Overview of XLIFF 2.0

Fredrik Lesteen@lionbridge.com
w. +46-31-3032663
m. +46-707-511247
Where is XLIFF 2.0 Today?

Features are mostly locked and the draft has undergone two rounds of public review.

Where is XLIFF 2.0 Today?

Approve OASIS Committee Spec early in 2014. We are trying to target January. Final OASIS Specification approved ballot in second quarter of 2014.

xliff-comment@lists.oasis-open.org
Current state of XLIFF

- Adoption
- Stability
- Usability
- Interoperability
- Complexity
- One way
Goals for next generation of XLIFF

- Move away from document only view in specification, include process
- Constrain extensibility while still allowing and embracing it
- Include missing features often implemented as extensions in the standard
  - Ex. Right to Left and BiDi text, XML invalid code points, change tracking and validation.
- Reduce number of different ways to do the same thing
  - When multiple ways exist provide mapping between them
- Modularize so that tools can support a simpler core and optional feature modules
- Tighter and less ambiguous specification
- Interoperability - Ecosystem of tools
- Ease of implementation
Key principles for Design of XLIFF 2.0:

1. Not backwards compatible. If there is a need to make a radical change to properly fix an issue, we should do that.
2. Support all used features of XLIFF 1.2 to allow migration to the new standard. Drop by dropping unused features.
3. Provide detailed processing requirements and include requirements on processing agents in the specification.
Streamlined core tree structure

Remove elements that was not used in an interoperable way. The optional metadata module can be used instead.

Remove unnecessary hierarchy

The `<group>` element is now single purpose and only used to group `<unit>` or `<group>` elements. Can no longer be used to merge content.

The `<group>` and `<unit>` structure is static during the processing of the document.
Flexible `<unit>` element

Total translatable content of `<unit>` stays constant during processing. But may be redistributed across `<segment>` and `<ignorable>` elements during processing. `<originalData>` holds content of inline elements. Order of `<targets>` can change logically while keeping them with their linguistically matching source.

Extraction tool stores units of content in a natural way:

```
<unit>
  <segment>
    <source>
    <target>
    <ignoreable>
    <source>
    <target>
  <notes>
    <note>
  <originalData>
    <data>
```
Simplified inline element model

XLIFF 2.0
One set of codes where transform between forms is defined.
<sc>, <ec>, <ph>, <pc>, <cp>

Placeholder <ph> replaces <ph> and <x>
Start code <sc> replaces <bpt>, <bx>, <it>
End code <ec> replaces <ept>, <ex>, <it>
Spanning code <pc> replaces <g>
<pc> can be transformed into <sc>/<ec> pair and vice versa.
<cp> represent characters not allowed in XML

XLIFF 1.2
Improve interoperability by easier to implement content model.

Masking: <bpt>, <ept>, <ph>, <it>
Replacing: <bx>, <ex>, <x>
Spanning: <g>

No defined way to include characters not allowed in XML
Different tools use different subsets of the available codes.
No way to break in the middle of span
Simplified sub flow

Sub flows are separate units within a text flow. They can be recursively nested using sub elements within other inline elements. Tools must support this specifically to translate all text without special handling of sub flows. Only if they display such content inline or recreate the original content there is a need to support it.
Consolidated annotations

The `<mrk>` element from XLIFF 1.2 remains as the core annotation element.

To allow splitting of annotated content into multiple segments start (`<sm>`) and end (`<em>`) marker forms has been added. And conversion between them and `<mrk>` defined.

All markers support the ‘translate’ attribute to control translatability of marked section.

Standard defined annotation types

- Base annotation only controlling translatability
- Terminology annotation. Markup terms and possibly refer to more data. Used with the optional Glossary module for example
- Translation candidate annotation. Markup a span and associate a translation candidate using the optional Translation Candidate Module.
- Comment annotation. Add comment to a span or reference comment in note element.
- Custom annotation with ability to reference resources.
Optional feature modules

Provide advance or non-generic functions in optional modules.
Translation Candidates Module

Key features:
- Matches related metadata
- Can be referenced from spans of text within a <unit>. Not restricted to paragraph or segment.
- More flexible
- More and better defined metadata
- Replaces the <alt-trans> element

Relationship to XLIFF candidates within the XLIFF document
Glossary Module

Can be referenced from spans of text within translation units. Easy to implement.

Key features:
- Provides standard defined format for glossary data. In 1.2 the format of the data was completely undefined.
- Replaces the `<glossary>` header element.
Format Style Module

Allows embedding of formatting information for preview or editing
Allows formatting both at the structural and inline level
Uses HTML for easy implementation
Possibility to create preview using XSLT style sheet
Not intended to provide perfect rendering

Relationship to XLIFF 1.2
New functionality, closes existing feature is the ‘css-style’ attribute available on some elements.
Provides standardized functionality instead of vendor extensions
Metadata Module

Relationship to XML structure and attributes levels in the document if used instead of custom extensions tools can present or index information without understanding it.

Applications can group metadata arbitrarily.
Resource Data Module

Replicates ability to do reference at different levels with a better structured and constrained mechanism.

Provides ability to associate non-textual content that may need to be changed during translation.

Dialog resource data, Graphics, etc.
Change Tracking Module

- Provides revision tracking of the data in the XLIFF file
- Supports tracking of revisions to source, target, and notes
- Flexible design allows module to be used to track changes to any XLIFF element if a tool wishes to do so.

Key features:
- Relationship to XLIFF 1.2
Size Restriction Module

Provides separate restrictions on storage and logical size.
Pre defined profiles for common Unicode encodings and number of characters/code-points
Support specification of text normalization and restriction of spans.
Extensible to handle most needs as vendor extensions

Relationship to XLIFF 1.2
Replaces several attributes used on <group> and <trans-unit>.
Provides better predefined restriction profiles.
Provides extendibility without adding non standard attributes in translatable text.
Validation Module

- **Key features**
  - Ability to specify constraints or rules that target text must follow
  - Hierarchical rules that can be overridden
  - Can be conditional on properties of the corresponding source text
    - Ex. contains, not contains, starts/ends-with, number of occurrences
  - Usually not stored in XLIFF document but embedded in tools or tool configuration

- **Relationship to XLIFF 1.2**