/os/schemas/xliff_core_2.0.xsd) that is a part of this multipart Work Product.
    c. As not all aspects of the XLIFF specification can be expressed in terms of XML Schemas, conformant *XLIFF Documents* MUST also comply with all relevant elements and attributes definitions, normative usage descriptions, and Constraints specified in this specification document.
    d. *XLIFF Documents* MAY contain custom extensions, as defined in the Extension Mechanisms section.
2. *Application Conformance*
    a. XLIFF *Writers* MUST create conformant *XLIFF Documents* to be considered XLIFF compliant.
    b. *Agents* processing conformant *XLIFF Documents* that contain custom extensions are not REQUIRED to understand and process non-XLIFF elements or attributes. However, conformant applications SHOULD preserve existing custom extensions when processing conformant *XLIFF Documents*, provided that the elements that contain custom extensions are not removed according to XLIFF Processing Requirements or the extension's own processing requirements.
    c. All *Agents* MUST comply with Processing Requirements for otherwise unspecified *Agents* or without a specifically set target *Agent*.
    d. Specialized *Agents* defined in this specification - this is *Extractor*, *Merger*, *Writer*, *Modifier*, and *Enricher Agents* - MUST comply with the Processing Requirements targeting their specifically defined type of *Agent* on top of Processing Requirements targeting all *Agents* as per point c. above.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

   e. XLIFF is a format explicitly designed for exchanging data among various *Agents*. Thus, a conformant XLIFF application MUST be able to accept *XLIFF Documents* it had written after those *XLIFF Documents* were *Modified* or *Enriched* by a different application, provided that:
      i. The processed files are conformant *XLIFF Documents*,
      ii. in a state compliant with all relevant Processing Requirements.
3. *Backwards Compatibility*
   a. Conformant applications are REQUIRED to support XLIFF 2.0.
   b. Conformant applications are NOT REQUIRED to support XLIFF 1.2 or previous Versions.

Note

XLIFF Documents conformant to this specification are not and cannot be conformant to XLIFF 1.2 or earlier versions. If an application needs to support for whatever business reason both XLIFF 2 and XLIFF 1.2 or earlier, these will need to be supported as separate functionalities.

3 Fragment Identification

Because *XLIFF Documents* do not follow the usual behavior of XML documents when it comes to element identifiers, this specification defines how *Agents* MUST interpret the fragment identifiers in IRIs pointing to *XLIFF Documents*.

Note

Note that some identifiers may change during the localization process. For example `<data>` elements may be re-grouped or not depending on how tools treat identical original data.

Constraints

- A fragment identifier MUST match the following format:

```
<expression>      ::= "#" ["/"] <selector>
                      {<selectorSeparator> <selector>}
<selector>        ::= [<prefix> <prefixSeparator>] <id>
<prefix>          ::= NMTOKEN
<id>              ::= NMTOKEN
<prefixSeparator> ::= "="
<selectorSeparator>  ::= "/"
```

- There MUST NOT be two identical prefixes in the expression.
- When used, the following selectors MUST be declared in this order: file selector, group selector and unit selector.
- The selectors for modules or extensions, `<note>`, `<segment>` or `<ignorable>` or source inline elements, target inline elements and `<data>` have the following constraints:
  - Only one of them MAY be used in the expression.
  - The one used MUST be the last selector of the expression.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

Warning

Please note that due to the above Constraints, referencing fragments using third party namespaces within *Modules* or extensions (including but not limited to *XLIFF Core* or the Metadata Module) is not possible. This is to restrict the complexity of the fragment identification mechanism, as it would otherwise have potentially unlimited depth.

3.1 Selectors for Core Elements

- The prefix `f` indicates a `<file>` id and the value of that id is unique among all `<file>` `id` attribute values within the enclosing `<xliff>` element.
- The prefix `g` indicates a `<group>` id and the value of that id is unique among all `<group>` `id` attribute values within the enclosing `<file>` element.
- The prefix `u` indicates a `<unit>` id and the value of that id is unique among all `<unit>` `id` attribute values within the enclosing `<file>` element.
- The prefix `n` indicates a `<note>` id and the value of that id is unique among all `<note>` `id` attribute values within the immediate enclosing `<file>`, `<group>`, or `<unit>` element.
- The prefix `d` indicates a `<data>` id and the value of that id is unique among all `<data>` `id` attribute values within the enclosing `<unit>` element.
- The prefix `t` indicates an id for an inline element in the `<target>` element and the value of that id is unique within the enclosing `<unit>` element (with the exception of the matching inline elements in the `<source>`).
- No prefix indicates an id for a `<segment>` or an `<ignorable>` or an inline element in the `<source>` element and the value of that id is unique within the enclosing `<unit>` element (with the exception of the matching inline elements in the `<target>`).

3.2 Selectors for Modules and Extensions

A selector for a module or an extension uses a registered prefix and the value of that id is unique within the immediate enclosing `<file>`, `<group>` or `<unit>` element.

Constraints

- The prefix of a module or an extension MUST be an NMTOKEN longer than 1 character and MUST be defined in the module or extension specification.
- The prefix of a module or an extension MUST be registered with the XLIFF TC.
- A given module or extension namespace URI MUST be associated with a single prefix.
- A prefix MAY be associated with more than one namespace URI (to allow for example different versions of a given module or extension to use the same prefix).

See also the constraints related to how IDs need to be specified in extensions (which applies for modules as well).

3.3 Relative References

Fragment identifiers that do not start with a character `/` (U+002F) are relative to their location in the document, or to the document being processed.

Documents
Document Sharing Hub

© ISO 2024 – All rights reserved

Full document details: BS ISO 21720:2024

Any unit, group or file selector missing to resolve the relative reference is obtained from the immediate enclosing unit, group or file elements.

3.4 Examples

Given the following XLIFF document:

```
<xliff xmlns="urn:oasis:names:tc:xliff:document:2.0" version="2.0"
    srcLang="en" trgLang="fr">
  <file id="f1">
    <notes>
      <note id="n1">note for file.</note>
    </notes>
    <unit id="u1">
      <my:elem xmlns:my="myNamespaceURI" id="x1">data</my:elem>
      <notes>
        <note id="n1">note for unit</note>
      </notes>
      <segment id="s1">
        <source><pc id="1">Hello <mrk id="m1"
type="term">World</mrk>!</pc>
            </source>
        <target><pc id="1">Bonjour le <mrk id="m1"
type="term">Monde</mrk>
            ! </pc></target>
      </segment>
    </unit>
  </file>
</xliff>
```

You can have the following fragment identifiers:

- `#f=f1/u=u1/1` refers to the element `<pc id="1">` of the source content of the element `<unit id="u1">`.
- `#f=f1/u=u1/t=1` refers to the element `<pc id="1">` of the target content of the element `<unit id="u1">`.
- `#f=f1/n=n1` refers to the element `<note id="n1">` of the element `<file id="f1">`.
- `#f=f1/u=u1/n=n1` refers to the element `<note id="n1">` of the element `<unit id="u1">`.
- `#f=f1/u=u1/s1` refers to the element `<segment id="s1">` of the element `<unit id="u1">`.
- Assuming the extension defined by the namespace URI `myNamespaceURI` has registered the prefix `myprefix`, the expression `#f=f1/u=u1/myprefix=x1` refers to the element `<my:element id="x1">` of the element `<unit id="u1">`.

4 The Core Specification

XLIFF is a bilingual document format designed for containing text that needs *Translation*, its corresponding translations and auxiliary data that makes the *Translation* process possible.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

At creation time, an *XLIFF Document* MAY contain only text in the source language. Translations expressed in the target language MAY be added at a later time.

The root element of an *XLIFF Document* is `<xliff>`. It contains a collection of `<file>` elements. Typically, each `<file>` element contains a set of `<unit>` elements that contain the text to be translated in the `<source>` child of one or more `<segment>` elements. Translations are stored in the `<target>` child of each `<segment>` element.

4.1 General Processing Requirements

- An *Agent* processing a valid *XLIFF Document* that contains *XLIFF-defined* elements and attributes that it cannot handle MUST preserve those elements and attributes.
- An *Agent* processing a valid *XLIFF Document* that contains custom elements and attributes that it cannot handle SHOULD preserve those elements and attributes.

4.2 Elements

This section contains a description of all elements used in *XLIFF Core*.

4.2.1 Tree Structure

Legend:

1 = one
+ = one or more
? = zero or one
* = zero or more

```
<xliff>
|
+---<file> +
  |
  +---<skeleton> ?
  | |
  | +---<other> *
  |
  +---<other> *
  |
  +---<notes> ?
  | |
  | +---<note> +
  |
  +---At least one of (<unit> OR <group>)
      | |
      | +---<unit>
      |   |
      |   +---<other> *
      |   |
      |   +---<notes> ?
      |   | |
      |   | +---<note> +
      |   |
      |   +---<originalData> ?
      |   | |
```

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

```
|    | +---<data> +
|    |
|    +---At least one of (<segment> OR <ignorable>)
|        | |
|        | +---<segment>
|        |    |
|        |    +---<source> 1
|        |    |
|        |    +---<target> ?
|        |
|        +---<ignorable>
|            |
|            +---<source> 1
|            |
|            +---<target> ?
|
+---<group>
    |
    +---<other> *
    |
    +---<notes> ?
    | |
    | +---<note> +
    |
    +---At least one of (<unit> OR <group>)
```

4.2.2 Structural Elements

The structural elements used in *XLIFF Core* are: `<xliff>`, `<file>`, `<skeleton>`, `<group>`, `<unit>`, `<segment>`, `<ignorable>`, `<notes>`, `<note>`, `<originalData>`, `<data>`, `<source>` and `<target>`.

4.2.2.1 xliff

Root element for XLIFF documents.

Contains:

- One or more `<file>` elements

Attributes:

- `version`, REQUIRED
- `srcLang`, REQUIRED
- `trgLang`, OPTIONAL
- `xml:space`, OPTIONAL
- attributes from other namespaces, OPTIONAL

Constraints

- The `trgLang` attribute is REQUIRED if and only if the *XLIFF Document* contains `<target>` elements that are children of `<segment>` or `<ignorable>`.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

- The following *XLIFF Module* attributes are explicitly allowed by the wildcard `other`:

  - the `its:version`attribute from the namespace `http://www.w3.org/2005/11/its`, OPTIONAL.

4.2.2.2 file

Container for localization material extracted from an entire single document, or another high level self contained logical node in a content structure that cannot be described in the terms of documents.

> **Note**
>
> Sub-document artifacts such as particular sheets, pages, chapters and similar are better mapped onto the `<group>` element. The `<file>` element is intended for the highest logical level. For instance a collection of papers would map to a single *XLIFF Document*, each paper will be represented with one `<file>` element, whereas chapters and subsections will map onto nested `<group>` elements.

Contains:

- Zero or one `<skeleton>` element followed by
- elements from other namespaces, OPTIONAL
- Zero or one `<notes>` element followed by
- One or more `<unit>` or `<group>` elements in any order.

Attributes:

- `id`, REQUIRED
- `canResegment`, OPTIONAL
- `original`, OPTIONAL
- `translate`, OPTIONAL
- `srcDir`, OPTIONAL
- `trgDir`, OPTIONAL
- `xml:space`, OPTIONAL
- attributes from other namespaces, OPTIONAL

Constraints

- The following *XLIFF Module* elements are explicitly allowed by the wildcard `other`:

  - Zero or one `<mda:metadata>` elements
  - Zero or one `<res:resourceData>` element
  - Zero or one `<slr:profiles>` elements
  - Zero or one `<slr:data>` elements

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

- Zero or one `<val:validation>` elements
- Zero, one, or more `<its:provenanceRecords>` elements

- *Module* and Extension elements MAY be used in any order.
- The following *XLIFF Module* attributes are explicitly allowed by the wildcard `other`:

  - attributes from the namespace `urn:oasis:names:tc:xliff:fs:2.0`, OPTIONAL, provided that the Constraints specified in the Format Style Module are met.
  - attributes from the namespace `urn:oasis:names:tc:xliff:sizerestriction:2.0`, OPTIONAL, provided that the Constraints specified in the Size and Length Restriction Module are met.
  - attributes from the namespace `http://www.w3.org/2005/11/its`, OPTIONAL, provided that the Constraints specified in the ITS Module are met.
  - attributes from the namespace `urn:oasis:names:tc:xliff:itsm:2.1`, OPTIONAL, provided that the Constraints specified in the ITS Module are met.

4.2.2.3 skeleton

Container for non-translatable material pertaining to the parent `<file>` element.

Contains:

Either

- Non-translatable text
- elements from other namespaces

or

- is empty.

Attributes:

- `href`, OPTIONAL

Constraints

- The attribute `href` is REQUIRED if and only if the `<skeleton>` element is empty.

Processing Requirements

- *Modifiers* and *Enrichers* processing an *XLIFF Document* that contains a `<skeleton>` element MUST NOT change that element, its attributes, or its content.
- *Extractors* creating an *XLIFF Document* with a `<skeleton>` element MUST leave the `<skeleton>` element empty if and only if they specify the attribute `href`.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

4.2.2.4 group

Provides a way to organize units into a structured hierarchy.

Note that this is especially useful for mirroring a source format's hierarchical structure.

Contains:

- elements from other namespaces, OPTIONAL
- Zero or one `<notes>` element followed by
- Zero, one or more `<unit>` or `<group>` elements in any order.

Attributes:

- `id`, REQUIRED
- `name`, OPTIONAL
- `canResegment`, OPTIONAL
- `translate`, OPTIONAL
- `srcDir`, OPTIONAL
- `trgDir`, OPTIONAL
- `type`, OPTIONAL
- `xml:space`, OPTIONAL
- attributes from other namespaces, OPTIONAL

Constraints

- The following *XLIFF Module* elements are explicitly allowed by the wildcard `other`:

  - Zero or one `<mda:metadata>` elements
  - Zero or one `<slr:data>` elements
  - Zero or one `<val:validation>` elements
  - Zero, one, or more `<its:provenanceRecords>` elements

- *Module* and Extension elements MAY be used in any order.
- The following *XLIFF Module* attributes are explicitly allowed by the wildcard `other`:

  - attributes from the namespace `urn:oasis:names:tc:xliff:fs:2.0`, OPTIONAL, provided that the Constraints specified in the Format Style Module are met.
  - attributes from the namespace `urn:oasis:names:tc:xliff:sizerestriction:2.0`, OPTIONAL, provided that the Constraints specified in the Size and Length Restriction Module are met.
  - attributes from the namespace `http://www.w3.org/2005/11/its`, OPTIONAL, provided that the Constraints specified in the ITS Module are met.
  - attributes from the namespace `urn:oasis:names:tc:xliff:itsm:2.1`, OPTIONAL, provided that the Constraints specified in the ITS Module are met.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

4.2.2.5 unit

Static container for a dynamic structure of elements holding the extracted translatable source text, aligned with the *Translated* text.

Contains:

- elements from other namespaces, OPTIONAL
- Zero or one `<notes>` elements followed by
- Zero or one `<originalData>` element followed by
- One or more `<segment>` or `<ignorable>` elements in any order.

Attributes:

- `id`, REQUIRED
- `name`, OPTIONAL
- `canResegment`, OPTIONAL
- `translate`, OPTIONAL
- `srcDir`, OPTIONAL
- `trgDir`, OPTIONAL
- `xml:space`, OPTIONAL
- `type`, OPTIONAL
- attributes from other namespaces, OPTIONAL

Constraints

- A `<unit>` MUST contain at least one `<segment>` element.
- The following *XLIFF Module* elements are explicitly allowed by the wildcard `other`:

  - Zero or one `<mtc:matches>` elements
  - Zero or one `<gls:glossary>` elements
  - Zero or one `<mda:metadata>` elements
  - Zero or one `<res:resourceData>` elements
  - Zero or one `<slr:data>` elements
  - Zero or one `<val:validation>` elements
  - Zero, one, or more `<its:locQualityIssues>` elements
  - Zero, one, or more `<its:provenanceRecords>` elements

- *Module* and Extension elements MAY be used in any order.
- The following *XLIFF Module* attributes are explicitly allowed by the wildcard `other`:

  - attributes from the namespace `urn:oasis:names:tc:xliff:fs:2.0`, OPTIONAL, provided that the Constraints specified in the Format Style Module are met.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

- attributes from the namespace `urn:oasis:names:tc:xliff:sizerestriction:2.0`, OPTIONAL, provided that the Constraints specified in the Size and Length Restriction Module are met.
- attributes from the namespace `http://www.w3.org/2005/11/its`, OPTIONAL, provided that the Constraints specified in the ITS Module are met.
- attributes from the namespace `urn:oasis:names:tc:xliff:itsm:2.1`, OPTIONAL, provided that the Constraints specified in the ITS Module are met.

4.2.2.6 segment

This element is a container to hold in its aligned pair of children elements the minimum portion of translatable source text and its *Translation* in the given Segmentation.

Contains:

- One `<source>` element followed by
- Zero or one `<target>` element

Attributes:

- `id`, OPTIONAL
- `canResegment`, OPTIONAL
- `state`, OPTIONAL
- `subState`, OPTIONAL

4.2.2.7 ignorable

Part of the extracted content that is not included in a segment (and therefore not translatable). For example tools can use `<ignorable>` to store the white space and/or codes that are between two segments.

Contains:

- One `<source>` element followed by
- Zero or one `<target>` element

Attributes:

- `id`, OPTIONAL

4.2.2.8 notes

Collection of comments.

Contains:

- One or more `<note>` elements

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

4.2.2.9 note

This is an XLIFF specific way how to present end user readable comments and annotations. A note can contain information about `<source>`, `<target>`, `<unit>`, `<group>`, or `<file>` elements.

Contains:

- Text

Attributes:

- `id`, OPTIONAL
- `appliesTo`, OPTIONAL
- `category`, OPTIONAL
- `priority`, OPTIONAL
- attributes from other namespaces, OPTIONAL

Constraints

- The following *XLIFF Module* attributes are explicitly allowed by the wildcard `other`:

  - `fs:fs`, OPTIONAL
  - `fs:subFs`, OPTIONAL

4.2.2.10 originalData

Unit-level collection of original data for the inline codes.

Contains:

- One or more `<data>` elements

4.2.2.11 data

Storage for the original data of an inline code.

Contains:

- Non-translatable text
- Zero, one or more `<cp>` elements.

Non-translatable text and `<cp>` elements MAY appear in any order.

Attributes:

- `id`, REQUIRED
- `dir`, OPTIONAL

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

- xml:space, OPTIONAL, the value is restricted to preserve on this element

4.2.2.12 source

Portion of text to be translated.

Contains:

- Text
- Zero, one or more <cp> elements
- Zero, one or more <ph> elements
- Zero, one or more <pc> elements
- Zero, one or more <sc> elements
- Zero, one or more <ec> elements
- Zero, one or more <mrk> elements
- Zero, one or more <sm> elements
- Zero, one or more elements

Text and inline elements may appear in any order.

Attributes:

- xml:lang, OPTIONAL
- xml:space, OPTIONAL

Constraints

- When a <source> element is a child of or <ignorable>, the explicit or inherited value of the OPTIONAL xml:lang attribute MUST be equal to the value of the srcLang attribute of the enclosing <xliff> element.

4.2.2.13 target

The translation of the sibling <source> element.

Contains:

- Text
- Zero, one or more <cp> elements
- Zero, one or more <ph> elements
- Zero, one or more <pc> elements
- Zero, one or more <sc> elements
- Zero, one or more <ec> elements
- Zero, one or more <mrk> elements
- Zero, one or more <sm> elements
- Zero, one or more elements

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

Text and inline elements may appear in any order.

Attributes:

- xml:lang, OPTIONAL
- xml:space, OPTIONAL
- order, OPTIONAL

Constraints

- When a <target> element is a child of or <ignorable>, the explicit or inherited value of the OPTIONAL xml:lang MUST be equal to the value of the trgLang attribute of the enclosing <xliff> element.

4.2.3 Inline Elements

The *XLIFF Core* inline elements at the <source> or <target> level are: <cp>, <ph>, <pc>, <sc>, <ec>, <mrk>, <sm> and .

The elements at the <unit> level directly related to inline elements are: <originalData> and <data>.

4.2.3.1 cp

Represents a Unicode character that is invalid in XML.

Contains:

This element is always empty.

Parents:

<data>, <mrk>, <source>, <target> and <pc>

Attributes:

- hex, REQUIRED

Example:

```
<unit id="1">
  <segment>
    <source>Ctrl+C=<cp hex="0003"/></source>
  </segment>
</unit>
```

The example above shows a character U+0003 (Control C) as it has to be represented in XLIFF.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

Processing Requirements

- *Writers* MUST encode all invalid XML characters of the content using `<cp>`.
- *Writers* MUST NOT encode valid XML characters of the content using `<cp>`.

4.2.3.2 ph

Represents a standalone code of the original format.

Contains:

This element is always empty.

Parents:

`<source>`, `<target>`, `<pc>` and `<mrk>`

Attributes:

- `canCopy`, OPTIONAL
- `canDelete`, OPTIONAL
- `canReorder`, OPTIONAL
- `copyOf`, OPTIONAL
- `disp`, OPTIONAL
- `equiv`, OPTIONAL
- `id`, REQUIRED.
- `dataRef`, OPTIONAL
- `subFlows`, OPTIONAL
- `subType`, OPTIONAL
- `type`, OPTIONAL
- attributes from other namespaces, OPTIONAL

Example:

```
<unit id="1">
  <originalData>
    <data id="d1">%d</data>
    <data id="d2">&lt;br/></data>
  </originalData>
  <segment>
    <source>Number of entries: <ph id="1" dataRef="d1" /><ph id="2"
        dataRef="d2"/>(These entries are only the ones matching the
        current filter settings)</source>
  </segment>
</unit>
```

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

Constraints

- The following *XLIFF Module* attributes are explicitly allowed by the wildcard `other`:

  - attributes from the namespace `urn:oasis:names:tc:xliff:fs:2.0`, OPTIONAL, provided that the Constraints specified in the Format Style Module are met.
  - attributes from the namespace `urn:oasis:names:tc:xliff:sizerestriction:2.0`, OPTIONAL, provided that the Constraints specified in the Size and Length Restriction Module are met.

- No other attributes MUST be used.

Processing Requirements

- *Extractors* MUST NOT use the `<ph>` element to represent spanning codes.

  Rationale: Using a standalone placeholder code for a spanning code does not allow for controlling the span (for instance tag order and data integrity) when *Modifying* inline content and is in *direct contradiction* to the business logic described in Representation of the codes and normative statements included in Usage of <pc> and <sc>/<ec>

  Note

  It is possible although not advised to use <ph> to mask non translatable inline content. The preferred way of protecting portions of inline content from translation is the *Core* Translate Annotation. See also discussion in the ITS Module section on representing translatability inline..

4.2.3.3 pc

Represents a well-formed spanning original code.

Contains:

- Text
- Zero, one or more `<cp>` elements
- Zero, one or more `<ph>` elements
- Zero, one or more `<pc>` elements
- Zero, one or more `<sc>` elements
- Zero, one or more `<ec>` elements
- Zero, one or more `<mrk>` elements
- Zero, one or more `<sm>` elements
- Zero, one or more `<em>` elements

Text and inline elements may appear in any order.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

Parents:

- `<source>`
- `<target>`
- `<pc>`
- `<mrk>`

Attributes:

- `canCopy`, OPTIONAL
- `canDelete`, OPTIONAL
- `canOverlap`, OPTIONAL
- `canReorder`, OPTIONAL
- `copyOf`, OPTIONAL
- `dispEnd`, OPTIONAL
- `dispStart`, OPTIONAL
- `equivEnd`, OPTIONAL
- `equivStart`, OPTIONAL
- `id`, REQUIRED
- `dataRefEnd`, OPTIONAL
- `dataRefStart`, OPTIONAL
- `subFlowsEnd`, OPTIONAL
- `subFlowsStart`, OPTIONAL
- `subType`, OPTIONAL
- `type`, OPTIONAL
- `dir`, OPTIONAL
- attributes from other namespaces, OPTIONAL

Example:

```
<unit id="1">
  <originalData>
    <data id="1">&lt;B&gt;</data>
    <data id="2">&lt;/B&gt;</data>
  </originalData>
  <segment><pc id="1" dataRefStart="1" dataRefEnd="2">
      Important</pc> text</source></segment>
</unit>
```

Constraints

- The following *XLIFF Module* attributes are explicitly allowed by the wildcard `other`:

  - attributes from the namespace `urn:oasis:names:tc:xliff:fs:2.0`, OPTIONAL, provided that the Constraints specified in the Format Style Module are met.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

- attributes from the namespace `urn:oasis:names:tc:xliff:sizerestriction:2.0`, OPTIONAL, provided that the Constraints specified in the Size and Length Restriction Module are met.

- No other attributes MUST be used.

Processing Requirements

- *Extractors* MUST NOT use the `<pc>` element to represent standalone codes.

  Rationale: Using a spanning code for a standalone code can easily result in having text inside a span where the original format does not allow it.

4.2.3.4 sc

Start of a spanning original code.

Contains:

This element is always empty.

Parents:

`<source>`, `<target>`, `<pc>` and `<mrk>`

Attributes:

- `canCopy`, OPTIONAL
- `canDelete`, OPTIONAL
- `canOverlap`, OPTIONAL
- `canReorder`, OPTIONAL
- `copyOf`, OPTIONAL
- `dataRef`, OPTIONAL
- `dir`, OPTIONAL
- `disp`, OPTIONAL
- `equiv`, OPTIONAL
- `id`, REQUIRED
- `isolated`, OPTIONAL
- `subFlows`, OPTIONAL
- `subType`, OPTIONAL
- `type`, OPTIONAL
- attributes from other namespaces, OPTIONAL

Example:

```
<unit id="1">
  <segment>
    <source><sc id="1" type="fmt" subType="xlf:b"/>
```

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

```
        First sentence. </source>
  </segment>
  <segment>
    <source>Second sentence.<ec startRef="1" type="fmt"
        subType="xlf:b"/></source>
  </segment>
</unit>
```

Constraints

- The following *XLIFF Module* attributes are explicitly allowed by the wildcard `other`:

  - attributes from the namespace `urn:oasis:names:tc:xliff:fs:2.0`, OPTIONAL, provided that the Constraints specified in the Format Style Module are met.
  - attributes from the namespace `urn:oasis:names:tc:xliff:sizerestriction:2.0`, OPTIONAL, provided that the Constraints specified in the Size and Length Restriction Module are met.

- No other attributes MUST be used.
- The values of the attributes `canCopy`, `canDelete`, `canReorder` and `canOverlap` MUST be the same as the values the ones in the `<ec>` element corresponding to this start code.
- The attribute `isolated` MUST be set to `yes` if and only if the `<ec>` element corresponding to this start marker is not in the same `<unit>`, and set to `no` otherwise.

Processing Requirements

- *Extractors* MUST NOT use the `<sc>` / `<ec>` pair to represent standalone codes.

  Rationale: Using a spanning code for a standalone code can easily result in having text inside a span where the original format does not allow it.

4.2.3.5 ec

End of a spanning original code.

Contains:

This element is always empty.

Parents:

`<source>`, `<target>`, `<pc>` and `<mrk>`

Attributes:

- `canCopy`, OPTIONAL
- `canDelete`, OPTIONAL
- `canOverlap`, OPTIONAL

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

- canReorder, OPTIONAL
- copyOf, OPTIONAL
- dataRef, OPTIONAL
- dir, OPTIONAL
- disp, OPTIONAL
- equiv, OPTIONAL
- id, OPTIONAL
- isolated, OPTIONAL
- startRef, OPTIONAL
- subFlows, OPTIONAL
- subType, OPTIONAL
- type, OPTIONAL
- attributes from other namespaces, OPTIONAL

Example:

```
<unit id="1">
  <originalData>
    <data id="d1">\b </data>
    <data id="d2">\i </data>
    <data id="d3">\b0 </data>
    <data id="d4">\i0 </data>
  </originalData>
  <segment>
    <source>Text in <sc id="1" dataRef="d1"/>bold <sc id="2"
        dataRef="d2"/> and<ec startRef="1" dataRef="d3"/>
         italics<ec startRef="2" dataRef="d4"/>. </source>
  </segment>
</unit>
```

Constraints

- The following *XLIFF Module* attributes are explicitly allowed by the wildcard `other`:

  - attributes from the namespace `urn:oasis:names:tc:xliff:fs:2.0`, OPTIONAL, provided that the Constraints specified in the Format Style Module are met.
  - attributes from the namespace `urn:oasis:names:tc:xliff:sizerestriction:2.0`, OPTIONAL, provided that the Constraints specified in the Size and Length Restriction Module are met.

- No other attributes MUST be used.
- The values of the attributes canCopy, canDelete and canOverlap MUST be the same as the values the ones in the <sc> element corresponding to this end code.
- The value of the attribute canReorder MUST be `no` if the value of canReorder is `firstNo` in the <sc> element corresponding to this end code.
- The attribute isolated MUST be set to `yes` if and only if the <sc> element corresponding to this end code is not in the same <unit> and set to `no` otherwise.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

- If and only if the attribute `isolated` is set to `yes`, the attribute `id` MUST be used instead of the attribute `startRef` that MUST be used otherwise.
- If and only if the attribute `isolated` is set to `yes`, the attribute `dir` MAY be used, otherwise the attribute `dir` MUST NOT be used on the `<ec>` element.

Processing Requirements

- *Extractors* MUST NOT use the `<sc>` / `<ec>` pair to represent standalone codes.

  Rationale: Using a spanning code for a standalone code can easily result in having text inside a span where the original format does not allow it.

4.2.3.6 mrk

Represents an annotation pertaining to the marked span.

Contains:

- Text
- Zero, one or more `<cp>` elements
- Zero, one or more `<ph>` elements
- Zero, one or more `<pc>` elements
- Zero, one or more `<sc>` elements
- Zero, one or more `<ec>` elements
- Zero, one or more `<mrk>` elements
- Zero, one or more `<sm>` elements
- Zero, one or more `<em>` elements

Text and inline elements may appear in any order.

Parents:

`<source>`, `<target>`, `<pc>` and `<mrk>`

Attributes:

- `id`, REQUIRED
- `translate`, OPTIONAL
- `type`, OPTIONAL
- `ref`, OPTIONAL
- `value`, OPTIONAL
- attributes from other namespaces, OPTIONAL

Constraints

- The [XML namespace] MUST NOT be used at this extension point.
- The following *XLIFF Module* attributes are explicitly allowed by the wildcard `other`:

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

- attributes from the namespace `urn:oasis:names:tc:xliff:fs:2.0`, OPTIONAL, provided that the Constraints specified in the Format Style Module are met.
- attributes from the namespace `urn:oasis:names:tc:xliff:sizerestriction:2.0`, OPTIONAL, provided that the Constraints specified in the Size and Length Restriction Module are met.
- attributes from the namespace `http://www.w3.org/2005/11/its`, OPTIONAL, provided that the Constraints specified in the ITS Module are met.
- attributes from the namespace `urn:oasis:names:tc:xliff:itsm:2.1`, OPTIONAL, provided that the Constraints specified in the ITS Module are met.

See the Annotations section for more details and examples on how to use the `<mrk>` element.

4.2.3.7 sm

Start marker of an annotation where the spanning marker cannot be used for well-formedness reasons.

Contains:

This element is always empty.

Parents:

`<source>`, `<target>`, `<pc>` and `<mrk>`

Attributes:

- `id`, REQUIRED
- `translate`, OPTIONAL
- `type`, OPTIONAL
- `ref`, OPTIONAL
- `value`, OPTIONAL
- attributes from other namespaces, OPTIONAL

Constraints

- The [XML namespace] MUST NOT be used at this extension point.
- The following *XLIFF Module* attributes are explicitly allowed by the wildcard `other`:

  - attributes from the namespace `urn:oasis:names:tc:xliff:fs:2.0`, OPTIONAL, provided that the Constraints specified in the Format Style Module are met.
  - attributes from the namespace `urn:oasis:names:tc:xliff:sizerestriction:2.0`, OPTIONAL, provided that the Constraints specified in the Size and Length Restriction Module are met.
  - attributes from the namespace `http://www.w3.org/2005/11/its`, OPTIONAL, provided that the Constraints specified in the ITS Module are met.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

- attributes from the namespace `urn:oasis:names:tc:xliff:itsm:2.1`, OPTIONAL, provided that the Constraints specified in the ITS Module are met.

See the Annotations section for more details and examples on how to use the `<sm>` element.

4.2.3.8 em

End marker of an annotation where the spanning marker cannot be used for well-formedness reasons.

Contains:

This element is always empty.

Parents:

`<source>`, `<target>`, `<pc>` and `<mrk>`

Attributes:

- `startRef`, REQUIRED

See the Annotations section for more details and examples on how to use the `<em>` element.

4.3 Attributes

This section lists all the various attributes used in XLIFF core elements.

4.3.1 XLIFF Attributes

The attributes defined in XLIFF 2.0 are: `appliesTo`, `canCopy`, `canDelete`, `canOverlap`, `canReorder`, `canResegment`, `category`, `copyOf`, `dataRef`, `dataRefEnd`, `dataRefStart`, `dir`, `disp`, `dispEnd`, `dispStart`, `equiv`, `equivEnd`, `equivStart`, `hex`, `href`, `id`, `isolated`, `name`, `order`, `original`, `priority`, `ref`, `srcDir`, `srcLang`, `startRef`, `state`, `subFlows`, `subFlowsEnd`, `subFlowsStart`, `subState`, `subType`, `trgLang`, `translate`, `trgDir`, `type`, `value` and `version`.

4.3.1.1 appliesTo

Comment target - indicates the element to what the content of the note applies.

Value description: `source` or `target`.

Default value: undefined.

Used in: `<note>`.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

4.3.1.2 canCopy

Replication editing hint - indicates whether or not the inline code can be copied.

Value description: `yes` if the code can be copied, `no` if the code is not intended to be copied.

Default value: `yes`.

Used in: `<pc>`, `<sc>`, `<ec>`, `<ph>`.

4.3.1.3 canDelete

Deletion editing hint - indicates whether or not the inline code can be deleted.

Value description: `yes` if the code can be deleted, `no` if the code is not allowed to be deleted.

Default value: `yes`.

Used in: `<pc>`, `<sc>`, `<ec>`, `<ph>`.

4.3.1.4 canOverlap

Code can overlap - indicates whether or not the spanning code where this attribute is used can enclose partial spanning codes (i.e. a start code without its corresponding end code, or an end code without its corresponding start code).

Value description: `yes` or `no`.

Default value: default values for this attribute depend on the element in which it is used:

- When used in `<pc>`: `no`.
- When used in `<sc>` or `<ec>`: `yes`.

Used in: `<pc>`, `<sc>` and `<ec>`

Example:

```
<unit id="1">
  <originalData>
    <data id="1">\i1 </data>
    <data id="2">\i0 </data>
    <data id="3">{\b </data>
    <data id="4">}</data>
  </originalData>
  <segment>
    <source><pc id="1" dataRefStart="3" dataRefEnd="4"
canOverlap="no">
        Bold, <sc id="2" dataRef="1" canOverlap="yes"/>both</pc>,
         italics<ec startRef="2" dataRef="2"/></source>
  </segment>
</unit>
```

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

4.3.1.5 canReorder

Re-ordering editing hint - indicates whether or not the inline code can be re-ordered. See Editing Hints section for more details.

Value description: `yes` in case the code can be re-ordered, `firstNo` when the code is the first element of a sequence that cannot be re-ordered, `no` when it is another element of such a sequence.

Default value: `yes`.

Used in: `<pc>`, `<sc>`, `<ec>`, `<ph>`.

For the normative Usage Description see Constraints and Processing Requirements in the Editing Hints section.

4.3.1.6 canResegment

Can resegment - indicates whether or not the source text in the scope of the given `canResegment` flag can be reorganized into a different structure of `<segment>` elements within the same parent `<unit>`.

Value description: `yes` or `no`.

Default value: default values for this attribute depend on the element in which it is used:

- When used in `<file>`:

  The value `yes`.

- When used in any other element:

  The value of the `canResegment` attribute of its parent element.

Used in: `<file>` `<group>` `<unit>`, and `<segment>`.

4.3.1.7 category

Category - provides a way to categorize notes.

Value description: Text.

Default value: undefined

Used in: `<note>`.

4.3.1.8 copyOf

Reference to base code - holds the `id` of the base code of a copied code.

Value description: NMTOKEN. The `id` value of the base code of which this code is a copy.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

Default value: undefined

Used in: <ph>, <pc>, <sc>, <ec>.

Example:

```
<unit id="1">
  <segment>
    <source>Äter <pc id="1">katter möss</pc>?</source>
    <target>Do <pc id="1">cats</pc> eat <pc id="2" copyOf="1">
        mice</pc>? </target>
  </segment>
</unit>
```

4.3.1.9 dataRef

Original data reference - holds the identifier of the <data> element that contains the original data for a given inline code.

Value description: An [XML Schema Datatypes] NMTOKEN that MUST be the value of the id attribute of one of the <data> element listed in the same <unit> element.

Default value: undefined.

Used in: <ph>, <sc>, <ec>.

Example:

```
<unit id="1">
  <originalData>
    <data id="d1">{0}</data>
  </originalData>
  <segment>
    <source>Error in '<ph id="1" dataRef="d1"/>'.</source>
    <target>Erreur dans '<ph id="1" dataRef="d1"/>'.</target>
  </segment>
</unit>
```

The example above shows a <ph> element that has its original data stored outside the content, in a <data> element.

4.3.1.10 dataRefEnd

Original data reference - holds the identifier of the <data> element that contains the original data for the end marker of a given inline code.

Value description: An [XML Schema Datatypes] NMTOKEN that MUST be the value of the id attribute of one of the <data> element listed in the same <unit> element.

Default value: undefined.

Used in: <pc>.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

Example:

```
<unit id="1">
  <originalData>
    <data id="d1">&lt;EM></data>
    <data id="d2">&lt;/EM></data>
  </originalData>
  <segment>
    <source><pc id="1" dataRefStart="d1" dataRefEnd="d2">
        Efficiency</pc> is the operative word here.</source>
    <target><pc id="1" dataRefStart="d1" dataRefEnd="d2">
        Efficacité</pc> est le mot clé ici.</target>
  </segment>
</unit>
```

The example above shows two `<pc>` elements with their original data stored outside the content, in two `<data>` elements.

4.3.1.11 dataRefStart

Original data reference - holds the identifier of the `<data>` element that contains the original data for the start marker of a given inline code.

Value description: An [XML Schema Datatypes] NMTOKEN that MUST be the value of the `id` attribute of one of the `<data>` element listed in the same `<unit>` element.

Default value: undefined.

Used in: `<pc>`.

Example:

```
<unit id="1">
  <originalData>
    <data id="d1">&lt;EM></data>
    <data id="d2">&lt;/EM></data>
  </originalData>
  <segment>
    <source><pc id="1" dataRefStart="d1" dataRefEnd="d2">
        Efficiency</pc> is the operative word here.</source>
    <target><pc id="1" dataRefStart="d1" dataRefEnd="d2">
        Efficacité</pc> est le mot clé ici.</target>
  </segment>
</unit>
```

The example above shows two `<pc>` elements with their original data stored outside the content, in two `<data>` elements.

4.3.1.12 dir

Directionality - indicates the directionality of content.

Value description: `ltr` (Left-To-Right), `rtl` (Right-To-Left), or `auto` (determined heuristically, based on the first strong directional character in scope, see [UAX #9]).

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

Default value: default values for this attribute depend on the element in which it is used:

- When used in a <pc>, <sc>, or <ec> element that has a <source> element as its parent:

  The value of the srcDir attribute of the <unit> element, in which the elements are located.

- When used in a <pc>, <sc>, or <ec> element that has a <target> element as its parent:

  The value of the trgDir attribute of the <unit> element, in which the elements are located.

- When used in a <pc>, <sc>, or <ec> element that has a <pc> element as its parent:

  The value of the dir attribute of the parent <pc> element.

- When used in <data>:

  The value auto.

Used in: <data>, <pc>, <sc>, and <ec>.

4.3.1.13 disp

Display text - holds an alternative user-friendly display representation of the original data of the inline code.

Value description: Text.

Default value: undefined

Used in: <ph>, <sc>, <ec>.

Example:

```
<unit id="1">
  <originalData>
    <data id="d1">{1}</data>
  </originalData>
  <segment>
    <source>Welcome back <ph id="1" disp="[UserName]" dataRef="d1"/>!
        </source>
  </segment>
</unit>
```

Note

To provide a plain text equivalent of the code, use the equiv attribute.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

4.3.1.14 dispEnd

Display text - holds an alternative user-friendly display representation of the original data of the end marker of an inline code.

Value description: Text.

Default value: undefined

Used in: `<pc>`.

Example:

```
<unit id="1">
  <originalData>
    <data id="d1">\cf1\ul\b\f1\fs24 </data>
    <data id="d2">\cf0\ulnone\b0\f0\fs22 </data>
  </originalData>
  <segment>
    <source>Example of <pc id="1" dataRefStart="d1" dataRefEnd="d2"
        dispStart="&lt;span>" dispEnd="&lt;/span>">
        formatted text</pc>.</source>
  </segment>
</unit>
```

In the example above, the `dispStart` and `dispEnd` attributes provide a more user-friendly representation of the original formatting codes.

Note

To provide a plain text equivalent of the code, use the `equivEnd` attribute.

4.3.1.15 dispStart

Display text - holds an alternative user-friendly display representation of the original data of the start marker of an inline code.

Value description: Text.

Default value: undefined

Used in: `<pc>`.

Example:

```
<unit id="1">
  <originalData>
    <data id="d1">\cf1\ul\b\f1\fs24 </data>
    <data id="d2">\cf0\ulnone\b0\f0\fs22 </data>
  </originalData>
  <segment>
    <source>Example of <pc id="1" dataRefStart="d1" dataRefEnd="d2"
        dispStart="&lt;span>" dispEnd="&lt;/span>">
```

Documents
Document Sharing Hub

© ISO 2024 – All rights reserved

Full document details: BS ISO 21720:2024

```
        formatted text</pc>.</source>
  </segment>
</unit>
```

In the example above, the dispStart and dispEnd attributes provide a more user-friendly representation of the original formatting codes.

Note

To provide a plain text equivalent of the code, use the equivStart attribute.

4.3.1.16 equiv

Equivalent text - holds a plain text representation of the original data of the inline code that can be used when generating a plain text representation of the content.

Value description: Text.

Default value: an empty string.

Used in: <ph>, <sc>, <ec>.

Example:

```
<unit id="1">
  <originalData>
    <data id="d1">&amp;</data>
  </originalData>
  <segment>
    <source>Open <ph id="1" equiv="" dataRef="d1"/>File</source>
  </segment>
</unit>
```

In this example the equiv attribute of the <ph> element is used to indicate that the original data of the code can be ignored in the text representation of the string. This could, for instance, help a spell-checker tool to process the content as "Open File".

Note

To provide a user-friendly representation, use the disp attribute.

4.3.1.17 equivEnd

Equivalent text - holds a plain text representation of the original data of the end marker of an inline code that can be used when generating a plain text representation of the content.

Value description: Text.

Default value: an empty string

Used in: <pc>.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

Example:

```
<unit id="1">
  <originalData>
    <data id="d1">&lt;span class="link" onclick="linkTo('dbId5345')">
        </data>
    <data id="d2">&lt;/span></data>
  </originalData>
  <segment>
    <source>The jam made of <pc id="1" dataRefStart="d1"
equivStart=""
        dataRefEnd="d2" equivEnd="">lingonberries</pc> is quite
        tasty.</source>
  </segment>
</unit>
```

Note

To provide a user-friendly representation, use the `dispEnd` attribute.

4.3.1.18 equivStart

Equivalent text - holds a plain text representation of the original data of the start marker of an inline code that can be used when generating a plain text representation of the content.

Value description: Text.

Default value: an empty string

Used in: `<pc>`.

Example:

```
<unit id="1">
  <originalData>
    <data id="d1">&lt;span class="link" onclick="linkTo('dbId5345')">
        </data>
    <data id="d2">&lt;/span></data>
  </originalData>
  <segment>
    <source>The jam made of <pc id="1" dataRefStart="d1"
equivStart=""
        dataRefEnd="d2" equivEnd="">lingonberries</pc> is quite
        tasty.</source>
  </segment>
</unit>
```

Note

To provide a user-friendly representation, use the `dispStart` attribute.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

4.3.1.19 hex

Hexadecimal code point - holds the value of a Unicode code point that is invalid in XML.

Value description: A canonical representation of the hexBinary [XML Schema Datatypes] data type: Two hexadecimal digits to represent each octet of the Unicode code point. The allowed values are any of the values representing code points invalid in XML, between hexadecimal 0000 and 10FFFF (both included).

Default value: undefined

Used in: `<cp>`.

Example:

```
<cp hex="001A"/><cp hex="0003"/>
```

The example above shows a character U+001A and a character U+0003 as they have to be represented in XLIFF.

4.3.1.20 href

href - a pointer to the location of an external skeleton file pertaining to the enclosing `<file>` element..

Value description: IRI.

Default value: undefined

Used in: `<skeleton>`.

4.3.1.21 id

Identifier - a character string used to identify an element.

Value description: NMTOKEN. The scope of the values for this attribute depends on the element, in which it is used.

- When used in a `<file>` element:

  The value MUST be unique among all `<file>` `id` attribute values within the enclosing `<xliff>` element.

- When used in `<group>` elements:

  The value MUST be unique among all `<group>` `id` attribute values within the enclosing `<file>` element.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

- When used in `<unit>` elements:

  The value MUST be unique among all `<unit>` `id` attribute values within the enclosing `<file>` element.

- When used in `<note>` elements:

  The value MUST be unique among all `<note>` `id` attribute values within the immediate enclosing `<file>`, `<group>`, or `<unit>` element.

- When used in `<data>` elements:

  The value MUST be unique among all `<data>` `id` attribute values within the enclosing `<unit>` element.

- When used in `<segment>`, `<ignorable>`, `<mrk>`, `<sm>`, `<pc>`, `<sc>`, `<ec>`, or `<ph>` elements:
  - The inline elements enclosed by a `<target>` element MUST use the duplicate `id` values of their corresponding inline elements enclosed within the sibling `<source>` element if and only if those corresponding elements exist.
  - Except for the above exception, the value MUST be unique among all of the above within the enclosing `<unit>` element.

  Note

  All of the above defined uniqueness scopes ignore *Module* and *Extension* data. It would be impossible to impose those uniqueness requirements onto *Module* or *Extension* data. As *Core* only *Modifiers* could inadvertently cause conflicts with *Modules* or *Extensions* based data they cannot access. *Modules* and *Extensions* reusing *Core* need to specify their own uniqueness scopes for the xlf:id. In general, *Modules* and *Extensions* are advised to mimic the *Core* uniqueness requirement within their specific wrapper elements enclosing the reused *Core* elements or attributes, yet *Module* or *Extensions* are free to set wider uniqueness scopes if it makes business sense.

Default value: undefined

Used in: `<file>`, `<group>`, `<unit>`, `<note>`, `<segment>`, `<ignorable>`, `<data>`, `<sc>`, `<ec>`, `<ph>`, `<pc>`, `<mrk>` and `<sm>`.

4.3.1.22 isolated

Orphan code flag - indicates if the start or end marker of a spanning inline code is not in the same `<unit>` as its corresponding end or start code.

Value description: `yes` if this start or end code is not in the same `<unit>` as its corresponding end or start code, `no` if both codes are in the same `<unit>`.

Default value: `no`.

Used in: `<sc>`, `<ec>`.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

Example:

```
<file id="f2" xmlns:abc="urn:abc">
  <unit id="1">
    <mtc:matches>
      <mtc:match id="tc01" ref="seg2">
        <source><sc id="1" isolated="yes"/>Warning:</source>
        <target><sc id="1" isolated="yes"/>Attention :</target>
      </mtc:match>
    </mtc:matches>
    <segment id="seg2">
      <source><pc id="1">Warning: File not found.</pc></source>
    </segment>
  </unit>
</file>
```

In the example above the `<sc>` elements have their `isolated` attribute set to `yes` because they do not have their corresponding `<ec>` elements.

4.3.1.23 name

Resource name - the original identifier of the resource corresponding to the *Extracted* `<unit>` or `<group>`.

For example: the key in the key/value pair in a Java properties file, the ID of a string in a Windows string table, the index value of an entry in a database table, etc.

Value description: Text.

Default value: undefined.

Used in: `<unit>` and `<group>`.

4.3.1.24 order

target order - indicates the order, in which to compose the target content parts.

Value description: A positive integer.

Default value: implicit, see below

When order is not explicitly set, the `<target>` `order` corresponds to its sibling `<source>`, i.e. it is not being moved anywhere when composing target content of the enclosing `<unit>` and the implicit `order` value is of that position within the `<unit>`.

Used in: `<target>`.

Constraints

- The value of the `order` attribute MUST be unique within the enclosing `<unit>` element.

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024

- The value of each of the `order` attributes used within a `<unit>` element MUST NOT be higher than N, where N is the number of all current `<segment>` and `<ignorable>` children of the said `<unit>` element.

See the Segments Order section for the normative usage description.

4.3.1.25 original

Original file - a pointer to the location of the original document from which the content of the enclosing `<file>` element is extracted.

Value description: IRI.

Default value: undefined

Used in: `<file>`.

4.3.1.26 priority

Priority - provides a way to prioritize notes.

Value description: Integer 1-10.

Default value: `1`

Used in: `<note>`.

> **Note**
>
> Please note that 1 is the highest priority that can be interpreted as an alert, e.g. an [ITS] Localization Note of the type alert. The best practice is to use only one alert per an annotated element, and the full scale of 2-10 can be used for prioritizing notes of lesser importance than the alert.

4.3.1.27 ref

Reference - holds a reference for the associated annotation.

Value description: A value of the [XML Schema Datatypes] type anyURI. The semantics of the value depends on the type of annotation:

- When used in a term annotation, the URI value is referring to a resource providing information about the term.
- When used in a translation candidates annotation, the URI value is referring to an external resource providing information about the translation candidate.
- When used in a comment annotation, the value is referring to a `<note>` element within the same enclosing `<unit>`.
- When used in a custom annotation, the value is defined by each custom annotation.

Default value: undefined

© ISO 2024 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 21720:2024