#### Segmentation Modification

When *Modifying* the segmentation of a <unit>, *Modifiers* MUST meet the constraints and follow the processing requirements defined below:

*Constraints*

* Integrity of the inline codes MUST be preserved. See the section on [Inline Codes](http://docs.oasis-open.org/xliff/xliff-core/v2.0/xliff-core-v2.0.html#inlineCodes) and on [Annotations](http://docs.oasis-open.org/xliff/xliff-core/v2.0/xliff-core-v2.0.html#annotations) for details.
* The entire source content of any one [<unit>](http://docs.oasis-open.org/xliff/xliff-core/v2.0/xliff-core-v2.0.html#unit) element MUST remain logically unchanged; [<segment>](http://docs.oasis-open.org/xliff/xliff-core/v2.0/xliff-core-v2.0.html#segment) elements or their data MUST NOT be moved or joined across units.

**Warning**

Note that when splitting or joining segments that have both source and target content it is advisable to keep the resulting segments linguistically aligned, which is likely to require human linguistic expertise and hence manual re-segmentation. If the linguistically correct alignment cannot be guaranteed, discarding the target content and retranslating the resulting source segments is worth considering.

*Processing Requirements*

* When Modifiers perform a split operation:
	+ Only <ignorable> elements or <segment> elements that have their canResegment value resolved to yes MAY be split.
	+ All new <segment> or <ignorable> elements created and their <source> and <target> children MUST have the same attribute values as the original elements they were created from, as applicable, except for the id attributes and, possibly, for the order, state and subState attributes.
	+ Any new id attributes MUST follow the <segment> or <ignorable> id constraints.
	+ If there was a target content in the original segment and if the state attribute of the original segment was not “initial”, the state attribute of the segments resulting from the split (and possibly its corresponding subState attribute) MAY be changed to reflect the fact that the target content may need to be verified as the new segmentation may have desynchronized the alignment between the source and target contents.
* When Modifiers perform a join operation:
	+ Only <ignorable> elements or <segment> elements that have their canResegment value resolved to yes MAY be join with other elements.
* When Modifiers or Mergers perform a join operation:
	+ Two elements (<segment> or <ignorable>) MUST NOT be joined if their <target> have resolved order values that are not consecutive.
	+ The attributes of the elements to be (<segment> or <ignorable>) and the attributes of their <source> and <target> MUST be carried over in the resulting joined elements.
	+ If attributes of elements to be joined (<segment> or <ignorable>) differ, or if the attributes of their <source> or <target>children differ, the resulting joined elements MUST comply with following rules:
		- If the state attributes of the <segment> elements differ: the state attribute of the joined <segment> MUST be set to the “earliest” of the values specified in the original <segment> elements. The sequence of state values are defined in the following order: 1: “initial”, 2: “translated”, 3: “reviewed”, and 4: “final”.
		- The subState attribute MUST be the one associated with the state attribute selected to be used in the joined <segment>. If no subState attribute is associated with that state, the joined <segment> MUST NOT have a subState.
		- If the xml:space attributes differ: The <source> and <target> of the joined element MUST be set to xml:space=”preserve”.
		- If the dir attributes of the <source> or <target> elements differ: The content of the <source> or <target> elements set to a different directionality than the directionality for the <source> or <target> elements of the joined segment MUST be enclosed between Unicode bi-directional control characters reflecting their original directionality (U+202A and U+202C for left-to-right spans, and U+202B and U+202C for right-to-left spans).
* When Modifiers perform either a split or a join operation:
	+ If any <segment> or <ignorable> element of the <unit> had a <target> with an order attribute before the segmentation modification, the <target> of all <segment> and <ignorable> elements in the <unit> MUST be examined and if necessary their order attributes updated to preserve the ordering of the target content before the segmentation modification.
	+ All inline codes and inline annotations MUST be adjusted to respect the notations described in sections “Usage of <pc> and <sc>/<ec>” and “Splitting annotations”.