# On the annotation of TMX translation memories for advanced leveraging in computer-aided translation

**Mikel L. Forcada**

Departament de Llenguatges i Sistemes Informàtics, Universitat d'Alacant, Spain
mlf@dlsi.ua.es

**Abstract**

The term *advanced leveraging* refers to extensions beyond the current usage of translation memory (TM) in computer-aided translation (CAT). One of these extensions is the ability to identify and use matches on the sub-segment level — for instance, using sub-sentential elements when segments are sentences— to help the translator when a reasonable fuzzy-matched proposal is not available; some suchas functionalities have started to become available in commercial CAT tools. Resources such as statistical word aligners, external machine translation systems, glossaries and term bases could be used to identify and annotate segment-level translation units at the sub-segment level, but there is currently no single, agreed standard supporting the interchange of sub-segmental annotation of translation memories to create a richer translation resource. This paper discusses the capabilities and limitations of some current standards, envisages possible alternatives, and ends with a tentative proposal which slightly *abuses* (repurposes) the usage of existing elements in the TMX standard.
**Keywords:** Translation memory, TMX, advanced leveraging, standards, bilingual corpus, machine translation

## 1. Introduction

The term *advanced leveraging* refers to extensions or enhancements (Garcia, 2012) beyond the current usage of translation memory (TM) in computer-aided translation (CAT); the term was probably coined by TAUS, the Translation Automation User Society, who define it as "new translation features that build upon and extend the capabilities of classic TMs by identifying sub-sentence repetitions".[1] One of these extensions is the ability to identify and use matches on the *sub-segment level* — for instance, using sub-sentential elements when segments are sentences— to help the translator when a reasonable fuzzy-matched proposal is not available; some such functionalities have started to become available in commercial CAT tools (Deep Miner in Déjà Vu,[2] Auto-Suggest in SDL Trados,[3] Advanced Leveraging in Multicorpora[4]). Resources such as statistical word aligners (Och and Ney, 2003) followed by *phrase pair* (corresponding subsegment pair) extraction (Zens et al., 2002; Koehn, 2010), external machine translation systems, glossaries and term bases could be used to identify and annotate segment-level translation units at the sub-segment level, but there is currently no single, agreed standard supporting the interchange of sub-segment annotation of translation memories to create a richer translation resource. This paper discusses the capabilities and limitations of current standards and envisages possible alternatives.

The paper is organized as follows: Section 2. gives an example of using machine translation to discover and annotate sub-segment correspondences in the translation memory; Section 3. discusses stand-off versus embedded annotation; Section 4. discusses the limitations of current standards for stand-off annotation; Section 5. studies how elements in the TMX standard can be slightly *abused* to accomodate embedded annotation of sub-segment units. The article ends with concluding remarks (Section 6.).

## 2. An example: using an external machine translation system to detect sub-segment correspondences

Machine translation (MT) is one of the sources of information that may be used to discover sub-segment correspondences in a TM. Following the example in (Esplà-Gomis et al., 2011), if a Spanish–English TM contains the translation unit *("la situación humanitaria parece ser difícil", "the humanitarian situation appears to be difficult")*, and one sends all the possible subsegments of the Spanish phrase to a Spanish–English machine translation system, and then searches in the English part for their translations (and even sends all possible subsegments of the English phrase to an English–Spanish MT system and then searches in the Spanish part for their translations), one can identify the following sub-segment correspondences (where the numbers in brackets indicate the word positions spanned): *("la" [1-1],"the" [1-1]), ("situación" [2-2], "situation" [3-3]), ("humanitaria" [3-3], "humanitarian" [2-2]), ("ser" [5-5],"be" [6-6]), ("difícil" [6-6],"difficult" [7-7]), ("situación humanitaria" [2-3],"humanitarian situation" [2-3]), ("ser difícil" [5-6], "be difficult" [6-7]), ("la situación humanitaria" [1-3], "the humanitarian situation" [1-3])*, which could then be used to perform advanced leveraging on the TM. Note that in the example, annotations would only occur when both the queried MT system and the professionally-produced TM "agree", and that one could avoid short, one-word subsegments, as they may be too ambiguous. When carefully obtained, these annotations may be used as "safer" alternatives to the usage of raw MT output; for instance, if they are used to propose completions in an interactive machine translation system (Pérez-Ortiz et al., 2014).

Once these correspondences are found, it would be desirable to have a standard way of annotating the TM by indi-

[1] https://www.taus.net/reports/advanced-leveraging.

[2] http://tinyurl.com/dvx2dm

[3] http://tinyurl.com/sdltas

[4] http://multicorpora.com/resources/advanced-leveraging/

cating the initial and final position of the spans covered in each language together with information about the source of "evidence" used (an MT system, in this case).

## 3. Stand-off or embedded annotation?

Let us assume that the translation memory is stored using TMX, the translation memory exchange format.[5] The annotation should be designed in such a way that, once the CAT system retrieves a TU from the TM as a fuzzy match for the current segment, one can easily retrieve the associated sub-segment annotations.

There are two ways to annotate the TMX file. One possibility is to use a *stand-off* annotation, that is, one "that resides in a location [file] different from the location [file] of the data being described by it".[6] Another possibility is to try and find ways to use existing elements in the TMX standard to create an *embedded* annotation in the translation memory.

Any annotation should (a) make it easy to retrieve sub-segment alignment information after retrieving a fuzzy matched translation unit and (b) make it possible to use a given sub-segment translation unit to annotate more than one TU in the original TM: sub-segment translation units are useful because they are likely to repeat more often than full-segment translation units.

If using stand-off notation, there should be an easy way to access the sub-segment translation units that annotate a full-segment translation unit in the TM, perhaps through the value of attribute `tuid`, the unique identifier of each translation unit `<tu>` in the TMX file.

The following sections explore a range of embedded and stand-off alternatives and discuss their pros and cons.

## 4. GrAF-style standoff annotation?

A well-known stand-off annotation standard for text is GrAF (Ide and Suderman, 2007), designed to represent multiple linguistic annotations as a single, XML-serialized directed graph: character (not word) spans are defined as *sink* nodes (that is, having an *outdegree* of zero),

```
<seg:sink seg:id="41"
 seg:start="113" seg:end="129"/>
<seg:sink seg:id="42"
 seg:start="189" seg:end="214"/>
```

(two segments, 41 and 42, spanning characters 113–129 and 189–214), and then an annotation is defined for any node by connecting an annotation node (representing a translation relation)

```
<msd:node msd:id="315"
 name="machine-translation"
 value="google-translate-de-en"/>
```

through (in this case, two) edges

```
<msd:edge from="msd:315" to="41"/>
<msd:edge from="msd:315" to="42"/>
```

to indicate that the segments 41 and 42 are related by machine translation.

In approximately this way, GRaF would allow for alternative segmentations of the same segment (overlapping or nested character spans may be defined), but it assumes a single plain text (seen as a *corpus*). Granted, a TMX file is indeed a text file, and character spans could be used to refer to sub-segments inside the `<seg>` elements of each TU, but this would be very fragile, as it is much more likely for TUs to be inserted in a TM than for a plain-text corpus to be edited, and maintenance would be very hard or subject to a strict discipline. But worse, GrAF does not provide a way to easily encode repeated sub-segment translation units, as it assumes that each text span is a different sink. Finally, there would be no easy way to retrieve all sub-segment TUs for a fuzzy matched segment-level TU, even if `tuid` values are part of the annotation.

This is not to say that the ideas in GrAF could not be used to inspire an annotation scheme, but the standard is not ready yet to be used with TMX; therefore, it can be safely discarded for the time being.

## 5. "Abusing" TMX

This section explores the capabilities of the TMX standard to perform sub-segment annotation of the kind desired. This has the advantage that there is no need to bring together two different XML standards, but, as will be seen, can only be done by stretching ("abusing") somewhat the TMX definition. The proposals provide inspiration for possible extensions of TMX that could be considered for embedded sub-segment annotation; a proposal in that direction would need a more careful study.

### 5.1. Additional, *phrase* TUs and `<prop>`

Sub-segment TUs are also TUs, and could in principle be stored in a TMX-compliant file (either in the same TMX file, or in a separated, stand-off one). TMX provides for some mechanisms that could be explored, or perhaps abused, to implement the desired annotation.

For instance, if one has the following pair of segments or *sentences*:

```
<tu segtype="sentence" tuid="13123123">
 <tuv xml:lang="de">
   <seg>Ich habe einen Artikel
   geschrieben.</seg>
 </tuv>
 <tuv xml:lang="en">
  <seg>I have written an article</seg>
 </tuv>
</tu>
```

and wants to annotate the fact that the subsegments or *phrases* "einen Artikel" and "an article" correspond to each other, one could use the same TMX file (embedded annotation) or a different TMX file (stand-off annotation) in this way:

```
<tu segtype="phrase" tuid="984120312">
 <prop
   type="annotated-tuid">13123123</prop>
```

[5] http://www.gala-global.org/oscarStandards/tmx/tmx14b.html

[6] http://wiki.tei-c.org/index.php/Stand-off_markup

```
 <prop
   type="source"
 >google-translate-de-en</prop>
 <tuv xml:lang="de">
  <prop type="start-pos">10</prop>
  <prop type="end-pos">22</prop>
  <seg>einen Artikel</seg>
 </tuv>
 <tuv xml:lang="en">
  <prop type="start-pos">16</prop>
  <prop type="end-pos">25</prop>
  <seg>an article</seg>
 </tuv>
</tu>
```

This means that *einen Artikel* (spanning character positions 10 to 22 in *Ich habe einen Artikel geschrieben*) and *an article* (spanning positions 16 to 25 in *I have written an article*) correspond to each other. To that end, we have abused a bit the meaning of the tag `<prop>`, which, according to the specification "is used to define the various properties of the parent element".

**Pro:** It treats sub-segmental correspondences as translation units, which they obviously are, and encodes them as such using element `<tu>`. **Con:** There is no easy way —apart from using a cumbersome way to overload `<prop>` elements with lists— to annotate more than one TU with this subsegment, and one would end up having similar problems as with GRaF.

### 5.2. Using inline elements (content markup)

#### 5.2.1. Using `<hi>`

One possible alternative is offered by the inline element `<hi>` (which stands for *highlight*). According to the TMX specification, it may be used to delimit "a section of text that has special meaning"; then, an atribute `x` may be "used to match inline elements [such as] `<hi>` between each `<tuv>` element of a given `<tu>` element". It may also carry an attribute `<type>`. Furthermore, according to the specification,[7] `<hi>` elements may be nested. The usage of the optional attribute `type` could be stretched to identify the source of information used to match the sub-segments.

The running example (where only the sub-segment pair (*einen Artikel*, *an article*) has been marked) could be marked as follows:

```
<tu segtype="sentence" tuid="13123123">
 <tuv xml:lang="de">
   <seg>Ich habe
   <hi x="1"
   type="google-translate-de-en">einen
   Artikel</hi>
   geschrieben.</seg>
 </tuv>
 <tuv xml:lang="en">
  <seg>I have written
  <hi x="1"
  type="google-translate-de-en">an
  article</hi></seg>
 </tuv>
</tu>
```

**Pro:** This allows for a rather rich annotation of subsegment correspondence without having to stretch too far the intended semantics of the `<hi>` element. **Con:** It does not allow for the annotation of overlapping sub-segments — for instance, one could not annotate *habe einen Artikel* and *einen Artikel geschrieben* simultaneously, as this would generate overlapping elements, which are not well-formed XML.

#### 5.2.2. Using `<bpt>` and `<ept>`

In TMX, elements `<bpt>` and `<ept>` are used in pairs to mark the beginning and the end of a paired sequence of *native codes* (that is, formatting information). Each `<bpt>` has a corresponding `<ept>` element within the segment. The attribute `x` is used as in the case of `<hi>`, to match them between different `<tuv>` elements; an additional attribute, `i`, matches each `<bpt>` with its `<ept>`; the elements contain the "begin formatting sequence" and the corresponding "end formatting sequence" respectively. This allows for overlapping spans. According to the specification, "this mechanism provides TMX with support to markup a possibly overlapping range of codes. Such constructions are not used often, however several formats allow them ". An example (taken from the TMX specification) would be the (invalid) HTML segment:

```
<B>Bold, <I>Bold + Italic</B>, Italic</I>.
```

A translation unit containing the above segment and its Spanish translation would be

```
<tu segtype="sentence" tuid="877">
 <tuv xml:lang="en">
  <seg>
   <bpt i="1" x="1">&lt;B></bpt>Bold,
   <bpt i="2" x="2">&lt;I></bpt>Bold +
   Italic<ept i="1">&lt;/B</ept>,
   Italic<ept i="2">&lt;/I>.</ept>
 </seg>
 </tuv>
 <tuv xml:lang="es">
  <seg>I have written
   <bpt i="1" x="1">&lt;B></bpt>Negrita,
   <bpt i="2" x="2">&lt;I></bpt>Negrita +
   Cursiva<ept i="1">&lt;/B</ept>,
   Cursiva<ept i="2">&lt;/I>.</ept>
 </tuv>
</tu>
```

Furthermore, the specification allows both `<bpt>` and `<ept>` to be empty (i.e., containing no formatting information); therefore, they could be repurposed to delimit corresponding sub-segments — as if a "null format-opening tag" and a "null format-closing tag" were delimiting each subsegment. As above, the attribute `type` may be used to identify the source of information used, or some other linguistic information annotating the subsegment.

The example above (where only one sub-segment pair has been marked) could be marked as follows:

[7] `<!ELEMENT **hi** (#PCDATA|bpt|ept|it|ph|**hi**|ut)*>`

```
<tu segtype="sentence" tuid="13123123">
 <tuv xml:lang="de">
   <seg>Ich habe
   <bpt i="1" x="1"
   type="google-translate-de-en"/>einen
   Artikel<ept i="1"/>
   geschrieben.</seg>
 </tuv>
 <tuv xml:lang="en">
  <seg>I have written
  <bpt i="1" x="1"
  type="google-translate-de-en"/>an
  article<ept i="1"/></seg>
 </tuv>
</tu>
```

Overlapping sub-segment correspondences could also be marked

```
<tu segtype="sentence" tuid="13123123">
 <tuv xml:lang="de">
   <seg>Ich
   <bpt i="1" x="1"
   type="google-translate-de-en"/>gehe
   <bpt i="2" x="2"
   type="google-translate-de-en"/>
   ins<ept i="1"/>
   Haus<ept i="2"/>.</seg>
 </tuv>
 <tuv xml:lang="en">
   <seg>I
   <bpt i="1" x="1"
   type="google-translate-de-en"/>go
   <bpt i="2" x="2"
   type="google-translate-de-en"/>
   into the<ept i="1"/>
   house<ept i="2"/>.</seg>
  <seg>I have written
  <bpt i="1" x="1"
  type="google-translate-de-en"/>an
  article<ept i="1"/></seg>
 </tuv>
</tu>
```

In the last example, the German sentence *Ich gehe ins Haus* and its English counterpart *I go into the house* receive two overlapping annotations. One (`x="1"`) relates *gehe ins* with *go into the* and the other one (`x="2"`) relates *ins Haus* with *into the house*.[8]

**Pro:** This method allows for a very general annotation of all kinds of subsegment correspondences. Also, a closely related standard, XLIFF,[9] which represents one or more documents being localized (translated), may also use and propagate (slightly different) `<bpt>` and `<ept>` annotation, as the definition of the segment contents `<seg>` is actually similar to that in TMX. **Con:** This scheme implies a repurposing of the semantics of tags `<bpt>` and `<ept>` which may not be acceptable and could give trouble with CAT systems that do take care of them instead of just stripping them from each `<seg>`[10] and does not explicitly encode sub-segment correspondences as translation units `<tu>` (although one could use unique `tuid` identifiers for sub-segment TU, encoded as in section 5.1. as the value of `x` across the whole TM).

Figure 1 shows an example of the application of this method to the example sentence pair discussed in section 2..

## 6. Concluding remarks

This paper presents possible alternatives to enrich TMX-encoded translation memories with information about subsegment equivalence, which may be obtained from external sources such as machine translation systems, term bases, glossaries, etc., or by using a statistical word aligner followed by *phrase* extraction. The resulting enriched TMX file encodes the information needed for *advanced leveraging* of the translation memory, that is, for a more efficient usage of sub-segment information present in the translation memory.

The study describes these alternatives and discusses their pros and cons. In particular, it indicates possible ways to reinterpret or repurpose existing elements in the TMX standard to annotate the sub-segment equivalences found in translation units. Tentatively, the use of empty versions of the "begin paired tagging" (`<bpt>`) and the "end paired tagging" (`<ept>`) elements, which usually contain and match begin-of-formatting and end-of-formatting tags between segments in different languages, described in Section 5.2.2., seems to be the one that holds most promise, as it allows for indefinitely nested or overlapped sub-segment equivalence annotation.

**Acknowledgements:** Support from the Spanish Ministry of Economy and Competitiveness through grant TIN2012-32615 is gratefully acknowledged. I also thank Felipe Sánchez-Martínez and Juan Antonio Pérez-Ortiz for interesting suggestions.

## 7. References

Miquel Esplà-Gomis, Felipe Sánchez-Martínez, and Mikel L Forcada. 2011. Using machine translation in computer-aided translation to suggest the target-side words to change. In *Proceedings of the 13th Machine Translation Summit*, pages 172–179, Xiamen, China, September.

Ignacio Garcia. 2012. Machines, translations and memories: language transfer in the web browser. *Perspectives: Studies in Translatology*, 20(4):451–461.

N. Ide and K. Suderman. 2007. GrAF: A graph-based format for linguistic annotations. In *Proceedings of the Linguistic Annotation Workshop*, pages 1–8.

Philipp Koehn. 2010. *Statistical machine translation*. Cambridge University Press.

[8] As noted by one of the reviewers, the correspondence between *go into the* and the *gehe ins* subsegments would only be acceptable when the subject of *go* is 1st person singular and the German noun following *ins* is neuter singular, as in the example.

[9] `http://docs.oasis-open.org/xliff/xliff-core/xliff-core.html`

[10] OmegaT, `http://www.omegat.org`, the main free/open-source computer-aided translation program, simply strips that information off.

```
<tu segtype="sentence" tuid="48123">
 <tuv xml:lang="es">
   <seg>
   <bpt i="1" x="1" type="mt-es-en"/>
   <bpt i="8" x="8" type="mt-es-en"/>La<ept i="1"/>
   <bpt i="6" x="6" type="mt-es-en"/>
   <bpt i="2" x="2" type="mt-es-en"/>situación<ept i="2"/>
   <bpt i="3" x="3" type="mt-es-en"/>humanitaria<ept i="3"/>
   <ept i="6"/>
   <ept i="8"/>
   parece
   <bpt i="4" x="4" type="mt-es-en"/>
   <bpt i="7" x="7" type="mt-es-en"/>ser<ept i="4"/>
   <bpt i="5" x="5" type="mt-es-en"/>difícil<ept i="5"/>
   <ept i="7"/>.
   </seg>
 </tuv>
 <tuv xml:lang="en">
   <seg>
   <bpt i="1" x="1" type="mt-es-en"/>
   <bpt i="8" x="8" type="mt-es-en"/>The<ept i="1"/>
   <bpt i="3" x="3" type="mt-es-en"/>
   <bpt i="6" x="6" type="mt-es-en"/>humanitarian<ept i="3"/>
   <bpt i="2" x="2" type="mt-es-en"/>situation<ept i="2"/>
   <ept i="6"/>
   <ept i="8"/>
   appears to
   <bpt i="4" x="4" type="mt-es-en"/>
   <bpt i="7" x="7" type="mt-es-en"/>be<ept i="4"/>
   <bpt i="5" x="5" type="mt-es-en"/>difficult<ept i="5"/>
   <ept i="7"/>.
   </seg>
 </tuv>
</tu>
```

**Figure 1:** Sub-sentential annotation of the sentence-pair example in section 2. in TMX using the proposal in section 5.2.2.


Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–21.

Juan Antonio Pérez-Ortiz, Daniel Torregrosa, and Mikel L. Forcada. 2014. Black-box integration of heterogeneous bilingual resources into an interactive translation system. In *HaCAT 2014: Workshop on Humans and Computer-assisted Translation at EACL 2014*. forthcoming.

Richard Zens, Franz Josef Och, and Hermann Ney. 2002. Phrase-based statistical machine translation. In Matthias Jarke, Jana Koehler, and Gerhard Lakemeyer, editors, *KI*, volume 2479 of *Lecture Notes in Computer Science*, pages 18–32.